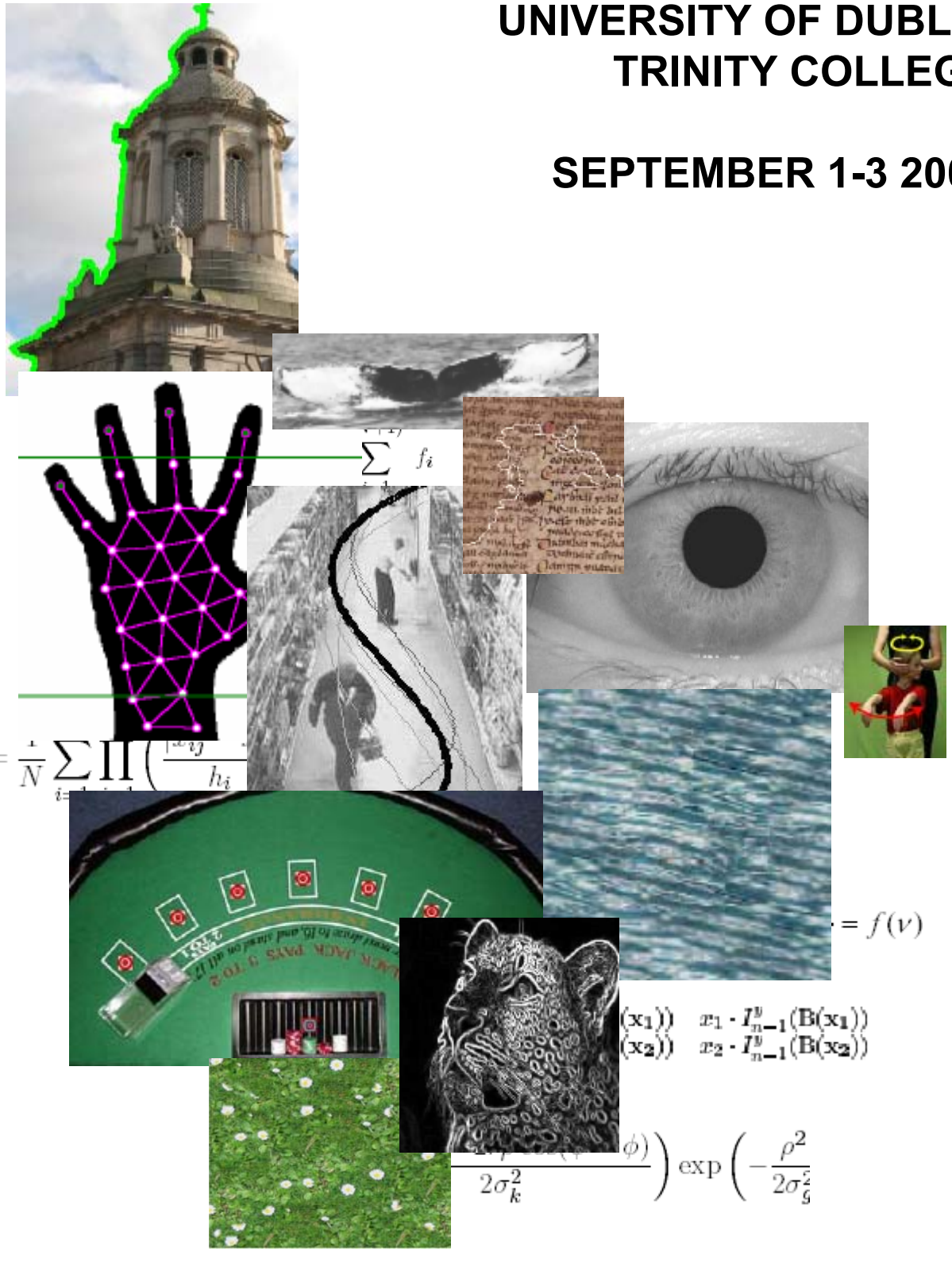
PROCEEDINGS OF THE IRISH MACHINE VISION CONFERENCE

Eds: K. Dawson-Howe, A. C. Kokaram and F. Shevlin

UNIVERSITY OF DUBLIN TRINITY COLLEGE

SEPTEMBER 1-3 2004



Foreword

On behalf of the organising committee, we would like to welcome all speakers and delegates to the 2004 Irish Machine Vision and Image Processing Conference, which is being hosted jointly by the Department of Computer Science and the Department of Electrical and Electronic Engineering, Trinity College, Dublin.

IMVIP 2004 is the eighth conference in the series. Previous IMVIP conferences have been organised by Magee College, University of Ulster (1997), NUI, Maynooth (1998), Dublin City University (1999), Queens University of Belfast (2000), NUI, Maynooth (2001), NUI, Galway (2002 in conjunction with Opto-Ireland) and University of Ulster, Coleraine (2003).

Once again IMVIP 2004 brings together theoreticians and practitioners, industrialists and academics from numerous related disciplines involved in the processing and analysis of image-based information. The initial call for papers was issued in January 2004. Sixty-five submissions were received and each of these was blind reviewed by members of the programme committee. Of these Twenty-two papers were accepted for oral presentation and a further fourteen were accepted for poster presentation.

We would like to thank the members of the programme committee for their help in the review process, without whom a conference of this nature would not be possible. Thanks are also due to the local organising committee including members of the Sigmedia Group; in particular Hugh Denman for his excellent work in creating the electronic submission system and Dr. Sid-Ahmed Berrani for his careful assistance in creating this proceedings.

We are grateful to our invited speakers for taking the time to present at the conference: Sarah Witt (Sony Research UK), Dr. Bill Collis (The Foundry) and James Mahon (Agilent Technologies).

IMVIP 2004 is run in association with the Irish Pattern Recognition and Classification Society (IPRCS), a member organisation of the International Association for Pattern Recognition (IAPR).

Kenneth Dawson-Howe Anil Kokaram Fergal Shevlin
Departments of Computer Science and Electrical and Electronic Engineering
Trinity College
Dublin 2
Ireland

Publisher: Trinity College Dublin, Ireland
Published September 1st 2004

Conference Chairs

Dr. Kenneth Dawson-Howe, Dept. of Computer Science, Trinity College, Dublin
Dr. Anil Kokaram, Dept. of Electrical and Electronic Engineering, Trinity College Dublin
Dr. Fergal Shevlin, Dept. of Computer Science, Trinity College Dublin

Local Organising Committee

Dr. Sid-Ahmed Berrani, Dept. of Electrical and Electronic Engineering, Trinity College Dublin
Mr. Hugh Denman, Dept. of Electrical and Electronic Engineering, Trinity College Dublin
Dr. Rozenn Dahyot, Dept. of Statistics, Trinity College Dublin
Dr. Laurent Joyeux, Dept. of Electrical and Electronic Engineering, Trinity College Dublin

Programme Committee

Dr. Sid-Ahmed Berrani, Dept. of Electrical and Electronic Engineering, Trinity College Dublin, Ireland
Prof. Frank Boland, Dept. of Electrical and Electronic Engineering, Trinity College Dublin, Ireland
Dr. Ahmed Bouridane, School of Computer Science, The Queens University of Belfast, UK
Prof. Michael Brady, Dept. of Engineering Science, Oxford University, UK
Mr. Don Braggins, Machine Vision Systems Consultancy, UK
Dr. T. J. Brown, School of Computer Science, The Queens University of Belfast, UK
Dr. Jonathan G. Campbell, Dept. of Computing, Letterkenny Institute of Technology, Ireland
Dr. Darryl Charles, School of Computing and Information Eng., University of Ulster Coleraine, UK
Ms. Sonya Coleman, School of Computing and Intelligent Sys., University of Ulster Londonderry, UK
Dr. J. Condell, School of Computing and Intelligent Systems, University of Ulster Londonderry, UK
Prof. Danny Crookes, School of Computer Science, The Queen's University of Belfast, UK
Dr. Rozenn Dahyot, Dept. of Statistics, Trinity College Dublin, Ireland
Dr. Kenneth Dawson-Howe, Dept. of Computer Science, Trinity College Dublin, Ireland
Dr. Robert Fisher, School of Informatics, University of Edinburgh, UK
Dr. Riad I. Hammoud, Delphi Automotive Systems, Indiana, USA
Dr. Conor Heneghan, Dept. Electronic and Electrical Engineering, University College Dublin, Ireland
Prof. David Hogg, School of Computing, University of Leeds, UK
Dr. Laurent Joyeux, Dept. of Electrical and Electronic Engineering, Trinity College Dublin, Ireland
Dr. Anil Kokaram, Dept. of Electrical and Electronic Engineering, Trinity College Dublin, Ireland
Mr. John Mc Donald, Dept. of Computer Science, National University of Ireland Maynooth, Ireland
Prof. Paul Mc Kevitt, School of Computing and Intelligent Sys., University of Ulster Londonderry, UK
Dr. Michael McNeill, School of Computing and Information Eng., University of Ulster Coleraine, UK
Dr. G. Moore, School of Computing and Mathematics, University of Ulster Newtownabbey, UK
Dr. Philip Morrow, School of Computing and Information Eng., University of Ulster Coleraine, UK
Dr. Noel Murphy, School of Electronic Engineering, Dublin City University, Ireland
Prof. Fionn Murtagh, School of Computer Science, Queen's University Belfast, UK
Dr. Hiroshi Sako, Hitachi Central Research Laboratory, Tokyo, Japan
Dr. Bryan W. Scotney, School of Computing and Information Eng., University of Ulster Coleraine, UK
Prof. Andy Shearer, Dept. of Information Technology, National University of Ireland Galway, Ireland
Dr. Fergal Shevlin, Dept. of Computer Science, Trinity College Dublin, Ireland
Prof. David Vernon, Etisalat University, UAE
Prof. Paul F. Whelan, Vision Systems Lab, School of Electronic Eng., Dublin City University, Ireland
Dr. John Winder, Life and Health Sciences, Univeristy of Ulster Newtownabbey, UK
Dr. Adam C. Winstanley, Dept. of Computer Science, National University of Ireland Maynooth, Ireland

Contents

PROCESSING VIDEO ON A PLAYSTATION2 AND GPUS

Sarah Witt
Sony Broadcast & Professional Research Labs,
Jays Close,
Viables,
Basingstoke,
Hants,
RG22 4SB, UK
email: sarah.witt@eu.sony.com

Abstract

This paper describes research carried out at Sony BPRL to investigate the real-time video processing capabilities of the Sony PlayStation2 and, more recently, consumer graphics cards (specifically Nvidia GeForceFX chipsets). The paper will illustrate some of the relative strengths of using these devices, but also some of the difficulties encountered. This paper will also contain a guide to the PlayStation2 architecture, to show how it can be used for this kind of application.
Keywords: Video processing, Graphics processors

1 Introduction

We received a DTL-T10000 PlayStation2 (PS2) developer kit towards the end of 2000, kindly lent by Sony Computer Entertainment Europe (SCEE) developer support in London. After some initial familiarisation work, and some time spent on 2D text and still image rendering work, in mid-2001 we started investigating the possibilities of using the PS2 for video applications.
Having worked on digital video effects devices before (large, expensive, hardware ones), it seemed like an interesting challenge to me to try and implement 3D nonlinear effects, such as pageturn and ripple, in the PS2. The fast 3D processing and graphics rendering power of the PS2 made it seem a suitable platform for such an application. As a result, many interesting video effects, both 3D and otherwise, were successfully implemented in the PS2 [1].
Recently, consumer graphics cards have become available that have similar, if not greater, potential to be able to process video in many ways. Using Nvidia GeForceFX-based graphics cards with video outputs, we have just started investigating implementing similar work in Graphics Processing Units (GPUs).

2 PlayStation2 Architecture

The following introduction to the PS2 architecture is by no means exhaustive, but is meant as a background to explain the way the effects were created, and how it might be used for other video processing applications.
The Sony PlayStation2 consists of four main processor devices, as shown in Fig.1.

- The main CPU (the “Emotion Engine”, or EE)
- The rendering engine (the “Graphics Synthesiser”, or GS)
- The IO Processor (or IOP)
- The Sound Processor (or SPU2)

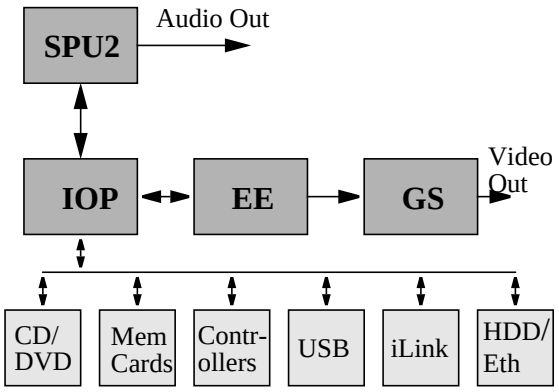


Fig.1 PlayStation2 Architecture

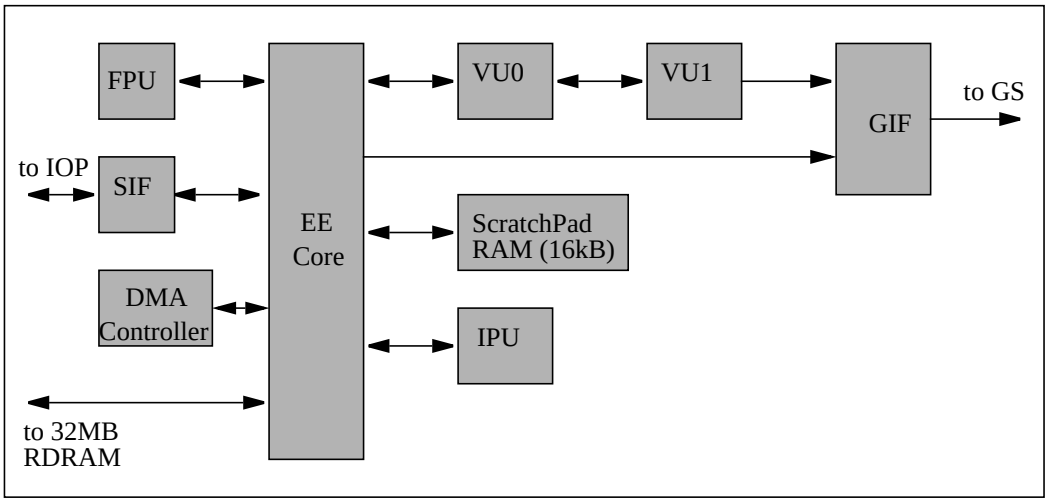


Fig.2 Emotion Engine Architecture

3 The *Emotion Engine*

The *Emotion Engine*, the main CPU in the PS2, is a 128-bit MIPS processor at its core. It operates at just short of 300MHz, and is supplied with 32MB of RAM. While such figures may sound modest compared to current desktop PCs, it is the PS2's potential for parallel processing which gives it its power.
Most of the PS2 software is written in C (C++ is also possible), using provided libraries to drive all the devices, but some parts are written in specific assembly languages, either because that was all that was possible, or to maximise efficiency.
In addition to 128-bit multimedia instructions, which can operate on sixteen 8-bit numbers, or eight 16-bit numbers (or four 32-bit numbers, or even two 64-bit numbers) in parallel, the EE also contains a number of co-processors, which can operate independently of, and in parallel with, the CPU core. These co-processors are illustrated in Fig.2.

3.1 Vector Units

The EE contains two floating-point vector processing units. These are each capable of processing 4 32-bit floating point elements of vectors in parallel. Most instructions (e.g. multiply accumulate) can be done in a clock cycle. As a result, a 4-element vector can be multiplied by a 4x4 matrix in just four clock cycles. Using four element vectors allows for homogenous 3D coordinates, so that translations, as well as rotations, can be done by matrix multiplications. This means that a single "Local to Screen" matrix can be used, instead of separate translations and rotations.
Normally, the vector units are used independently from the EE core. Instructions and data are sent to the VUs, and they are left to get on with their processing, and the results fetched on completion. The VU instructions are normally written using VU assembly language, which may be hard to read and write, but can be very efficient.
The two vector units differ in a number of ways. VU0 can be used as a co-processor directly from the EE core. VU1, however, can send data directly to the GS (through the Graphics Interface, GIF). VU1 also has more memory: 16kB each of data and instruction RAM, as opposed to VU0's 4kB each.

3.2 Image Processing Unit (IPU)

The IPU is essentially most of an MPEG-1 and MPEG-2 decoder. It can decode MPEG Fixed/Variable Length Coding (FLC/VLC) bitstreams, and perform Inverse Discrete Cosine Transforms (IDCTs) on the input coefficients to reproduce video. The only part of MPEG-1 and MPEG-2 decodes it is unable to do itself is the motion compensation in long GOP MPEG (such as is found on DVDs). This is normally carried out in the EE core.

I-frame (Intra-frame) only MPEG can be decoded completely in the IPU, so this is the format we have used. MPEG bitstream data can be DMA-ed to the IPU, and RGB video DMA-ed back on completion.

3.3 ScratchPad RAM

The 16kB ScratchPad RAM (SPR) is very fast access speed RAM (one clock cycle accesses, as opposed to several clock cycles from main RAM). Data can be DMA-ed between it and main memory. It is very useful for performing repetitive processes on large chunks of fixed-point data - for instance pixel colour operations, such as a chromakey.

3.4 DMA Controller

The DMA controller can send data between main memory and the following:
- to/from the IOP
- to/from the IPU
- to/from the ScratchPad RAM
- to VU0
- to/from VU1
- to GIF (to GS)

As the DMA Controller works on 128-bit quadwords, it is capable of transferring data between these devices very quickly, as well as being able to transfer in parallel to other operations running in the EE core. Consequently large amounts of data can be moved about within the PS2 very quickly and easily - a useful asset for video processing!

3.5 Graphics Interface (GIF)

The GIF receives data and commands from both the EE core and VU1, converts them for the physical interface between the EE and GS, and sends them to the GS. This interface is capable of data rates of over 1GB/sec. This allows fast transfer of textures to the GS - another feature useful for video. The GIF commands sent to the GS include, for instance, *xyz* coordinates, RGBA values, texture coordinates corresponding to the *xyz* coordinates, and configuration commands for texture mapping, alpha-blending, etc.

4 The *Graphics Synthesiser*

The GS is the rendering engine of the PS2. It can draw primitives, such as points, lines and polygons, according to instructions and coordinates sent from the EE. Primitives can be Gouraud shaded, texture mapped, alpha-blended and fogged. Z buffers can also be used to ensure that only primitives in front of those already visible are drawn. It operates at 150MHz, and can render 16 pixels per cycle (or eight per cycle if texture mapped).

4.1 GS Local RAM

The GS contains 4MB of embedded RAM. While this is much smaller than that found in typical PC graphics cards, the fact that it is embedded in the chip means that it can be connected using very wide buses (2048 data bits), thus allowing for parallel pixel rendering. In fact the frame buffer bandwidth is nearly 40GB/sec.
The GS RAM would normally be divided into the following sections:
- Display frame/field buffer (16 or 32-bit RGBA)
- Draw frame/field buffer (16 or 32-bit RGBA)
- Z buffer (16, 24 or 32-bit)
- Texture buffer (16, 24 or 32-bit RGBA, or 4 or 8-bit indexes for Colour Look up Tables)
- Colour Look up Tables (CLUTs) (16 or 32-bit)

The divisions between these buffers are in fact arbitrary, and can change during a program, so that, for instance, a draw buffer can temporarily swap with a texture buffer, etc. The draw and display buffers are effectively

double buffers, so that while one field is being rendered, the previous one can be displayed. They alternate between fields.

4.2 Texture Mapping

Texture mapping requires a texture coordinate that corresponds to each vertex coordinate sent to the GS. As each pixel is rendered, its corresponding texture pixel is read from the texture buffer and applied to the rendered pixel. For intermediate texture coordinates, bilinear filtering between adjacent texture pixels (texels) can be used.
Texture mapping can be done in the GS with or without perspective correction. Perspective correction produces a much better effect when mapping a texture to a 3D shape. Without it, the texture appears to be mapped in a flat, 2D way to the polygon coordinates. To perform perspective correction, as part of the "Local to Screen" mapping done in the EE, the value (called *Q*) used to normalise the homogenous 4-element vertex vectors is sent to the GS. This *Q* value is an indication of the level of scaling, caused by perspective, applied to a vertex. By extending the vertex values over the polygon, the perspective can be corrected for in each pixel to be rendered.
The same *Q* value can be used for Mip-Mapping. This is a process whereby a series of textures are created - normally the same image in successively smaller sizes (half width and height, then quarter width and height, etc.), and all of them stored in the texture buffer. These images are called MipMaps. Then, depending on the amount of scaling applied to a texture as it is mapped to the screen, different MipMaps can be used for the texture. For intermediate levels of scaling, a mixture of two adjacent MipMaps can be used. This process can be used to reduce image aliasing when a texture is reduced in size as it is mapped to the screen.

5 The *IO Processor*

The IOP has the same core as the processor used in the original PlayStation, and is therefore used for running PS1 games. But for PS2 software, its main function is to deal with all the peripherals that the PS2 can take. It can be programmed separately from the EE, and has 2MB of its own dedicated RAM. In the case of our video processing software, its principal function is to read the MPEG data off the hard disc drive efficiently and separate it into chunks of a single field.

6 PlayStation2 Video Effects

The video effects implemented in the PS2 include: wipes; 3D nonlinear effects, such as pageroll and ripple (as illustrated in Figures 3 and 4 below); and pixel colour based effects, such as chromakey, and an "old film" effect.

Fig.3 Pageroll effect

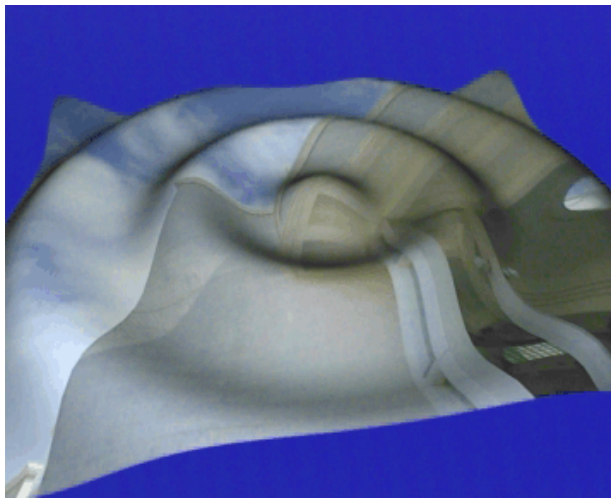


Fig.4 Ripple effect

6.1 3D nonlinear effects

These effects are created by dividing the foreground into horizontal triangle strips, creating a tile mesh (each tile is approximately eight pixels square). These tile vertices have a nonlinear effect applied to them first. For instance, in the case of a ripple effect, the *Z* value (distance to/from the screen) is modulated by a sine wave, with a phase that can change every field. Then, once the nonlinear part has been applied, 3D linear rotations and translations can be applied, along with some lighting, to enhance the 3D appearance. Overall, the nonlinear 3D effects are optimised to make as much use as possible of the parallel processing capabilities of the PS2. VU1 can be processing the linear part of the transform, and the lighting, for one triangle strip, while the EE core, with help from VU0, can be calculating the nonlinear part of the transform for the next triangle strip. Also, at the same time, the GS can be rendering the previous triangle strip, sent from VU1. While all this is happening, the IPU is also decoding the next fields of MPEG video, while the IOP is busy reading further MPEG data off the HDD, and separating out the audio, which SPU2 then plays out.
The overall data flow can be seen in Figure 5.

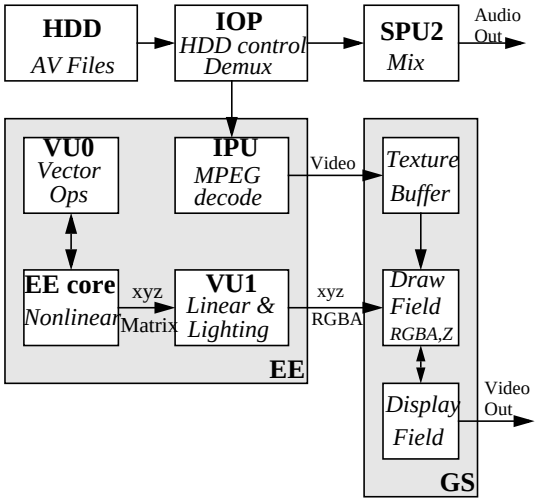


Fig.5 Use of PlayStation2 for nonlinear 3D effects

6.2 "Old Film" effect

The "Old Film" effects implemented reproduce typical film artefacts to make the video look as if it was shot on film some decades ago (how many decades depends on the type of film simulated!). The effect combines the following aspects:

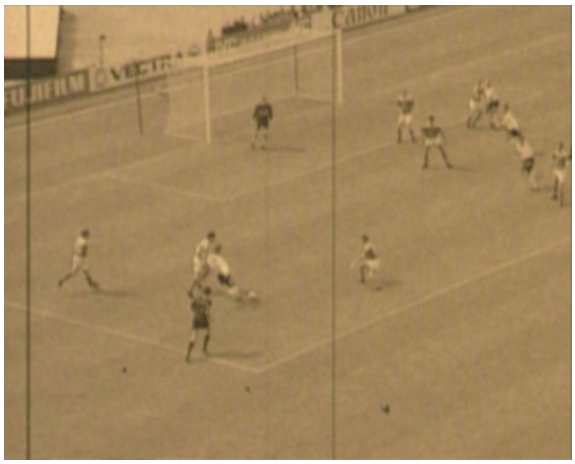


Fig.6 Typical "Old Film" frame

1.As film has no interlace, and only 24 frames/second, only one field of source video is used per frame;
2.The "film" can be colour, black & white, or sepia;
3.Film grain, blotches (specks of dirt) and scratches are added to the image;
4.A random shake is applied to the picture;
5.A random level of defocus is also applied, the level changing every frame;
6.Frames are dropped occasionally, to give jerky motion.

A typical sepia "old film" frame is shown in Fig.6.

The "old film" effect is applied almost entirely in the GS. The EE only has to decode the original video into luminance, rather than RGB, for black and white or sepia film.

6.3 Chromakey Effect

In contrast to the effects described so far, the chromakey effect (that of replacing a blue or green screen background with an alternative image) uses the EE core to perform the bulk of the processing. To speed up EE core processing, the RGB video data is sent in small batches to the Scratch Pad RAM (SPR, see above), and the 128-bit multimedia instruction used to process four 32-bit pixels at a time. The multimedia instructions are all

fixed-point (various bit-widths can be used), and are written in MMX-like assembly code. This processing takes up the majority of the time available per field.

7 Nvidia GPUs

The research we have carried out so far with consumer graphics cards has concentrated on the Nvidia GeForce-FX series of chipsets. These range from the GeForceFX5200, available for a few tens of pounds, to the GeForceFX5950Ultra, which can cost up to about £300. Most of the research has been carried out on the latter device, on a MicroStar International (MSI) card with video inputs and outputs, but we have also used GeForceFX5700-based cards. The higher-numbered GeForceFX chipsets are more powerful, running at faster clock speeds and performing more parallel operations. They also have a higher memory bandwidth. However, different versions of the various chipsets exist, with different specifications. At the time of writing, the newer 6800-series Nvidia devices are just becoming available - these will be more powerful still.

7.1 Development Setup

As mentioned above, most of the research has been implemented using a MSI board containing a GeForceFX5950Ultra device. This was plugged into a 8xAGP slot in a 3GHz single-CPU PC running RedHat Fedora Linux. The reasons for choosing Linux at this stage are many and varied, but include transparency, flexibility, and fundamentally, the ability to output 50Hz PAL video! Nvidia provide full Linux drivers (unified for all their current chipsets) some documentation, and sample application code on their website.

7.2 OpenGL

Whilst the Windows DirectX drivers may provide some functionality not available with OpenGL on Linux, using OpenGL means that the software can reasonably easily be ported to Windows or other operating systems at a later date, if required.
OpenGL is capable of rendering graphics either mostly on GPUs, or if the GPU is not sufficiently powerful, to perform some of the processing on the CPU instead. This makes OpenGL completely portable, not just between operating systems, but also across different graphics cards and PC setups. While this portability ensures that OpenGL graphics programs will run on just about any hardware, it does make the software very abstracted from the hardware on which it is running, making hardware-specific optimisations difficult.

7.3 Cg

A few years ago, Nvidia introduced their Cg language. This is a high-level shading language that can be used to program the GPUs. The GPUs can run both vertex programs and fragment (pixel) programs. These Cg programs can be loaded by OpenGL running on the CPU, and also have parameters controlled by the CPU. The Cg programs tend to be short (only a few lines), although current GPUs allow for longer programs, due to increased numbers of registers available in their vertex and fragment processors. Each vertex program is run from scratch on every vertex sent to the vertex processor by the CPU, and outputs vertex and texture data to the rasterisers, which then pass texels and other data to the fragment processors. Again, the fragment processors run their Cg programs on every pixel to be rendered, and can receive parameters from the CPU. The GPUs, and hence the Cg code, use 32-bit floating point numbers for each RGB and alpha channel (i.e. 128 bits per pixel), as well as for vertices and other parameters internally. This increases the quality of the final image, particularly if multiple rendering passes are used, over the standard 8-bit fixed-point used previously, and in the PS2. However, the drawback of using four times as many bits per pixel is the impact it has on the memory bandwidth.
Other similar shading languages now exist, such as OpenGL shading language, and Microsoft's DirectX HLSL (High-Level Shading Language). We chose to use Cg as it is now fairly well established, and hence is well supported with documentation and sample code.

7.4 Video

As mentioned above, the graphics cards that we used for research had video outputs (and in some cases, inputs). These were analogue S-Video interfaces (or analogue composite). As we are principally interested in processing standards-compliant video formats, rather than arbitrary frame sizes, having a video output is important. The graphics cards used are capable of outputting both standard definition (PAL and NTSC) video, as well as high definition (including 1920x1080i, and 1280x720p) - another aspect of interest for us.
Unfortunately, at the time of writing, we have not yet succeeded in getting the video inputs to work, due to an apparent lack of information, or driver, to do so. Consequently, the video processed on the graphics cards has been decoded in the CPU (from MPEG or DV sources), and sent via the AGP to the card.
The GPUs also contain some video processing hardware themselves, including parts of MPEG decoders (IDCTs and motion compensation, but not bitstream parsing). However, we have not yet started using these, partly due again to a lack of information/driver, but also because we haven't really needed to yet (a 3GHz CPU is more than capable of decoding two standard definition video channels)!

8 PlayStation2 / GPU comparison

It would be possible just to present a table of numbers, statistics (e.g. polygons/second, etc.), and checkboxes to compare the relative merits of using a PlayStation2, or a PC with particular graphics card, to process video. However, whilst these numbers and facts can possibly be used to indicate whether a particular requirement can be achieved or not, they scarcely tell the full story. This section will therefore try to illustrate the pro's and con's of each system, based on the author's experience of developing video-processing software on both.
But first, it is necessary to provide some numbers, for reference later.

Table 1: Numerical comparisons between PS2 and CPU/GPU

Parameter	PlayStation2	3GHz CPU & GeForceFX5700	3GHz CPU & GeForceFX5950
CPU speed	300MHz	3GHz	3GHz
CPU bit-width	128	32	32
Vertex processor	VU0, VU1	On GPU	On GPU
Vertex proc speed	300MHz	475MHz	475MHz
Vertex proc bits	128	128	128
Vertices/sec	150 million	356 million	356 million
Fragment/pixel processor	Limited, on GS	On GPU	On GPU
Pixel proc speed	150MHz	475MHz	475MHz
Bits/pixel	32 fixed	128 floating	128 floating
Pixels/sec	2.4 billion	1.9 billion	3.8 billion
VRAM bandwidth	38.4GB/s	14.4GB/s	30.4GB/s
VRAM capacity	4MB	128 - 256MB	256MB

Table 1: Numerical comparisons between PS2 and CPU/GPU

Parameter	PlayStation2	3GHz CPU & GeForceFX5700	3GHz CPU & GeForceFX5950
CPU->VRAM bandwidth	1.2GB/s	2.1GB/s (8xAGP)	2.1GB/s (8xAGP)
VRAM->CPU bandwidth	1.2GB/s	132MB/s (PCI)	132MB/s (PCI)
Video Input	No (but yes on PSX)	Yes (depends on card)	Yes (depends on card)
Video Output	Yes	Yes	Yes
MPEG decoder	Mostly (bitstream, IDCT)	Mostly (IDCT, motion comp.)	Mostly (IDCT, motion comp.)
Anisotropic filtering	No	Yes	Yes

[Nvidia figures from www.nvidia.com]

8.1 CPU performance

The CPU performance is not particularly critical in video-processing applications, as long as it is sufficient to decode the required number and size of video streams, and control the processing carried out elsewhere. For the PS2, the EE core is, however, needed to do more complex pixel operations, such as chromakey. For this, fast access memory (SPR) and wide bit-width multimedia instructions are invaluable.

8.2 Vertex Processors

The vertex processor performance in all cases is safely in excess of that likely to be required for this kind of application, and they can all be run in parallel with other processors. Vertex processors are more critical for high quality rendering of synthesised images.

8.3 Fragment/pixel processors

The fragment, or pixel processor is one area where the GPU out-performs the PS2. The PS2 has limited pixel operations in the GS (alpha-blending and testing, and Z-testing, bit-masking, etc.). More complex pixel operations, such as a chromakey, have to be performed further upstream in the EE core, using multimedia instructions. Clearly, in the CPU/GPU combination, the CPU could also be used in a similar way, but the presence of a fully-programmable fragment processor provides the capability to perform many more operations per pixel in real time. The use of Cg or other high-level shading languages also reduces the time required to implement processing algorithms in these processors.
For more straightforward rendering (i.e. including any functions that the PS2's GS can implement on a pixel), the pixel fill rates are comparable between the different systems.

8.4 Graphics Memory (VRAM)

It is clear from the figures quoted above that the PS2 has very little GS RAM, compared to the amounts of VRAM provided on the graphics cards. However, as the bandwidths available to the RAM, possible because the GS RAM is embedded in the GS, and between EE and GS are greater on the PS2, this does not normally cause problems. However, the size of the GS RAM does impose one limitation: on the size of video image that can be rendered. To allow for double-buffered frame (field) buffers, a Z buffer, and

video-sized texture buffer (not absolutely necessary, as the texture can be used in stages), the video size is effectively limited to standard definition. In fact, 4MB of RAM is not even sufficient to store one field of 1920x1080 HD video (1920x540x4 bytes per pixel).
Where the PS2 does have the upper hand, however, is the bandwidth to the VRAM. The values of bandwidth (tens of GB/sec) may seem massive, but they can potentially start to become a bottleneck, and of course the figures quoted are necessarily theoretical maxima - the actual values are likely to be much less. Consider, for instance, standard definition video (720x576x25, or 720x480x30 = 10.4Mpixels/sec, @4bytes/pixel, this is 41.5MB/sec): the available bandwidth is still hundreds of times larger. However, this 41.5MB/sec bandwidth figure must be multiplied up many times: for a typical rendering operation, the current pixel value, current Z value, and at least one texture value must be read from the frame buffer, and video and Z written, per pixel, making a total of at least five memory accesses per pixel rendered. If multiple textures are used per pass (only possible on the GPUs, not in the PS2), MipMapping is applied, or depending on the transform applied to the texture(s), many more than one texture sample may need to be read per pixel rendered, the required bandwidth increases further. Also, for the GPUs, the pixel operations normally use 128 bits pre pixel, increasing the bandwidth even more. For typical video processing operations, multiple rendering passes are required. Also, if fullscreen anti-aliasing is applied, the images may become effectively 16 times as large (four times in each of two dimensions).
As the fragment/pixel processors rely on the VRAM bandwidth to read and write their required data, if a large percentage of the time available each field/frame is taken up with memory accesses, this can substantially reduce the possible number of operations that can be applied to each pixel. For the GeForceFX5700 processor, the 14.4GB/sec quoted bandwidth is likely to become a bottleneck - and indeed, with early versions of our video processing software, we have found this to be the case. This bottleneck can be compounded in normal CPU/GPU use, as it is likely that two screens will be used: a high-resolution and frame-rate VESA monitor, as well as the video out.

8.5 CPU<->VRAM bandwidth

With the advent of PCI-Express graphics cards, this potential bandwidth bottleneck could be removed, as the bandwidth will be 4.2GB/sec in both directions. Currently, the bandwidth in the direction of the graphics card (or GS) is not too much of a problem (as long as the graphics card is in a 8xAGP slot), as for two streams of video sent as textures, only 83MB/s is required. However, the reverse direction is currently a severe bottleneck for graphics cards, if the final output video image is to be read back from the graphics card, as a single standard definition video stream takes up a third of the maximum bandwidth available (high definition video is impossible).

8.6 Low-level control

One major difference that seems apparent from working on both PS2 and GPU systems, is the amount of low-level control available. For the PS2, both the official developer kit, and the PS2 Linux kits come with comprehensive hardware manuals. While at the start of development on the platform, this wealth of information may seem overwhelming, it does provide the programmer with a fantastic visibility and direct control of the hardware. Also, as there is no multi-layered operating system taking its share of the available resources, the PS2 programmer has absolute control of the whole system. This makes it much easier to optimise code, as it can be deduced what functions actually do at a hardware level. Another benefit at this level that the PS2 provides is genuinely unified graphics memory. Any part of the 4MB of RAM inside the GS can store frame, Z, or texture, or in fact change between type at the whim of the software. This enables, for example, what was the frame buffer to now be considered a texture and re-rendered to what was the texture buffer, and is now the frame buffer. To achieve the same effect in OpenGL on a GPU would require copying from the frame buffer to another texture buffer, etc. - all of which adds to the memory bandwidth required, as well as processing time.
OpenGL is a fairly abstract and object-oriented language. As mentioned above, this has the benefit of making it very portable between hardware and operating systems, but does have the strong disadvantage of being somewhat obfuscatory when it comes to determining what the hardware is actually doing. This

is further not helped by a lack of hardware information available about the GPUs themselves from Nvidia. This lack of control and information can only lead to processing on GPUs being somewhat inefficient, relative to what is possible on a PS2.
Another problem with OpenGL is that it is really designed for rendering synthesised images, not processing ones that already exist. It seems likely that later versions of OpenGL (OpenGL2 is imminent, and later versions than that already being discussed) will deal better with processing video, as more and more developers see the potential in graphics cards to do so. This may increase support for streaming textures, and allowing buffers in the VRAM to be multi-purpose.

8.7 Development time

It is hard to compare development times between the two platforms fairly, as the PS2 came first, and it was the author's first experience of graphics hardware (as opposed to dedicated video hardware). However, it is probably fair to say that the PS2 is harder to get started with (although this is very likely improved now, with the amount of support information available, which has increased enormously since the PS2 was first released). OpenGL and Cg are reasonably straightforward to learn, and simple applications can be got working very quickly indeed. However, as explained above, at a certain point, it is useful to have more visibility of the hardware, to allow for optimisation, or implement certain specifications. At this point, the PS2 probably becomes the preferable platform.

9 Conclusion

With graphics processors (either in games consoles or normal PC peripherals) advancing at a far higher rate than CPUs, their use as a platform for processing video is likely to increase correspondingly. Manufacturers are beginning to recognise this, and are including more and more video-specific components in these processors. However, there is still some way to go before they have all the functionality in place to be the automatic choice for processing video, as they are still primarily designed for other uses. Currently, though, they can be used to implement some interesting and fairly powerful processes on video streams.

Acknowledgements

The author would like to acknowledge and thank Sony Computer Entertainment Europe developer support in London, and Sony PSNC in Atsugi, Japan. Thanks must also go to colleagues Alan Turner, Mike Williams and Matt Spencer for their help with the GPUs.

References

[1] Sarah Witt (2004). "Real Time Video Effects on a PlayStation2". *IEE CVMP Proceedings*, March 2004.

Information Retrieval from Image Databases: The Case of Automated Grading of Industrial Materials

Xiaoyu Qiao (1), Fionn Murtagh (1), Paul Walsh (2),
P.A.M. Basheer (2), Adrian Long (2), and Danny Crookes (1)
(1) School of Computer Science (2) School of Civil Engineering
Queen's University Belfast, Belfast BT7 1NN

July 22, 2004

Abstract

The success of content-based image finding and retrieval is most marked when the user's requirements are very specific. An example of a specific application domain is the grading of engineering materials. In this paper we describe such an application in the area of civil engineering construction materials. We describe an innovative solution to automated vision-based grading of construction materials. From the methodology viewpoint, we show the advantages of using a resolution scale based approach for content characterization of mixtures of fine granularity material and large granularity "aggregate". In particular we use (i) multiscale entropy; and (ii) significant wavelet coefficients. Links with recent vision model perspectives are discussed.

Keywords: Machine vision, aggregate, construction, wavelet transform, entropy, information, image database

1 Introduction

1.1 Information Retrieval from Image Databases

Driven by generic multimedia database applications, there has been much work in recent years on the theme of content-based image retrieval. Support has included cultural heritage and museum applications, and personal and journalist digitized photograph collections. Querying in generic database applications has often involved used of colour and object (sub-image) shape properties.

Retrieval from specialized databases (e.g. fingerprint databases, or astronomy image stores, or Earth observation databases) has usually availed of specialized image features. Users of information retrieval systems in these specialized areas are most often professional experts. Thus, for example, Earth observation imagery is usually queried on the basis of well-defined external properties: dates of observation, resolution scale of detector, geographic coordinates.

In the work described in this paper we are dealing with information retrieval support using a specialized image database of engineering materials. In

this context, we propose a range of general features, derived from the images. These features are defined from image resolution scale-based gradient, energy and entropy properties.

The context of information retrieval here is automated grading of materials. In this paper, we describe our current work of system evaluation and validation.

1.2 The Civil Engineering Application

In terms of quality control the civil engineering crushed aggregate construction sector is relatively underdeveloped, with only simple manual tests being applied to the end product. From a cost-benefit perspective it is therefore essential that maximum economic value be obtained from the quarried stone, which will require wastage to be eliminated from each stage of the processing chain. The quality of the aggregate produced in terms of the consistency of its size and shape also has a major influence on the quality (particularly in relation to workability and durability) of the concrete and blacktop mixes subsequently produced.

Round or cubic shape aggregate particles have traditionally been considered the most suitable in relation to meeting the needs of industry, although it has also been suggested that bituminous mixes including non-cubic fractions can lead to better road pavement layer stability [1, 2].

The development of a rapid and efficient means for classifying aggregate size and shape could therefore enable the beneficial properties of an aggregate to be more fully exploited.

Aggregate sizing is carried out in the industrial context by passing the material over sieves or screens of particular sizes. Aggregate is a 3-dimensional material and as such need not necessarily meet the screen aperture size in all directions so as to pass through that screen. The British Standard specification (and American and other European specifications) suggest that any single size aggregate may contain a percentage of larger and smaller sizes, the magnitude of this percentage depending on the use to which the aggregate is to be put.

To monitor the range of size of aggregate particles produced from any particular screen, regular laboratory testing is carried out. This involves sampling the aggregate from either the moving conveyor belt or alternatively from the stockpile produced. A sieve analysis test is carried out to assess the range of particle sizes present in accordance with the specification. This test is time-consuming and therefore only a relatively small fraction (2 kg per 400–500 tons) of the aggregate produced is ever tested. The quality of the result also relies heavily on good sampling technique, which means that feedback to the quarry operators can be slow and in many cases unrepresentative.

Certain shape parameters are also specified for particular uses, the most common being Flakiness and the Elongation indices. These tests are also very labour intensive and time consuming, and are carried out on an even more limited number of samples.

1.3 The Content-Based Image Retrieval Problem

An ability to measure the size and shape characteristics of an aggregate or mix of aggregate, ideally quickly, is therefore desirable to enable the most efficient use to be made of the aggregate and binder available.

This area of application is an ideal one for image content-based matching and retrieval, in support of automated grading. Compliance with mixture specification is tested by means of match against an image database of standard images.

For our work, the image data capture consists of an experimental environment which can be replicated in an operational setting: limitation of 3D effects and occlusion; use of diffuse homogeneous light; and avoidance of shadow.

For each class of aggregate mix, four separate samples were taken. Following each imaging, randomization was carried out on the aggregate mix. To provide a good sample in the case of each image, a subimage of dimensions 454×341 was extracted from a central region of each image. We took 50 images to represent each of the following aggregate classes, giving 600 images analyzed. A further set of 108 images (9 from each class) were used for testing. Classes: passing 6 mm sieve hole diameter; 30/40 mix; 50/10 mix; 10 mm; 14 mm; 28 mm drb; 40 mm; 28 mm dbc; 20 mm; 50-14 wc; 35 14 mm; and 510BC mm.

Figure 1 shows a representative image from the first of these classes.

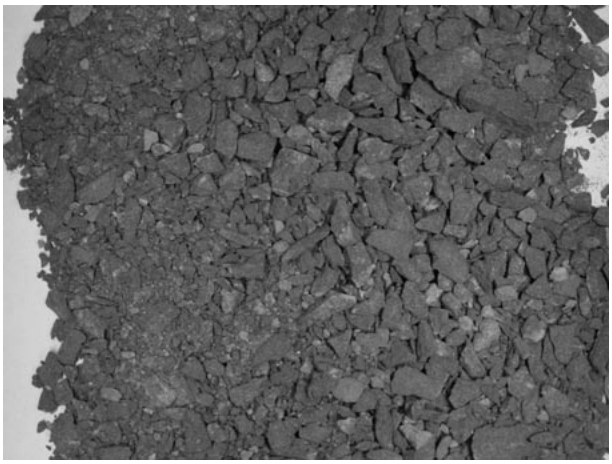


Figure 1: Image from class 1.

Our objective is to create an image-based "virtual sieve" which, through image matching against an image database of standard images, will provide automated grading. We use an unsupervised feature selection and multiple discriminant analysis approach, to support nearest neighbour image querying.

1.4 The Vision Model Perspective

For robotic and industrial images, the objects to be detected and analyzed are usually solid bodies. The appropriate vision model for such images is therefore based on the detection of the surface edges (see e.g. [3]). However diffuse structures are characteristic of many other fields, including remote sensing, hydrodynamic flows, astronomy, and biological studies. Specific vision models are needed in each case. Such a model could be one for which the image is the sum of a slowly variable background with superimposed small-scale objects. In [4], a vision model is proposed where each pixel, at each level of spatial resolution, with a value significantly greater than the scale-based background is considered

to belong to a real object. The same label is given to each significant pixel belonging to the same connected field, both spatially and in-scale (i.e. inter and intra wavelet band).

For the present work on civil engineering materials, we will use two vision models.

The first vision model will cater for fine grained material regions in an image. This image characterization is based on image entropy, determined by resolution scale (thereby catering for gradation in granularity size), and by spatial region (thereby catering for local subimage regions).

The second vision model will cater for coarse grained material regions in an image. It is based on significant wavelet coefficients at each resolution scale. Spatial (intra wavelet scale, and not inter scale) adjacency, only, is used to define object features. If it were necessary to analyze such objects, individually rather than globally, then inter scale analysis would be carried out.

2 Feature Selection

2.1 Multiple Scale Entropy to Quantify Aggregate Granularity

For fine grained image characterization we carry out a "texture" analysis, and the wavelet transform is, by now, a traditional way to do this ([5, 6, 7, 8, 9]). Our approach avoids any system parameter related to window size; and the undecimated wavelet transform used helps to avoid object aliasing.

We used multiple scale image entropy to quantify aggregate granularity. Using 5 wavelet scales, from the B_3 spline à trous redundant wavelet transform, an entropy-per-scale was determined, and thus provided a 5-valued feature vector for each image. Background on this is provided in [10], where it was concluded that this approach to feature definition performed well for discrimination of aggregate "textures".

We additionally used 5 wavelet scales, with the same wavelet transform method, used on the edge map, i.e. the image transformed with a Canny edge detector. In total, this provided 10 features per image.

A B_3 spline à trous wavelet transform gives the following decomposition of the original signal: $\{x_k \mid k = 1, 2, \ldots m\} = \{\sum_{j=1}^{l} w_{j,k} \mid k = 1, 2, \ldots m\}$. l is the number of scales, m is the number of samples in band (scale) m which is constant for this redundant transform. Scale l is the smooth or continuum scale, and all other scales consist of zero-mean (per scale) wavelet or detail coefficients. The value of l is set by the user (here, 6, implying 5 wavelet scales) and, given the dyadic property related to wavelet dilation, should be $< \log_2 m$. The feature set is defined from the resolution scale related decomposition as follows:

$$H = \{H_j \mid j = 1, 2, \ldots l-1\} = \{\sum_{k=1}^{m} h(w_{j,k}) \mid j = 1, 2, \ldots l-1\} \qquad (1)$$

with $h(w_{j,k}) = -\ln p(w_{j,k})$. The probability $p(w_{j,k})$ is the probability that the wavelet coefficient $w_{j,k}$ is due to noise. The smaller this probability, the more important will be the information relative to the wavelet coefficient. For

Gaussian noise we have

$$h(w_{j,k}) = \frac{w_{j,k}^2}{2\sigma_j^2} + \text{Const.} \tag{2}$$

where σ_j is the noise at scale j. If we were using a (bi-) orthogonal wavelet transform with an L^2 normalization, we would have $\sigma_j = \sigma$ for all j, where σ is the noise standard deviation in the input data.

2.2 Larger Aggregate Characterization

Larger pieces of aggregate are less well modelled using the "texture" oriented approach described above. We used, in addition to the features used so far, a set of features designed to describe larger pieces of aggregate.

Firstly, this "larger aggregate pieces" description was multiresolution based. We used 5 wavelet scales of a B_3 spline à trous redundant wavelet transform. This transform does not favour any orientation, and being redundant avoids decimation-related effects. If one assumes a simple statistical model, such as a Gaussian one, it is straightforward to determine the Gaussian parameters in the wavelet planes. Note that the wavelet planes are linear combinations of the original image pixel values, and that any linear combination of Gaussian-distributed random variables yields a random variable which is Gaussian. While thresholding based on this noise model has good theoretical credentials, we use this framework here as a heuristic to distinguish between interesting and uninteresting signal.

Feature 1 was the percentage of significant wavelet coefficients at each scale. Significance was determined from a 3σ threshold.

Feature 2 was the number of maxima at each scale.

Feature 3 was the number of structures (connected components of significant wavelet coefficients) at each scale.

Feature 4 was the size in pixels of the largest detected structure at each scale.

For 5 wavelet scales, using the aforementioned 4 features, this gave 20 features per image.

2.3 Multiple Discriminant Analysis

To facilitate assessment of discriminability between the classes in feature space, we used multiple discriminant analysis (also termed discriminant factor analysis, or the multi-class version of Fisher's linear discriminant analysis) [11, 12]. Discriminating axes are determined in this space, in such a way that optimal separation of the predefined groups is attained. As a linear discrimination method, we expect that such problems as training set size, and generalization, will be less pronounced than for a nonlinear method. See also Hand's [13] case for "simple is best" in regard to choice of classifier.

Consider the set of feature vectors, $i \in I$; they are characterized by a feature set, $j \in J$. A new orthogonal coordinate space is determined, such that the spread of class means in this new space is maximized, while the compactness of classes is restrained. Letting T be the total variance-covariance matrix of the n observations, and B be the between classes covariance matrix, we seek

eigenvectors of the matrix product $T^{-1}B$ associated with non-increasing eigenvalues. It can be shown that multiple discriminant analysis is equivalent to a principal components analysis of centred vectors, i.e. the group means, in the T^{-1} or Mahalanobis metric.

Having the transformed feature vectors, i.e. their projection in the discriminant factor space, allows straightforward nearest mean assignment of vectors to the closest among the 5 groups used. In discriminant factor space, the (unweighted) Euclidean distance is used.

3 Results

Figure 2 shows one example of the projected images in the principal discriminant plane, based on use of 5 classes. Classes 1, 2 and 5 are well separated. In the multidimensional space (inherent dimensionality 5 = minimum of numbers of: features less 1 due to centring; observation; and groups) the distinction between groups 3 and 4 is clearer. This figure used 250 images, characterized in 10-dimensional feature space. The first 5 features are the multiscale entropy ones described above. The second 5 features are multiscale entropies based on a Canny edge transformed image. (See section 2.1 above.) For five successive classes of image, among the 250 images used, we had numbers of images misclassified as: 0, 0, 3, 7, 0).

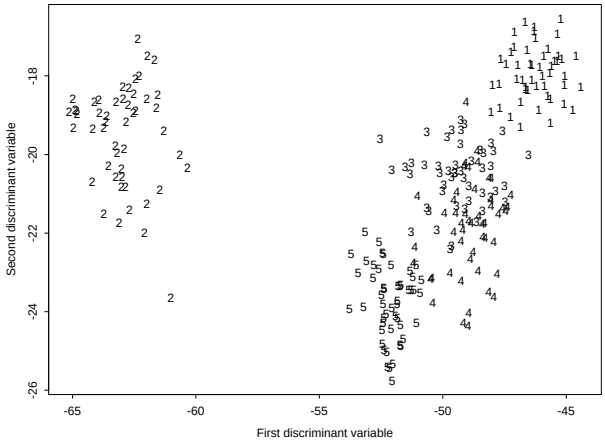


Figure 2: Principal discriminant factor plane, with projections of images from groups 1 to 5.

Among the many experiments carried out, a few important points are as follows.

1. We studied many different feature sets. Most were multiscale-based. In particular we initially used multiscale entropy features of a Canny edge transformed images to provide information on larger pieces of aggregate. But we later bypassed these Canny edge-based features in favour of the multiscale significant structures analysis described above.

2. We examined the denoising of the image data prior to analysis. This was not found to be of benefit.

3. Rather than the nearest mean classifier used in the multiple discriminant analysis, we also investigated nearest neighbour discrimination approaches (1-NN, 3-NN); and a multilayer perceptron. However the linear approach was found to give very good results, and its operation was easily controlled and managed.

4. We worked on rebinned 454×340 images, obtained from the originally sized 2272×1704 images. We exhaustively tested the processing used on a battery of 600 training set images used additionally on the original 2272×1704 images. Days of compute time on a Sun Microsystems cluster gave us an approximate gain of 2% misclassification on an original 12%. We saw no justification in continuing to process the original images in this way.

5. We may note that as long as the misclassification in a class is shown to be less than 50%, then a majority class assignment based on a number of images is likely to increase the success rate.

Using 25 features (see sections 2.1 and 2.2: entropy values for 5 resolution scales; and 4 significant wavelet features for 5 resolution scales, all per image) and a set of 600 images, we found the following results: for 12 classes, numbers of images misclassified were: 0, 0, 6, 5, 3, 6, 15, 12, 9, 0, 2, 0. This gave an overall misclassification rate of 9.7%.

We then took 108 unseen iamges, and determined their classes based on the 600 image result. We found the following: classes 1, 2, 3, 5, 11, 12: all perfect. Class 4: 2 incorrect out of 9; class 6: 8 incorrect out of 9; class 7: 2 incorrect out of 9; class 8: 4 incorrect out of 9; class 9: 8 incorrect out of 9; class 10: 2 incorrect out of 9. Overall the misclassification rate was 24.1% incorrect. This result was appreciably improved with the following information. Classes 6 and 9 were two different classes but with mixture specification bands (British Standard BS 812 Part 103 1985) that were identical. Given an identical specification, no image-based virtual sieve can possibly separate these classes. In all test set cases, a majority class assignment leads to correct assignment.

In order to study the needs for training and test set cardinalities, assessments were carried out. Five randomly chosen test sets (and correspondingly randomly chosen training sets) yielded numbers of images misclassified as: 1, 0, 1, 1, 2, out of 30 in each case. Therefore we had an average 97% success on the test sets. In the case of the smaller training sets and bigger test sets an average 95% success rate was obtained on the test sets.

Specification of mixes are within standard bands. We have now started to investigate necessary specification band coverage. Not surprisingly our initial results show that good coverage is needed. In other words, the feature space of training set examplars needs good coverage relative to the test set cases.

4 Conclusions

We have tested a new image content characterization approach, with excellent results on images of aggregates containing different object sizes and morpholo-

gies. Our algorithms are computationally inexpensive, and scalable. In our experimental evaluation, we have found these algorithms to be robust and stable.

From the point of view of operational use in the difficult conditions of the construction industry, we note that the algorithmic robustness and stability leaves just one area where care and attention will be required in practice: viz., the operational camera and lighting environment.

References

[1] P. Hobeda. Krossningens betydelse på stenkvalitet, starskilt med avseende på kornform. Literaturstudie Nr. 050001, Statnes vag-och Trafikinstitut, VTI, Linköping, Sweden, 1988.

[2] V. Reinhardt. Schlagfester Splitt 8–11mm oder stabiler Asphaltbeton 0–12mm. *Bitumen, Teere, Asphalte, Peche und verwandte Stoffe*, 1969. Nr. 11.

[3] H. Choi and R.G. Baraniuk. Multiscale image segmentation using wavelet-domain hidden Markov models. *IEEE Transactions on Image Processing*, 10:1309–1321, 2001.

[4] J.L. Starck, F. Murtagh, and A. Bijaoui. *Image Processing and Data Analysis: The Multiscale Approach*. Cambridge University Press, 1998.

[5] N. Fatemi-Ghomi. *Performance Measures for Wavelet-Based Segmentation Algorithms*. PhD thesis, Surrey University, 1997.

[6] S.G. Mallat. A theory of multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11:674–693, 1989.

[7] S. Livens, P. Scheunders, G. Van de Wouwer, D. Van Dyck, H. Smets, J. Winkelmans, and W. Bogaerts. A texture analysis approach to corrosion image classification. *Microscopy, Microanalysis, Microstructures*, 7:1–10, 1996.

[8] M. Unser. Texture classification and segmentation using wavelet frames. *IEEE Transactions on Image Processing*, 4:1549–1560, 1995.

[9] P. Scheunders, S. Livens, G. Van de Wouwer, P. Vautrot, and D. Van Dyck. Wavelet-based texture analysis. *International Journal of Computer Science and Information Management*, 1(2):22–34, 1998.

[10] F. Murtagh, Xiaoyu Qiao, D. Crookes, P. Walsh, P.A.M. Basheer, and A. Long. Machine vision methods for the grading of crushed aggregate. *Machine Vision and Applications*, 2003. submitted.

[11] F. Murtagh and A Heck. *Multivariate Data Analysis*. Kluwer, 1987.

[12] J.M. Romeder. *Méthodes et Programmes d'Analyse Discriminante*. Dunod, 1973.

[13] D.J. Hand. Academic obsessions and classification realities: ignoring practicalities in supervised classification. In *Classification, Clustering, and Data Mining Applications*, pages 209–232. Springer, 2004.

A MULTISCALE APPROACH TO SHOT CHANGE DETECTION

Hugh Denman
Electronic Engineering Department,
Trinity College Dublin
email: hdenman@cantab.net

Anil Kokaram
Electronic Engineering Department
Trinity College Dublin
email: anil.kokaram@tcd.ie

12 May 2004

Abstract

We describe a multistage approach to shot cut detection based on image descriptor differencing at a coarse temporal scale, followed by identification of shot cuts and fades at frame-level accuracy based on explicit modelling of image data evolution during fades.
Keywords*: Video parsing, cut detection, media discontinuity*

1 Introduction

Effective shot change detection for video sequences is an essential prerequisite for the automated and computer-assisted manipulation of digital visual media.

We propose here a framework for shot change detection based on analysis of image descriptors, such as luminance histograms. Our implementation focuses on shot change detection as the first stage of media processing for the addition of footage to a large media library, so we are interested in exploiting as far as possible image descriptors that will be of use generally in the media library. These descriptors can then be computed in a separate feature extraction pass and stored alongside the media files. Thus, that portion of the execution time of this algorithm that involves feature extraction should properly be amortized over the lifetime of the media asset in the library, taking into account use of the features in subsequent stages.

In the subsequent sections of this paper, we will first describe a general method for data fusion across multiple media descriptors for discontinuity detection, and an application of this method to shot change detection for video sequences. We then outline a new approach to dissolve modelling and describe its effectiveness. We wish to note that in the final stages of the preparation of this paper, we discovered work similar in approach to the techniques described in the next section, published by Taskiran and Delp in [5].

2 Change detection in frame descriptors

Consider a difference features n between two frames, and let $P(T)$ be the probability that the two frames span a shot change boundary. We have

$$P(T|n) = \frac{P(n|T)P(T)}{P(n)}$$

2.1 Prior probabilities

In order that new frame features can be added to the shot detection framework, and to make the system as generally applicable as possible, we will avoid using specific prior distributions. Thus, we assign a uniform probability distribution to the likelihood, $P(n|T)$. This is a working assumption which reflects that across shot transitions, a feature change may take on any value - for example, global motion estimation will be degenerate across hugely disimilar frames, and colour histograms may be arbitrarily different or similar. This likelihood could alternatively be computed for a particular feature using an existing corpus for which the ground truth is known, but there is then an attendant risk of specialising to the characteristics of that corpus.

Many researchers have assigned prior distributions to T parameterised on shot length, introducing a bias against very short shots, but shot length distribution is a characteristic of genre, and in music videos, for example, shots may be shorter than one half second. Furthermore, glitches and special effects in video may result in shots of only one frame long, e.g. where significant dropout has occured, or in a faux 'subliminal image' effect common in music videos. We therefore also assign a uniform probability to $P(T)$, to reflect that shots may conceivably be of any positive length. Any more informative prior will necessarily be genre-specific (and could easily be incorporated where consideration is restricted to a specific genre).

We are then left with the problem of computing the prior for the feature in question, $P(n)$. Again, this can be explicitly evaluated by off-line analysis of a corpus, but again we feel that greater generality is achieved by computing this distribution from the data itself. In this implementation, we assume that the distribution will be approximately normal within a shot for any given difference feature, and that the value across a shot transition will be a large outlier of the normal distribution. Then, evaluation of whether a given value is an outlier can proceed based on two windows, one to either side of the point under consideration. As estimation of the parameters for a normal distribution is sensitive to outliers, we cannot include the present point in the window, and would ideally exclude previously identified shot cuts as well.

The principal parameter to be determined is the appropriate window size for estimation of the statistics, and it is here that our prior conception of shot length must be taken into account. At present, we use a 60 frame window for frame-to-frame metrics, and aindow size of 6 samples for metrics spanning 10 frames. In future, some form of adaptive window sizing procedure, alongside a more sophisticated estimation process for the metric statistics inside the window, will be incorporated.

2.2 Video Features

The present implementation makes use of three principal frame-level features. The first is an estimate of the translation global motion, computed using integral projection based on an image model of

$$I_n(\mathbf{x}) = \mathbf{I_{n-1}}(\mathbf{x} + \mathbf{d})$$

where $\mathbf{d}$ is the global motion, i.e. not varying with pixel site. This measure can be used directly as a frame-to-frame difference, as generally large estimated displacements correspond to large frame differences. More sophisticated global estimation techniques can be used, as, for example, in the paper by Kokaram [4].

Histograms have also been recognised as suitable for shot change detection, for example as described by Han *et al* [3]. We employ here a the bin-to-bin histogram difference. Each frame of the sequence is converted into the L, U, V colourspace, and a 101-bin histogram is computed of the L plane:

$$h(i) = \sum_{L(\mathbf{x})=\mathbf{i}} 1$$

This full histogram h is then downsampled to a ten bin histogram H, and our difference measure between frames is the sum of the absolut bin-to-bin differences of their downsampled histograms:

$$D_{n_1,n_2} = \sum_{i=1:10} |H(i)_{n_1} - H(i)_{n_2}|$$

The final feature used is the frame-to-frame edge moment differential. An edge map of each of the two frames to be compared is found, using the Canny edge detector. These edge maps are then dilated using a five-by-five disk-shaped structuring element, to improve robustness under motion. We denote this dilated edge map E, taking on values $E(\mathbf{x}) = \mathbf{1}$ if there is an edge at site $\mathbf{x}$ and zero otherwise. The second order moment M of the dilated edge map is then found, where

$$M = \sum_{\mathbf{x}} |\mathbf{x}|\mathbf{E}(\mathbf{x})$$

This second order moment is strongly correlated with the distribution of edges in the image. Thus, the frame-to-frame difference of this moment is an indicator of the difference between frames. A variety of other edge-related features can be used for shot detection and characterisation, for example the Hough transform [2] and a disappearing edge count [1].

The global motion and histogram differences are computed off line, and used for first-pass cut detection: if any difference value exceeds 50 standard deviations, the corresponding frame is immediately assumed to mark the start of a new shot. The standard deviation value used is the lesser of two calculated from windows to either side of the frame under consideration; this results in more stable sequence statistics being automatically selected. The use of such a locally estimated measure is greatly preferable to a prior fixed threshold value. For example, a shot of a single frame in length can be detected, and a shot transition between an ordinary shot and a shot consisting of a succession of unrelated frames can be detected, but a shot consisting of a succession of unrelated frames is not artificially partitioned.

Using the assumption that each difference feature is independent of the others, we can also combine local deviations to find more subtle shot cuts. Adding local deviations is conceptually equivalent to multiplying and scaling the associated probabilities. Where a combined local deviation exceeds 50, we flag a shot transition.

Local deviations between 10 and 50 in any single difference feature we consider to be possible shot cuts, for further examination. At present the only subsequent feature in use is the frame-to-frame difference of the second order moment of the dilated edge distribution. We evaluate this difference vector around the possible shot cut and compute local deviations as before. These local deviations are added to those previously computed using the other features, and if the sum local deviation exceeds 50, we assume that a shot cut has been detected.

This process can be augmented naturally to add in more sophisticated frame difference techniques until the confidence (local deviation) at each frame has move outside the thresholds of uncertainty.

The framework as developed here uses differencing between adjacent frames. We expect that gradual shot transitions can be more easily detected at a coarse temporal scale. In the following section, we outline how a possible shot transition region is examined to see whether it is likely to be a fade.

3 Fades

Fades, also known as dissolves, are a common transition in many video genres, including motion pictures, sports footage, and music videos. While analysis of frame features at a coarse temporal scale is generally sufficient to localise fades and other gradual shot transitions, this method by itself will result in very low precision, as video regions with high motion content will also be found. Some researchers have used edge information to analyse possible fade regions, but this is a computationally expensive approach, especially as dilation of edge maps is crucial for robsutness to motion. We introduce here an efficient

scheme for modelling fades in which possible dissolve regions are characterised by an *alpha curve*, where alpha is a parameter varying from 1 to 0 as the fade progresses. Examination of this curve then informs classification of the video region as being a fade, or otherwise.

3.1 Fade model

Our model assumes that a frame occuring during a fade is made up of a linear combination of two *template frames*, designated I_{T_0} and I_{T_1}, at positions preceding and succeeding the fade region. The image predicted by this model, for a given crossfade strength α, is designated $I_{M(\alpha)}$, and calulated by:

$$I_{M(\alpha)} = \alpha I_{T_0} + (1 - \alpha) I_{T_1}$$

The likelihood of a given value of alpha is proportional to the agreement between the image predicted by the model and the observed data, which is the image at time t, designated I_t. Specifically,

$$p(\alpha | I_t) \propto \exp(-\sum_{\mathbf{x}} [(I_t(\mathbf{x}) - \mathbf{I_{M(\alpha)}(x)})^{\mathbf{2}}])$$

For a given image, we can estimate the MAP value of alpha by differentiation with respect to alpha. It transpires that the optimal value is given by

$$\alpha_{opt} = \frac{\sum I_t \nabla_{T_0,T_1} - \sum I_{T_1} \nabla_{T_0,T_1}}{\sum \nabla^2_{T_0,T_1}}$$

where ∇_{T0,T_1} is simply the difference image $I_{T_0} - I_{T_1}$.

3.2 Global motion

The model as presented makes no account of the motion content of the image sequence, which will result in probable failure to accurately estimate alpha in dissolves that occur between sequences with significant motion. As a first step to improving robustness in this instance, we introduce global motion compensation. When computing α for frame I_t, we first apply cumulative global motion parameters to frame I_{T_0} and inverse parameters to frame I_{T_1}, to compensate each template to time t. After this compensation, only a partial region of each template frame will contain valid data, and estimation of α is performed on the overlapping area of the valid regions. Naturally, this process introduces a dependency on the accuracy of the global motion estimator; the results described herein were based on an implentation using a fast, projection-based estimator, estimating translation motion only, and difficulties can be expected in sequences featuring fast zooms. Compounding this disadvantage is that we require global motion estimates in precisely the region where they are most difficult to compute, viz. within the dissolve itself; we intend to investigate a more sophisticated implementation which will extrapolate global motion parameters computed prior to the suspected dissolve region where possible. A further difficulty is that analysis of regions undergoing rapid global motion may be impossible, where the intersection of the regions valid after global motion compensation is the null set.

3.3 Local motion

Local motion in the dissolve region will introduce a large, localised discrepancy between the template image and the current frame. These discrepancies will then confound the alpha estimation process. To account for this, we employ an iterative reweighting scheme based on a Cauchy weighting function. In our first estimate, the weight at every site is 1. We then examine the residual image, $I_e = I_{M(\alpha)} - I_t$, and update the weight at each site according to:

$$w(\mathbf{x}) = \frac{\mathbf{1}}{(\mathbf{1} + \mathbf{r^2})}, \quad \mathbf{r} = \frac{\mathbf{I_e(x)}}{(\mathbf{2.385})(\mathbf{s})\sqrt{(\mathbf{1} - \mathbf{h})}}$$

In the above formula, the residuals $I_e(\mathbf{x})$ are being scaled to take into account the leverage of the point h (distance from data centroid), and s is related to the median absolute distance of the residuals from their median (a measure of the overall spread of the data).

This process is repeated until the number of residuals exceeding a certain threshold is zero, or the sum of the residuals begins to increase, or the number of iterations exceeds a certain limit. None of these halting conditions is entirely satisfactory, as correct estimation of alpha may indeed involve increasing the number of pixels assigned to local motion after some iterations, and choice of the appropriate thresholds is difficult (currently alpha estimation is discontinued if less than 2% of the image has an error of more than 20 graylevels). While the present, somewhat ad-hoc approach does produce satisfactory results, it is frequently apparent that the algorithm is performing more iterations than necessary.

3.4 Fade curve analysis

Having calculated the alpha values for each frame in the region of interest, we then examine the resulting curve to see whether it has the characteristics of a fade. We have adopted a simple approach in which the alpha curve is partitioned into three sets of adjacent values, and fit a line to each partition. We iterate over every possible choice of two changepoints in the alpha curve, and lines are fitted to each of the three resulting segments. A confidence measure is associated with each line, based on the mean squared distance from each point in the segement to the line. The partition that gives the highest average confidence over the three fitted lines is then selected. As the number of points in an alpha curve is only of the order of forty, this exhaustive search strategy is by no means computationally prohibitive.

Having found the lines of best fit, the slope of each of the three lines can then be examined to determine if a fade has occured: we expect a flat first line, followed by a line with a negative slope of moderate magnitude (corresponding to the transition region), followed by a final flat region. This method generally determines the start and endpoints of the fade to an accuracy of within +- one frame, depending on the motion characteristics of the shots involved.

We also examine the alpha curve for sudden discontinuities; if successive values differ by more than 0.7, we assume that a shot cut has occured.

3.5 Examples

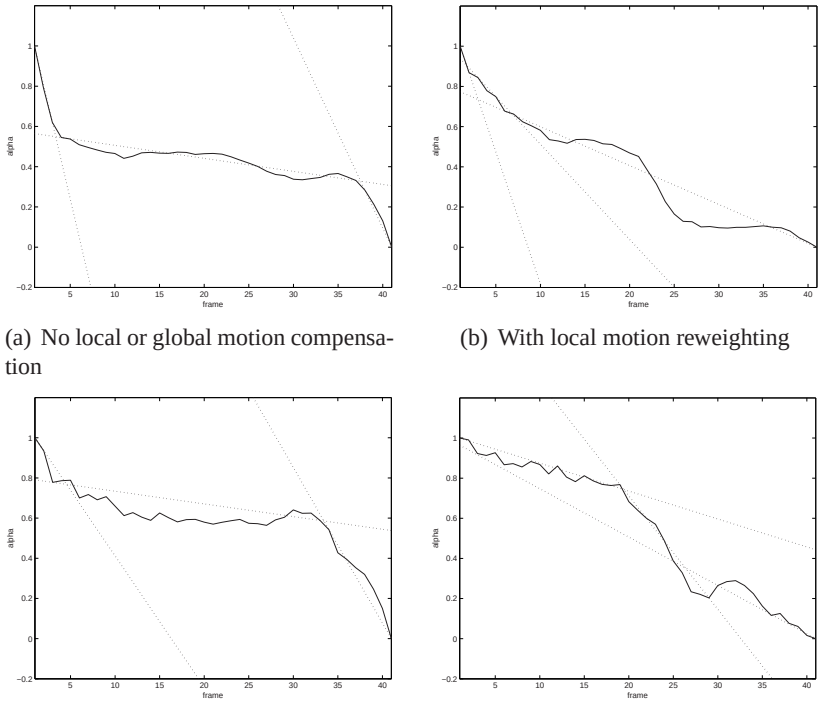
Figure 1 shows a fast dissolve in a cricket sequence, and figure 2 shows the extracted alpha curves. It can be seen that without either global motion compensation or reweighting local motion regions, fade detection has failed. The fade is detected, however, when both compensation strategies are employed.

We encounter again the issue of selection of an appropriate window size; where the window is too small, the flat areas corresponding to the unmixed shots will not be readily apparent, whereas with an overlarge window, cumulative motion effects can be expected to degrade the quality of the curve greatly. Furthermore, the slower the dissolve, the larger the window will be necessary. At present, a fixed window size of forty frames is employed.

4 Results

We have applied the shot transition detection framework described in section 2 to a variety of test sequences, though at the time of description the algorithm is under continual refinement and elaboration. The first is a simple 'proof of concept' sequence, referred to here as *News*, with 645 frames, 5 cuts, and 4 frames. For this sequence, we attempt to detect frame fades across regions even if the regions are already known to contain a cut, and discard cuts that are subsequently found to be within fade regions. Here we achieve 100% recall for both cuts and fades, and 100% precision for fades, with all fade start and end points detected to within one frame of the observed values. However, two spurious shot cuts are detected, bringing the cut detection precision down to 83.3%. These spurious shots correspond to sudden small

Figure 1: A cricket sequence containing fast local and global motion on both sides of a fast dissolve. The frames shown correspond with offsets 5, 10, 15, 20, 25, and 30 in figure 2.



(a) No local or global motion compensation

(b) With local motion reweighting

(c) With global motion compensation

(d) With global motion compensation and residual reweighting (fade region identified, between frames 19 and 30)

Figure 2: Alpha estimation across a fast dissolve with significant global and local motion. The dissolve starts at frame 19 and ends at frame 27. The dotted lines show the region partitioning.

differences from image to image in regions that are otherwise perfectly stable, so in a sense this kind of failure is intrinsic to the algorithm as presented. However, these false alarms could easily and cheaply be suppressed by imposing a minimum on the norm of the frame difference image, to insure that cuts are only detected when changes are over a significant region of the image.

We also analysed a 14,000 frame video of cricket play. This sequence contains 62 cuts and 20 fades. It is characterised by much fast global motion, including fast zooms, and quick crossfades, typically over 6 to 10 frames. Here we achieve 92% recall and 86% precision in cut detection using the media discontinuity scheme of section 2. Fades are identified with 70% recall and 80% precision. When we take into account the detection of cuts via discontinuities in the alpha curve, we score 94% recall and 86% precision.

The accuracy of the results in analysing the cricket sequence suffer due to the motion characteristics of the sequence. We expect that these results can be improved upon through improved global motion estimation.

References

[1] Paul Browne, Alan F Smeaton, Noel Murphy, Noel O'Connor, Sean Marlow, and Catherine Berrut. Evaluating and combining digital video shot boundary detection algorithms. In *Irish Machine Vision and Image Processing Conference*, 2000.

[2] Hugh Denman, Niall Rea, and Anil Kokaram. Content-based analysis for video from snooker broadcasts. *Journal of Computer Vision and Image Understanding*, 92:141–306, 2003.

[3] Seung-Hoon Han, Kuk-Jin Yoon, and In-So Kweon. A new technique for shot detection and key frames selection in histogram space. *Image Processing and Image Understanding*, 2000.

[4] Anil Kokaram and Perrine Delacourt. A new global estimation algorithm and its application to retrieval in sport events. In *IEEE International Workshop on Multimedia Signal Processing*, Cannes, France, October 2001.

[5] C. Taskiran and E. J. Delp. Video scene change detection using the generalized sequence trace. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Seattle, WA, May 1998.

BUILDING SHAPE AND TEXTURE MODELS OF DIATOMS FOR ANALYSIS AND SYNTHESIS OF DRAWINGS AND IDENTIFICATION

Y.Hicks, D.Marshall, R.R.Martin, P.L.Rosin
Cardiff School of Computer Science
Cardiff University, UK
email: y.a.hicks, dave, ralph, paul.rosin @cs.cf.ac.uk

S.Droop, D.G.Mann
Royal Botanic Garden Edinburgh
Edinburgh, UK
email: D.Mann@rbge.org.uk

Abstract

We describe tools for automatic identification of diatoms by comparing their photographs with other photographs and drawings, via a model. Identification of diatoms, *i.e.* assigning a new specimen to one of the known species, has applications in many disciplines, including ecology, paleoecology and forensic science. The model we build represents life cycle and natural variation of both external shape and internal texture over multiple species and is based on *principal curves*. The model is also suitable for automatically producing drawings of diatoms at any stage of their life cycle development. Similar drawings are traditionally used for diatom identification, and encapsulate visually salient diatom features. In this article we describe the methods used to analyse photographs and drawings, present our model of diatom shape and texture variation, and illustrate our approach with a collection of drawings synthesised from our model and derived from example photographs. Finally, we present the results of identification experiments using photographs and drawings.
Keywords*: Classification, automatic drawing synthesis, principal curves, diatoms.*

1 Introduction

Diatoms are unicellular algae with a highly ornate silica shell around each specimen. The shell contains two larger elements called valves, one on either side of the cell, which bear species-specific patterns. Identification of diatoms, i.e. assigning a new specimen to one of the known species, has applications in many disciplines, including ecology, paleoecology and forensic science. Specimens are usually identified by highly trained specialists by considering diatom morphological characteristics, including shape and texture, and comparing them to photographs and drawings of previously identified specimens. This task is challenging due to a huge number of diatom species, similarities between species and life cycle related changes in shape and texture.

Recently there have been various efforts in quantitative analysis of diatom shape variation [2, 6, 7]. A system for automatic identification of diatom specimens in photographs, based on the silica shell shape, size and pattern characteristics, was developed in the ADIAC project [1]. We seek to extend such capabilities through the inclusion of biological drawings. There is a wealth of diatom specimen drawings in the biological literature accumulated over many years. The drawings contain mainly the salient information required for identification and thus may serve as models of each species. Hence, including digitised drawings in the system and providing the ability to compare photographs and drawings has significant benefits for the biological community.

A different issue is automatic production of diatom drawings. Diatom type specimens are traditionally defined in the taxonomic literature using drawings and, although photographs have been used much more

often in the last 20 years, there remain a significant number of genera for which drawings are more appropriate. Automating the production of drawings would be especially useful as it is a time consuming and difficult task (Figure 1).

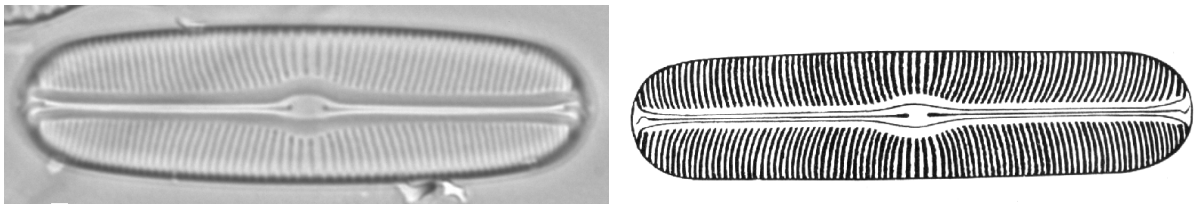


Figure 1: A photograph of a diatom valve and a drawing of a similar valve by a biologist.

In recent years, the problem of finding a mapping between photographs and drawings came to the attention of computer vision and computer graphics communities. For example, A.Hertzmann *et al.* [4] learn the mapping through correspondence of low-level pixel statistics in a drawing and a photograph. However, such approaches are unsuitable for the task at hand due to their requirement for an exact match between the drawings and the photographs, which is usually not available in biological materials.

Our approach is to transform the high-dimensional image space of both photographs and drawings into a lower-dimensional space where only relevant features are represented. We then use this space for the comparison of different specimens as well as for automatic production of drawings.

In our research we go further by not only developing a system capable of identifying new diatom specimens, but also producing a model describing life cycle related variation in the shape and pattern of multiple diatom species and suitable for synthesising example drawings of the species.

In this article we present methods for analysing diatom shape and texture, produce a model representing variation of shape and texture in multiple diatom species, and illustrate our approach with a number of drawings generated automatically from the model and original photographs. We finish with presenting the results of identification experiments.

2 External contour analysis and synthesis

Many diatom valves are sufficiently flat to give a repeatable view in all photographs. Traditionally, when analysing diatom shape, diatomists performed 2D contour analysis in this view. However, due to various reasons it is not an easy task to extract the contours from photographs automatically. Overlapping debris and diffraction effects may make it hard to locate the contour. In the course of ADIAC [1], several sophisticated methods for contour extraction have been developed. In this article we use the extracted contours provided to us from the ADIAC project.

To represent diatom contours in a compact way we use Fourier descriptors as we explain in [5]. Thus each diatom contour is represented with a 200 element vector consisting of 100 amplitude values and 100 corresponding phase angles obtained from Fourier descriptors. It is possible to reconstruct the shape of the diatom from these values, as we do in [5].

3 Texture analysis

Our goal here is to analyse the diatom silica shell patterns and represent them in a way suitable for synthesis. The variety of patterns occurring in diatoms is very great. A complete system would need to perform a series of tests to detect the type of pattern and then choose a suitable set of analytical tools to measure the values of appropriate pattern parameters. In the initial system reported in this article we restricted our approach to the analysis of pennate diatom species with striae patterns on their shells; most diatoms are of this kind. The striae are transverse lines of pores between the silica ribs coming out from

the diatom's long axes (raphe-sternum or sternum). The patterns formed by the striae are characterised by frequency and orientation. For simplicity, we model striae as straight, which is a good approximation in the majority of cases considered.

In ADIAC [1], Gabor wavelets were used to detect the frequency and orientation of the striae and to segment the diatom shells. However, unless the pattern orientation and frequency are known beforehand, or their range is very limited, a large bank of filters needs to be applied. In ADIAC, 28 filters were used, covering a range of 4 different orientations and 7 different frequencies.

Fourier analysis provides a more general approach to detecting the frequency and orientation of the striae patterns, and is more suitable for the purpose given the range of possible frequencies and orientations, thus it is our chosen tool. We perform an FFT within a sliding window of size 48 x 48 at each pixel inside the diatom contour. This size ensures that at least 3 striae fit inside the window (at our image resolution) for robust detection of pattern orientation and frequency.

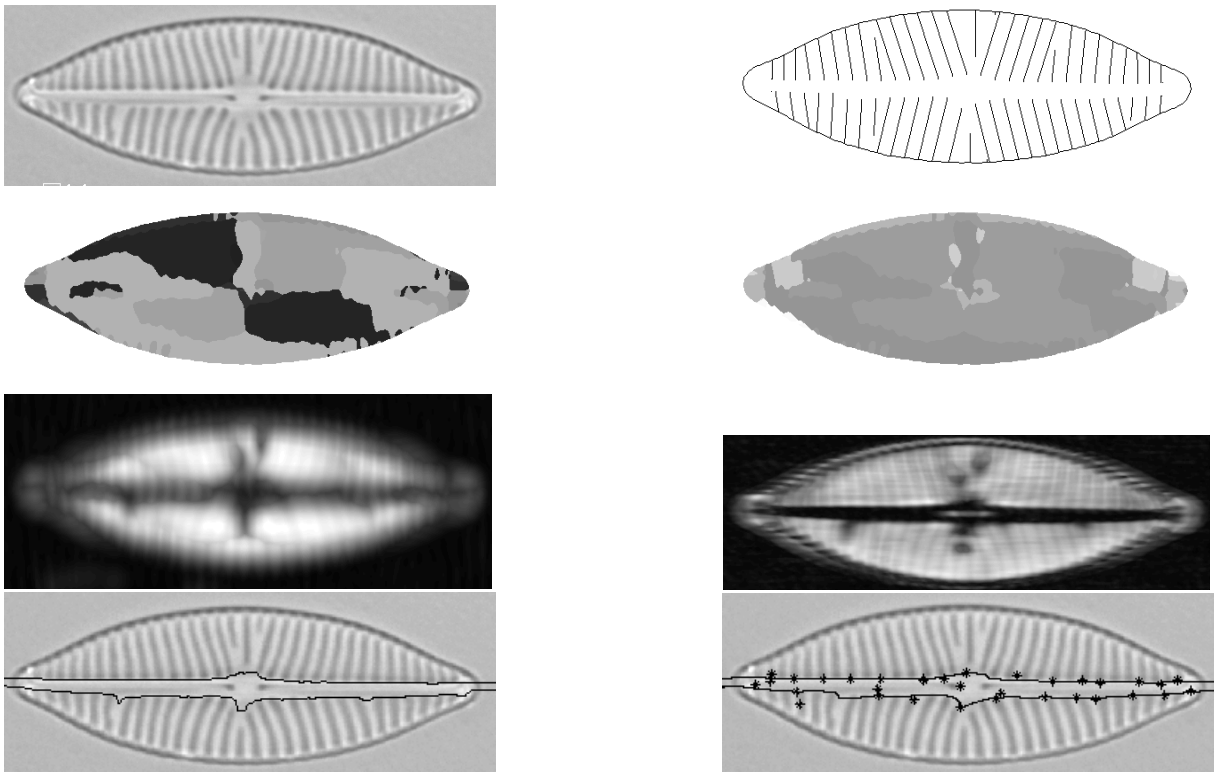


Figure 2: From left to right top down: a photograph of a diatom, synthesised drawing, orientation map, frequency map, energy map (using 48x48 window), energy map (using 2x48 window), central part borders, fitted splines together with control points.

For each window we find the energy values corresponding to the Fourier coefficients. Then we set to zero the DC Fourier component as well as the values corresponding to the frequencies of 1 and 1/2, as we expect at least three striae in each window. We also set to zero the values corresponding to almost horizontal orientations, as we do not expect to find striae in such orientations. Finally, we find the maximum among the remaining FFT energy values to give the orientation and frequency. Thus we obtain three maps from the run of the FFT. The first one contains the striae orientation values for each pixel inside the diatom contour, the second contains the striae frequency for each pixel inside the diatom contour, and the third map contains energy values (FFT amplitude) for each pixel inside the diatom contour (Figure 2). We use these maps at a later stage to find the average striae orientation and frequency values in different areas of a diatom.

Apart from knowing the striae orientation and frequency, we also need to detect the borders of the

central area of the diatom with no striae (the sternum or raphe-sternum). The energy map gives us some idea of where there are striae. However, its borders are hard to pinpoint due to the size of the sliding FFT window. We perform a second windowed FFT on the whole image, this time using a window of size 2 x 48, finding the largest peaks in the Fourier domain in the same way as before. However, this time we are only interested in the energy map. We find the vertical borders of the central area by traversing the energy values in each column of the map up and down from the centre, looking for the first value above the threshold, which we set at three quarters of the average energy value over the whole energy map. Finally, we fit a set of cubic splines into the top and bottom borders, thus describing each border with 19 spline control points.

To obtain parameter values characterising the texture, we split the inside of the diatom contour into 12 parts, 6 above the sternum and 6 below. The borders of the parts are determined by splitting the curves approximating the top and the bottom borders of the central diatom area into equal lengths. We find the average orientation and frequency inside each of these parts as the weighted average of all orientation and frequency values, where the weights are the corresponding energy values.

The internal pattern of each diatom is described using a 100 element vector, where 76 elements are the coordinates of the 38 control points and another 24 values are orientation and frequency values.

In conclusion, we would like to point out that the method presented above is suitable for the analysis of diatoms represented in both photographic and drawing form.

4 Texture synthesis

To draw the internal structure of the diatom, we draw lines representing the striae between the external contour and the sternum borders. This is done using the average orientation and frequency values in several areas inside the diatom contour.

To model or mimic actual valves satisfactorily, the requirements for the generated striae are that they should have the appropriate orientation and frequency values, and should be continuous across each area of different orientation and frequency. For example, if two striae diverge too far from each other, another stria should appear in between, or if they converge, eventually they should either merge or one of them should disappear.

In our synthesis algorithm we attempt to follow the way it is believed the diatom shell is formed naturally [9]. The striae are formed gradually, the ones near the centre of the diatom start growing first and may be partially completed by the time the striae further away from the centre start forming. We attempt to model this process in our iterative synthesis algorithm outlined below.

1. Starting at the centre of the top sternum border, going out towards the right end of the diatom add one more pixel to the length of all existing striae, keeping all striae of orientations appropriate to the areas of the diatom they are located in, checking that they have not reached the diatom contour yet and that they are not too close (less than half of the striae spacing appropriate to the corresponding area of diatom) or too far (more than twice the striae spacing appropriate to the corresponding area of diatom) from the nearest stria on the left. The threshold values for the striae spacing were derived experimentally to imitate the underlying natural processes.

2. If the stria on the left is too close to the current stria, or the current stria has reached the external contour, then the current stria becomes “completed”, and in that case no more pixels are added to it in the future.

3. If the stria on the left is too far away, then another stria is inserted between the two that have diverged too far.

4. After we have considered all existing striae on the right from the centre, and if we have not reached the contour of the diatom, we add one more stria to the right of the rightmost stria at the distance appropriate for the area.

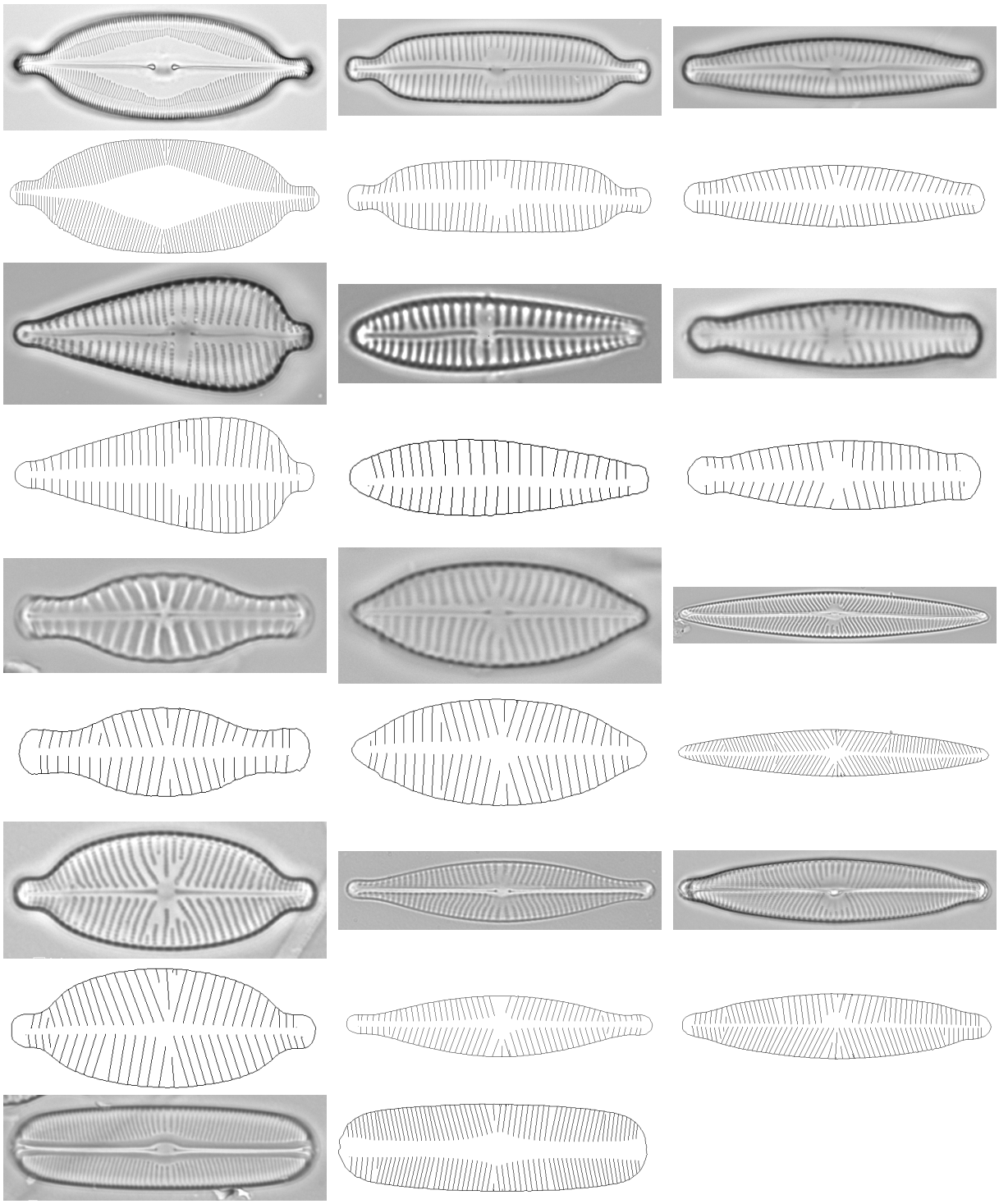


Figure 3: Photographs and drawings generated automatically from the photographs of 13 species. The species are in the following order: *Caloneis amphisbaena, Cymbella hybrida, Cymbella subaequalis, Gomphonema augur, Gomphonema minutum, Gomphonema species 1, Navicula capitata, Navicula menisculus, Navicula radiosa, Navicula constans, Navicula rhynchocephala, Navicula viridula, Sellaphora bacillum.* Please note that the original images are very high resolution and contain high frequency information which may not be adequately printed or displayed on some devices.

5. Repeat all the above steps until all the striae are “completed”.

6. Repeat all the above steps for the other three quarters of the diatom starting at the centre and going out towards the ends of the diatom along the top or bottom of the sternum.

5 A model of shape and texture

Previously [5], we presented a model of shape variation during the life cycle of several diatom species. The model was based on a collection of principal curves, where each curve modelled the growth trajectory of a diatom species. Individual shape variations within species are defined in the dimensions orthogonal to the principal curve.

Principal curves were first defined by Hastie and Stuetzle [3]. Intuitively, a principal curve is a smooth curve passing through the “middle” of a data distribution. Principal curves are estimated recursively for a given data set. In practice the curves are approximated with a number of knots and linear segments connecting them.

We have now extended our earlier model based on diatom contours to represent diatom texture as well. Prior to modelling the diatom shape and texture data (the set of parameter values described in Sections 2 and 3, for all specimens from all species) we normalise the data to have zero mean and standard deviation of one. We find main modes of variation in the data of all species through PCA. Then we model the life cycle shape and internal texture variation in each species using a principal curve going through the middle of the corresponding data set. This approach allows us to extend the model to include a new species easily, which is more difficult for a decision-based diatom identification method [1].

6 Experiments

6.1 Diatom analysis and automatic drawing generation

Our test data includes over 300 photographs of 13 different species, namely, *Navicula constans, Sellaphora bacillum, Navicula rhynchocephala, Gomphonema augur, Cymbella hybrida, Cymbella subaequalis, Navicula capitata, Caloneis amphisbaena, Navicula menisculus, Gomphonema minutum, Gomphonema species 1, Navicula radiosa, Navicula viridula* (examples are shown in Figure 3). We used these to produce drawings directly from each photograph. The quality of the produced drawings degraded gracefully with decreasing quality of the original photographs. Please note, that due to the reduced size of the photographs, it may be difficult to see the striae orientation and frequency of *Caloneis amphisbaena* in Figure 3.

6.2 Building a model and reconstructing drawings from the model

For this experiment, we selected the best quality photographs described in the previous section to make sure that the models produced were reliable and did not contain any errors from the analysis stage. The number of the specimens in each species set ranged from 5 for *Gomphonema augur* and *Navicula radiosa* to 20 for *Gomphonema minutum*, giving a total of 178 specimens. Prior to using principal curves to model the diatom shape data, we normalised the data and then found the main modes of variation in the data set of all species through PCA, as described earlier.

We built a model of diatom shape, length and internal texture variation over the life cycles of the above 13 species by fitting an individual principal curve to each of the available 13 data sets (Figure 6.2).

In Figure 5 we synthesise the drawings of diatoms from the principal curve nodes depicting the diatoms at different stages in their life cycle. Note that there may be no corresponding photograph for that stage – here the drawings are generated solely from the model, not directly from a photograph.

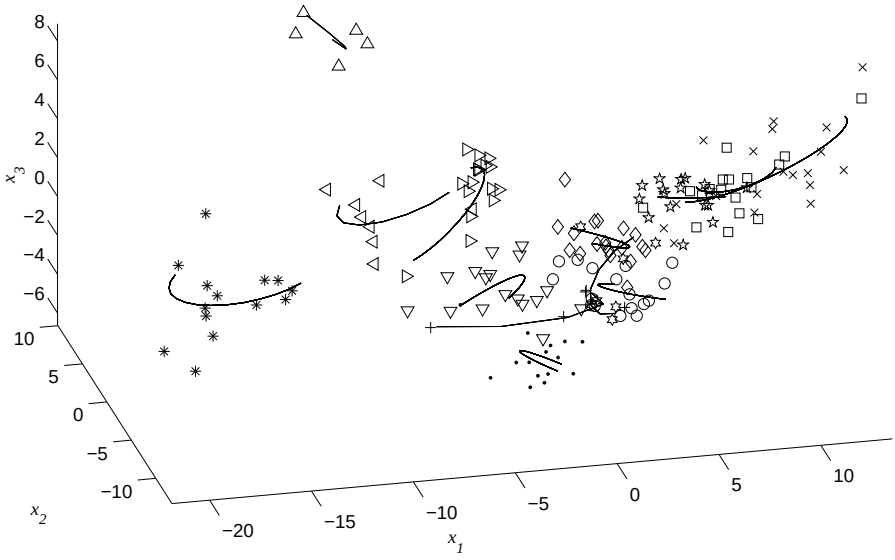


Figure 4: Principal curves and the data used for their training, projected into the space of three largest eigenvectors. Different species are represented with different symbols.

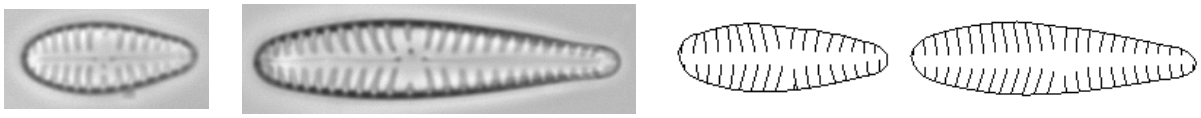


Figure 5: Some of the *Gomphonema minutum* photographs used for training a principal curve and drawings generated automatically from the principal curve at other stages of the life cycle.

6.3 Identifying diatoms from photographs and drawings using our model

The first experiment consisted of identifying diatoms whose images were not used for constructing the model. For this experiment we used the standard "leave one out" approach, where the model was trained on all the specimens apart from one and the remaining specimen was identified using the trained model. We repeated the experiment omitting each specimen out of the total 178 used in Section 6.2. We compared the identification accuracy between a model trained on the diatom shape and length data, a model trained on the texture data only, and a model trained on shape, texture and length data.

The error rate when using the external contour and length data was 19.66%. For the texture data only, the error rate was 6.18%. Using shape, texture and length data the error rate decreased to 3.37%, which is a significant improvement to using either contour or texture data alone, and is similar to the error rate achieved in the ADIAC project in similar experiments. However, the data set used in the ADIAC included a larger number of species, some of which had non-striae patterns.

We used several other standard classification methods on the same data set in leave-one-out experiments for comparison with our model. Using a support vector machine (SVM), developed by Ryan Rifkin at MIT's Center for Biological and Computational Learning with a linear kernel gave us a classification error rate of 6.18% on the normalised data, and a 19.1% error rate was achieved using OC1 decision tree approach [8] on the raw data without prior normalisation.

To identify a diatom in a drawing we used the same procedure as for the photographs. We obtained parameter values by image analysis of seven drawings of seven different diatom species also represented in the above photograph set. Four drawings were identified correctly. In the two out of three misidentified

drawings, the striae frequency was found to be double the real value due to the artistic technique used in the drawings. After we manually corrected the frequency values for these drawings, one more was identified correctly.

7 Evaluation and future work

We have presented a means of modelling shape, length and texture variation in multiple diatom species. The model is built from data automatically extracted from photographs, and is based on diatom features which are present in both photographs and drawings and used for diatom identification.

The model is suitable for identification of previously unseen diatoms represented in photographic or drawing form. It is also suitable for reconstructing drawings of diatoms at any stages of their life cycles, including those not explicitly represented in the original training set.

We have presented drawings produced by our methods and the results of identification experiments. Identification experiments achieved a similar accuracy to those resulting from the ADIAC project; however, ADIAC data set was larger and included some diatoms with non-striae patterns.

Currently biologists are working on applying the system presented to classification problems in a biological context (taxonomy).

8 Acknowledgments

This project is funded by the BBSRC/EPSRC under the Bioinformatics Programme, grant 754/BIO14261. In our experiments we used Chang's implementation of Probabilistic Principal Curves as a part of LANS Pattern Recognition Toolbox, *http://www.lans.ece.utexas.edu/ kuiyu/.* The data set of diatom photographs, used in the project, was provided to us by the ADIAC partners.

References

[1] H. du Buf and M.M. Bayer (eds.). Automatic Diatom Identification. Vol. 51, Series in Machine Perception and Artificial Intelligence, World Scientific Publishing Co., Singapore, 2002.

[2] N. Goldman *et al.*. Quantitative analysis of shape variation in populations of *Surirella fastuosa.* Diatom Research, vol.5, pp.25–42, 1990.

[3] T. Hastie and W. Stuetzle. Principal Curves. Journal of the American Statistical Association, vol.84, issue 406, pp.502–516, June 1989.

[4] A. Hertzmann *et al.*. Image Analogies. SIGGRAPH'2001 Proceedings, pp.327–340, 2001.

[5] Y.A. Hicks *et al.*. Modelling life cycle related and individual shape variation in biological specimens. Proc. BMVC'2002, Sept 2-5, Cardiff, Wales, Volume 1, pp.323–332, 2002.

[6] Y. Hicks *et al.*.Automatic Landmarking for Building Biological Shape Models. Proc. ICIP 2002, Rochester, NY, USA Vol II, pp.801-804, 2002.

[7] D. Mou and E.F. Stourmer. Separating *Tabellaria* (Bacillariophyceae) Shape Groups Based on Fourier Descriptors. Journal of Phycology, vol.28, pp.386–395, 1992.

[8] S. Murthy *et al.*. System for Induction of Oblique Decision Trees. Journal of Artificial Intelligence Research, vol.2, pp.1–33, 1994.

[9] F.E. Round *et al.*. The diatoms. Biology and morphology of the genera. Cambridge University Press, 1990.

WAVELET BASED TEXTURE SYNTHESIS

Claire Gallagher and Anil Kokaram
Department of Electronic and Electrical Engineering
Trinity College
Dublin, Ireland.
email: gallaghc@mee.tcd.ie

Abstract

This paper presents a new algorithm for synthesising image texture. Texture synthesis is an important process in image post-production. The best previous approaches have used non-parametric methods for synthesising texture. Unfortunately, these methods generally suffer from high computational cost and difficulty in handling scale in the synthesis process. This paper introduces a new idea of using wavelet decomposition as a basis for non-parametric texture synthesis. The results show an order of magnitude improvement in computational speed and a better approximation of the dominant scale in the synthesised texture.
Keywords*: Texture Synthesis, Complex Wavelet Transform, Image Processing, Non-parametric Image Modeling.*

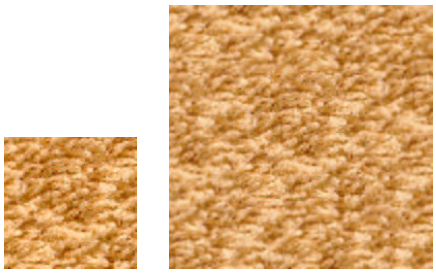


Figure 1: Texture synthesis: Given an example texture I_e as an input (left), the algorithm aims to reproduce new texture I_s (right).

1 Introduction

The problem of texture synthesis has been an active research topic in recent years [5, 4, 15, 10]. Given an example of texture as a small subimage, the idea is to create a much larger image by synthesising *more* texture. Figure 1 shows on the left a typical example image or "seed" of size 128×128 and on the right is the synthesised image of size 256×256 created by surrounding this "seed" with *new* texture. This kind of operation is often required in the post-production of digital images when a large area is to be covered with texture that *looks like* some smaller example. Picture editing often requires filling of missing information and texture synthesis processes like these can fill such holes with reasonable material.

The essential idea is to somehow estimate the p.d.f. of the image intensity $I(\mathbf{x})$, denoted by $P(I(\underline{\mathbf{x}}))$ at a pixel site $\underline{\mathbf{x}} = (i, j)$. The process of texture synthesis is then a matter of drawing a random sample from that distribution. What makes this difficult is estimating $P(I(\underline{\mathbf{x}}))$. Two different approaches have

emerged. Parametric techniques attempt to model $P(I(\underline{\mathbf{x}}))$ with some definable process. Heeger and Berger [6] analyse texture using histograms of filter responses at multiple scales and orientations. Portilla and Simoncelli [11] improve on this idea by matching pairwise statistics across different scales and orientations. Kokaram [10] uses an autoregressive model when synthesising texture. All of these methods work well on simple textures but fail for more structured textures [11]. Non-parametric approaches rather, attempt simply to *measure* the p.d.f. from the image. The visual quality of the generated textures will be influenced primarily by the accuracy of the model, while the efficiency of the sampling procedure will be directly related to the computational expense [15]. Because of the wide variability in image behavior non-parametric approaches have achieved by far the more visibly pleasing results [5, 15, 1, 14, 2].

Most of the non-parametric methods rely on an idea introduced by Efros and Leung in 1999 [5]. Their approach was based on empirical measurement of the p.d.f. of a pixel using neighbourhood similarity. This method assumes texture can be modeled by a Markov Random Field (MRF), i.e. the intensity value for a pixel given the intensities of its spatial neighbourhood is independent of the rest of the image. The p.d.f. $P(I(\underline{\mathbf{x}}))$ is then sampled and the newly assigned pixel is assigned to the synthesised image. This algorithm generates impressive results and works well on a large range of textures. However, computational cost is high because an entire search of the sample image is necessary for each of the pixels to be synthesised. In addition, the success of the algorithm is dependent on the correct choice of neighbourhood size. This user defined parameter controls the randomness of the texture to be generated.

Ashikhmin [1], Bornard [2] and Pei et al. [14] address the computational burden of the Efros algorithm by introducing coherent searching into the synthesis procedure. This speeds up the synthesis process by eliminating the need to search every possible neighbourhood in the sample image. Wei and Levoy [15] develop the algorithm further to include multi-resolution synthesis. They use Gaussian pyramids to represent the texture and transform a random noise sample to resemble the sample texture at different levels of the Guassian pyramid. This method works well on stochastic (random) textures but is not suitable for deterministic (structured) textures [4].

In order to explore the problems of scale and computational load associated with non-parametric methods, we have introduced the novel idea of using the complex wavelet transform as a basis for non-parametric texture synthesis. The introduction of the wavelet decomposition into the synthesis procedure has two advantages. Firstly, it facilitates the measurement of texture statistics at particular scales. Unlike previous methods, who use scale information as a control [15], we directly synthesise texture at these different scales. This allows us to exploit the dominant frequencies present in the texture image. The second advantage of our method is the reduction in computational load. By synthesising texture at coarser scales, the original information is represented by fewer pixels. Large features which were present at a fine scale, are now much smaller and can be represented by smaller neighbourhoods. Synthesising texture at these coarser scales is much more computationally efficient than synthesising texture at a fine scale. This is due to the reduction in size of the image to be synthesised (sub image of original image). In addition, because large features can be represented by smaller neighbourhoods, the neighbourhood size is reduced considerably thus improving computational cost further.

The following sections outline the single resolution non-parametric algorithm and illustrate how wavelet decomposition may be used as a basis for this non-parametric texture synthesis. A comparison is given between our proposed method and the best previous approaches. This comparison is based on computational load as well as visual texture results. Finally, advantages and limitations of our algorithm are presented.

2 Single Resolution Texture Synthesis

Let $\mathbf{X}_s$ represent the image grid of size $M \times N$ to be synthesised and I_e be the sample input image of size $m \times n$ specified on the smaller grid $\mathbf{X}_e$. The algorithm assumes that I_e is large enough to capture the statistics of the underlying infinite texture. Let $\mathbf{p} \in \mathbf{X}$ be a pixel to be synthesised and $w(\mathbf{p})$ be the spatial

neighbourhood of pixels surrounding $\mathbf{p}$ with width w. To synthesise a value for $\mathbf{p}$ an approximation to the conditional probability distribution $P(\mathbf{p}|w(\mathbf{p}))$ is constructed and then sampled. The approximation is built by directly identifying all patches in the sample image that are *perceptually similar* in some way to the existing neighbourhood around the pixel to be synthesised. The pixels at the centre of these similar patches then represent an empirical measurement of the p.d.f. required.

Let $d(w(\mathbf{p_1}), w(\mathbf{p_2}))$ denote the perceptual distance between two neighbourhoods or patches centred at locations $\mathbf{p_1}$ and $\mathbf{p_2}$. d is defined to be the sum of squared intensity differences. The best matching patch w_{best} in the sample image, is first found, $w_{best} = argmin_{\mathbf{x}\in\mathbf{X_e}} d(w(\mathbf{p}), \mathbf{x})$. All example image patches w with $d(w(\mathbf{p}), w)) < (1+\epsilon)d(w(\mathbf{p}), w_{best})$ are included in the set $\Omega(\mathbf{p})$. In this application $\epsilon = 0.1$. The centre pixel values of patches in $\Omega(\mathbf{p})$ gives a histogram for $\mathbf{p}$ which can then be used to obtain a sample numerically. To preserve the local structure of the texture, the error for pixels near the centre of the neighbourhood i.e. that corresponding to $\mathbf{p}$, is larger than that for pixels close to the edge of the neighbourhood. This is achieved by weighting the distance measure $d(\cdot,\cdot)$ with a two-dimensional Gaussian Kernel. A kernel with variance $w/6.4$ is used.

In practice it is sensible to visit pixels in the synthesised image in an order specified by the number of known spatial neighbours. The algorithm initially seeks out pixel $\mathbf{p} \in \mathbf{I_s}$ with the most known spatial neighbours. As some of the spatial neighbours of $\mathbf{p}$ are unknown, the distance measure is modified to match only the known values in $w(\mathbf{p})$. This error is then normalised by the total number of known pixels when computing the conditional p.d.f. for $\mathbf{p}$. Figure 2 illustrates an overview of this searching procedure.

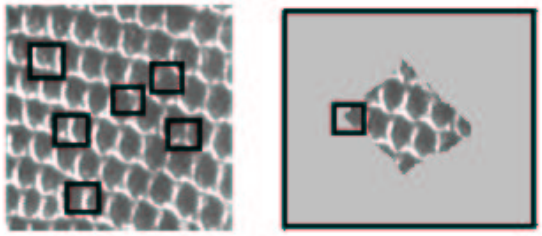


Figure 2: For each unknown pixel $\mathbf{p}$ in the synthesised image I_s (right) the algorithm searches all possible neighbourhoods in the sample image I_e (left) for a neighbourhood similar to that of the pixel $\mathbf{p}$. It then randomly chooses a matching neighbourhood and takes its centre to be the newly synthesised pixel.

Problems with boundary conditions are avoided by either treating the boundaries toroidally or padding with zeros. Here all boundaries were padded with zeros. The above algorithm generates impressive results on a wide variety of textures. However, searching the entire sample image for each pixel is computationally expensive and slows the algorithm considerably. A breakdown of the computational cost is given in section 3.2. In addition, the user defined neighbourhood width is critical to successful texture synthesis. To address these problems and also demonstrate the power of wavelets, the complex wavelet transform has been incorporated into the synthesis process.

3 Synthesising Texture using the Complex Wavelet Transform

The Dual Tree Complex Wavelet Transform (DT-CWT) originally proposed by Kingsbury has received much interest in image processing applications recently [8, 9, 13, 3, 7]. It builds upon the orthogonal Discrete Wavelet transform (DWT) and addresses some of its limitations such as, lack of shift invariance and poor directional selectivity [9]. The DT-CWT uses a dual tree of wavelet features that are assigned as real and imaginary components of complex wavelet coefficients. A full explanation of how the wavelet transform operates is beyond the scope of this paper but the interested reader is directed towards [9, 12]

for some supplementary material. As an outline however, the biorthogonal 2D DWT produces three band pass sub images at each level of the transform. These correspond to the lo-hi, hi-hi, hi-lo. The lo-lo sub image is passed onto the next level of the transformation. It is found that with real images, most of the significant information is contained within the first and second quadrants of the spectrum [13]. The 2D DT-CWT exploits this by producing three band pass sub images in each of the spectral quadrants 1 and 2. This gives a total of six band pass images with complex coefficients at each level. These images are strongly oriented at angles of $\pm 15^o, \pm 45^o, \pm 75^o$. Figure 3 shows the complex wavelet decomposition of an image containing a single bright circle. The sub band and lowpass images for the first level of decomposition are shown. The figure illustrates the directional sensitivity of the transform since different bands emphasise different parts of the circle contour. The DT-CWT gives a 4:1 redundancy for 2D images,. In a sense it is this redundancy that allows both shift invariance and good directional sensitivity.

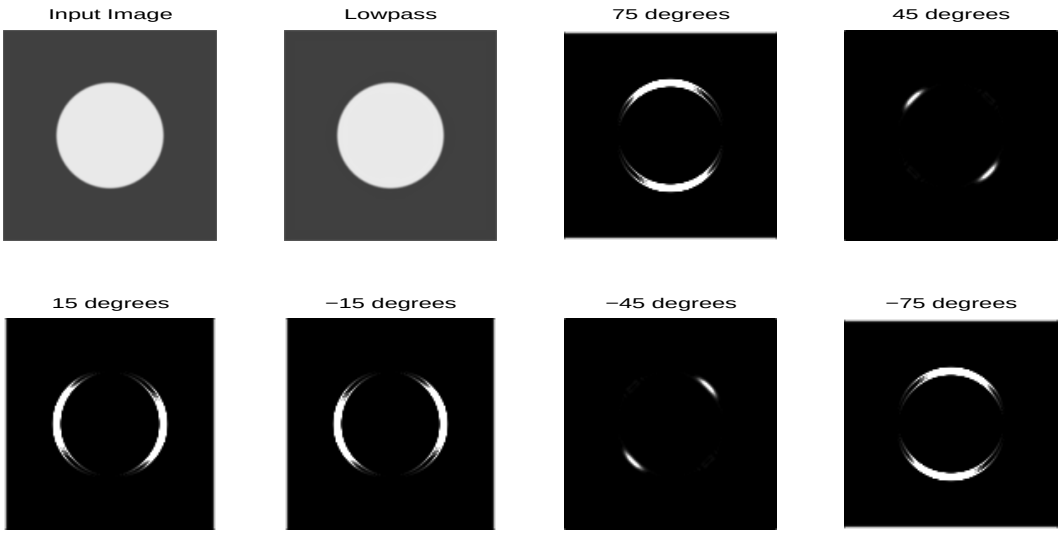


Figure 3: Illustration of sub image produced using DT-CWT. Figure shows input image (top left), lowpass image (top middle) and the bandpass images.

3.1 Algorithm

Given the initial sample image I_e of size $n \times m$ and the required output size $N \times M$ of the image to be synthesised I_s, the algorithm proceeds as follows.

- The n level complex wavelet transform is performed on the example image I_e. Using the initial dimensions of the image to be synthesised I_s, the dimensions of each of the sub band images and the final lowpass image are calculated. A sample of I_e is placed at the centre of each of the sub images of I_s. The size of the sample used should be consistent among the levels, i.e. at level n the seed should be half that used at level $n-1$. This is because of sub sampling in the wavelet transform. This sample is then surrounded by negative ones to indicate wavelet coefficients values to be synthesised.

- At the highest level, which is the coarsest level in terms of detail, the Efros searching algorithm given in section 2 is used to synthesise unknown wavelet coefficients in each of the six sub band images. In order to account for the correlation among sub band images and to maintain the efficiency of the algorithm, each of these images are searched coherently. That is, the same wavelet coefficient coordinate in each image is synthesised in parallel. Neighbourhoods from the six sub

band images and with the same centre coordinates are represented by a vector. The distance between two neighbourhoods is then given by the difference in magnitude between the two vectors representing them.

- Once the chosen wavelet coefficient has been selected from the sample image I_e, the six sub band images at the highest level are updated. Wavelet coefficients on the levels below follow the movement at the top level. This relationship is shown in Figure 3.1. That is, the wavelet coefficient at position (i, j) at level n corresponds to coefficients $(2i, 2j), (2i-1, 2j), (2i, 2j-1)$ and $(2i-1, 2j-1)$.

- This process is repeated for all unknown wavelet coefficients at the highest level. Once all of the wavelet coefficients have been generated, the synthesised image is inverse transformed to give an image that should resemble that of the sample texture. Note that, in order to avoid problems with boundary conditions, it is necessary to pad each sub image with zeros before performing the algorithm. This padding should be removed prior to inverse transform.

The above steps are based on generating grayscale images. To synthesise colour textures, first transform the image from the rgb colour space to the yuv colour space. Perform synthesis on the y (luminance) component and then propagate relevant coordinates to u and v components.

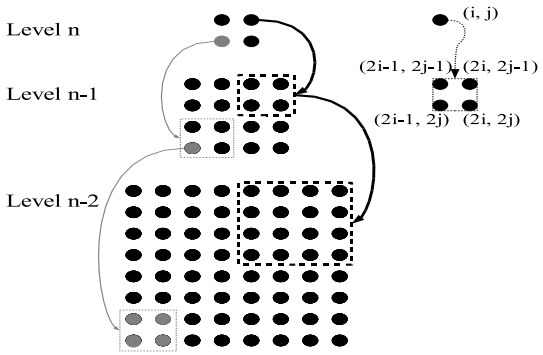


Figure 4: Simplified sub image of the DT-CWT showing the relationship between the wavelet coefficients across the different levels.

3.2 Computational Load

In order to demonstrate the advantages of our algorithm in terms of computational cost we have compared it against the original Efros algorithm [5]. Given a sample image I_e of size $m \times n$ and the image to be synthesised I_s of size $M \times N$. Let $\mathbf{p} \in I_s$ be a pixel to be synthesised and let w be the width of the neighbourhood of the square spatial neighbourhood surrounding $\mathbf{p}$. For each pixel to be synthesised in I_s, the algorithm needs to search up to nm locations. At each of those locations, $4w^2$ operations need to be performed to calculate the weighted sum squared difference. This is $4nmw^2$ operations in total for each searched site. Therefore to generate I_s of size $M \times N$ the algorithm will have to perform $4NMnmw^2$ operations.

In comparison, the algorithm proposed in this paper synthesises texture at the third level of the complex wavelet transform. At this level the dimensions of the sample image are $nm/16$ and the image to be synthesised are $NM/16$. Since all the six sub images at this level must be searched, the total number of operations is given as $NM/16 \times nm/16 \times 4w_1^2 \times 6$. Here w_1 is the neighboiurhood size for this process and is typically smaller than that needed for the Efros algorithm. The load for the CWT is roughly $80NM$ and is negligible in comparison to the overall load. Therefore the overall computational load of the new CWT algorithm is given by $NMmnw_1^2/10$. This shows that the new algorithm is faster than the

original Efros algorithm by a factor $40w^2/w_1^2$. For the experiments shown in this paper $w = 11,\ w_1 = 5$ yielding an improvement of a factor of about 200.

Using a simple Matlab implementation for a grayscale image on a 2.4 GHz P4 PC, the CWT algorithm can generate a 256×256 image from a sample texture measuring 128×128 in approximately 60 seconds. For a colour image, this process takes just over 80 seconds.

4 Results

Synthesised images generated by the wavelet synthesis algorithm are shown in Figures 5 and 6. In order to demonstrate the effectiveness of the algorithm, it was tested on a wide range of different textures. A visual comparison of the results obtained using other approaches was also carried out. Some of these results are shown in Figure 6. In each case the sample image measured 128×128 pixels and the synthesised image measured 256×256 pixels. When using complex wavelets, it is optimal to use image sizes that are powers of 2. Texture synthesis was carried out at level 3 of the complex wavelet with a neighbourhood size of 5×5 pixels.

As can be seen from Figure 6, the wavelet texture synthesis algorithm compares well against results obtained using the Wei and Levoy [15] and Efros and Leung [5] methods. The Wei and Levoy algorithm is similar to the method proposed here in that it is based on multiresolution synthesis. In their case they use Gaussian pyramids to separate the image into various frequency bands. When synthesising a pixel they begin initially at the top level and work their way down the pyramid. The neighbourhood of each pixel incorporates those pixels situated a level above on the pyramid. This allows for correlation among the sub images. However, it implies that the neighbourhood size is large, thus slowing down the process. Their tree vectorisation overcomes this but synthesising the entire Gaussian pyramid one pixel at a time is still computationally expensive.

The synthesised text in Figure 5 shows the impact of scale in texture synthesis. Because the algorithm synthesises at level 3 of the complex wavelet transform, whole words are synthesised rather than letters. This clearly demonstrates the effect of scale. At high levels of the transform, large features (words) are represented by fewer pixels. By synthesising texture at this level, words rather than individual letters are generated. Because the Efros method synthesises on a fine scale it will grow letters rather than words. That is, it grows the texture rather than the individual text.

Visually the textures generated using our CWT method compare well against the sample texture. However, following close inspection, there is some blurring present in the synthesised texture. This is more perceivable in sharp textures than others, e.g. the text. This problem is due to using the coarse level synthesis to direct the synthesis of the other levels, thus the detail at the finer levels is not refined. In addition, if the original sample image is compressed then these compression articfacts will be propagated in the synthesised image thus leading to more visual errors. Resolving this problem is the direction of current work.

5 Final Comments

In this paper a new texture synthesis algorithm was introduced. Given an initial sample image, the algorithm generates new texture using a simple searching process and which incorporates the Dual Tree Complex Wavelet Transform (DT-CWT). Results show that the algorithm works well on a wide variety of textures and has the advantage of reduced computational cost. By exploiting the properties of the DT-CWT, the algorithm also addresses some the problems of scale and correct neighbourhood size. Future work involves addressing some of the blurring problems associated with the output results. This will involve refining the pixel choice rather than just copying and adjusting the coordinates from those attained at the highest level.

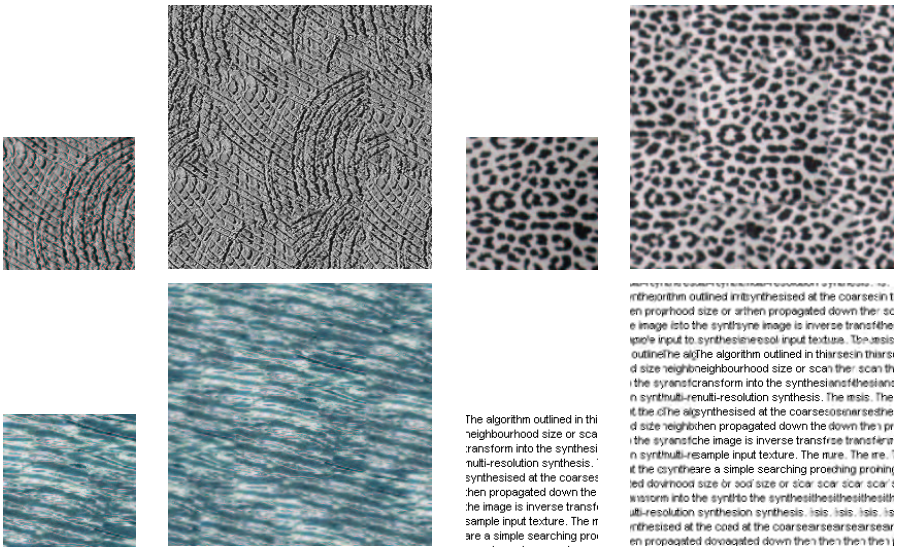


Figure 5: Results from the texture synthesis. The left hand image is the original image while the right is the synthesised image. Textures were synthesised on the third level of the transform.

Acknowledgements

This work was funded by the Irish Research Council Science Engineering and Technology (IRCSET) research scholarship foundation.

References

[1] Michael Ashikhmin. Synthesizing natural textures. In *ACM Symposium on Interactive 3D Graphics*, pages 217–226, Research Triangle Park, North Carolina, USA, March 2001.

[2] Raphaël Bornard. *Probabilistic Approaches for the Digital Restoration of Television Archives*. PhD thesis, École Centrale Paris, 2002.

[3] Peter de Rivaz and Nick Kingsbury. Complex wavelet features for fast texture image retrieval. *Proceedings of IEEE Conference on Image Processing, Kobe Japan*, pages 25–28, October 1999.

[4] Alexei A. Efros and William T. Freeman. Image quilting for texture synthesis and transfer. *Proceedings of SIGGRAPH 2001*, pages 341–346, August 2001.

[5] Alexei A. Efros and Thomas K. Leung. Texture synthesis by non-parametric sampling. In *IEEE International Conference on Computer Vision*, pages 1033–1038, Corfu, Greece, September 1999.

[6] D.J. Heeger and J.R. Bergen. Pyramid based texture analysis or synthesis. *SIGGRAPH 1995 Conference Proceedings, Annual Conference Series, ACM SIGGRAPH, Addison Wesley*, pages 229–238, August 1995.

[7] Cian W. Shaffrey Nick G. Kingsbury and Ian H. Jermyn. Unsupervised image segmentation via markov trees and complex wavelets. *Proceedings of IEEE Conference on Image Processing, New York, U.S.A*, September 2002.

[8] Nick Kingsbury. The dual-tree complex wavelet transform: A new technique for shift invariance and directional filters. In *IEEE DSP Workshop paper no. 86*, Bryce Canyon UT, USA, 1998.

[9] Nick Kingsbury. Image processing with complex wavelets. In *Phil. Trans. Royal Society London A, September 1999, on a Discussion Meeting on "Wavelets: The Key to Intermittent Information?"*, February 1999.

[10] Anil Kokaram. Parametric texture synthesis for filling holes in pictures. In *IEEE International Conference on Image Processing*, pages 325–328, Rochester, New York, USA, September 2002.

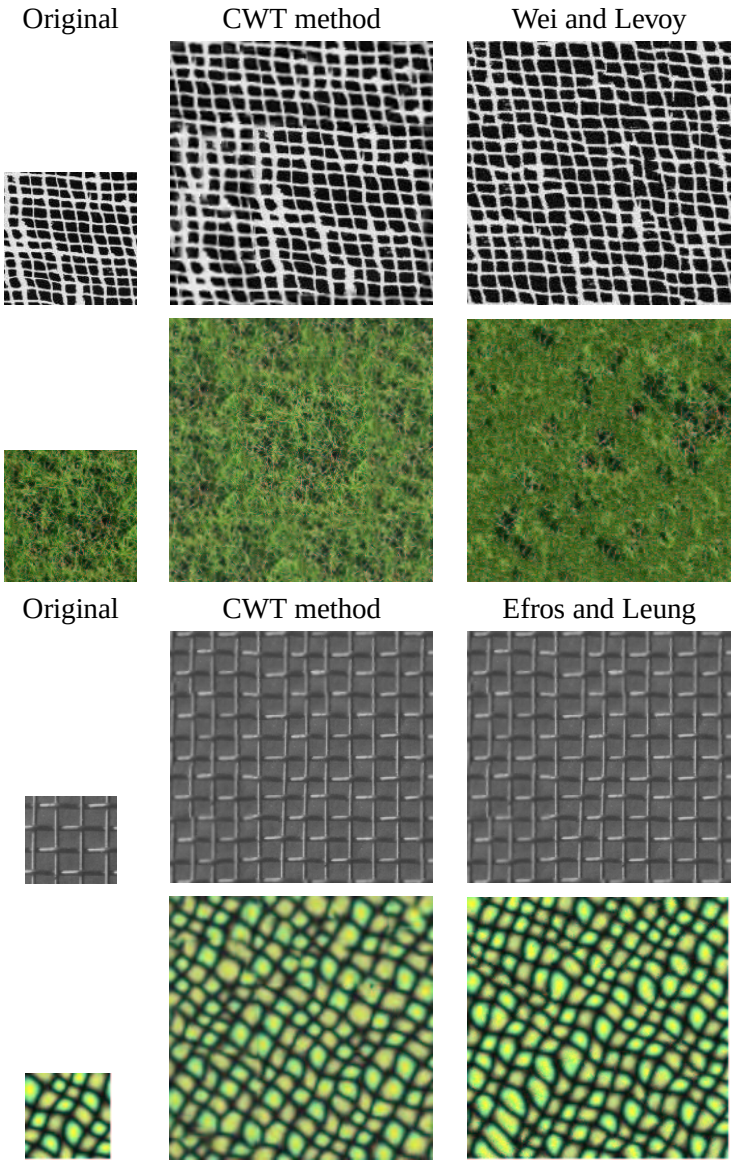


Figure 6: Comparison of different texture synthesis methods.

[11] J. Portilla and E.P. Simoncelli. A parametric texture model based on joint statistics of complex wavelet coefficients. *Int'l Journal of Computer Vision, vol.40(1)*, pages 49–71, December 2000.

[12] Olivier Rioul and Martin Vetterli. Wavelets and signal processing. *IEE Signal Processing Magazine vol 8, no 4*, October 1991.

[13] Sanjit K. Mitra Serkan Hatipoglu and Nick Kingsbury. Texture classification using dual-tree complex wavelet tranform. *Image Processing and its Applications, Conference Publication No. 465 IEE*, 1999.

[14] Yi-Chong Zeng Soo-Chang Pei and Ching-Hua Chang. Virtual restoration of ancient chinese paintings using color contrast enhancement and lacuna texture synthesis. *IEEE Transactions on Image Processing, volume 13, number 3*, March 2004.

[15] Li-Yi Wei and Marc Levoy. Fast texture synthesis using tree-structured vector quantization. *Proceedings of SIGGRAPH 2000*, pages 479–488, 2000.

NARROW BRANCH PRESERVATION IN MORPHOLOGICAL RECONSTRUCTION

Kevin Robinson* and Paul F. Whelan
Vision Systems Group
Dublin City University, Ireland

Abstract

We present a morphological approach to the reconstruction of fine branching structures in three dimensional data, developed from the basic procedures of reconstruction by dilation. We address a number of closely related questions arising from this reconstruction goal, including issues of structuring element size and shape, noise propagation, iterated approaches, and the relationship between geodesic and conditional dilation. We investigate and assess the effect and importance of these considerations in the context of the overall reconstruction process, and examine the effectiveness of the approach in addressing the task of reconstructing narrow branch features in noisy data.
Keywords*: Mathematical morphology, Reconstruction by dilation, Structuring element, Geodesic dilation, Conditional dilation*

1 Introduction

The classical reconstruction by dilation procedure [8, 10] is an effective and much utilised image processing tool applied extensively in the segmentation and classification of complex scenes [1, 2, 4, 6]. Seeded regions are retained while neighbouring unseeded regions are attenuated to the intensity level of the surrounding background data. The approach yields excellent results in isolating compact regions in noisy data. However when the regions of interest include fine branching structures the approach performs less well, especially in the presence of noise. This behaviour is due to the geodesic growth properties at the heart of the definition of reconstruction by dilation. The geodesic dilations which constitute a reconstruction by dilation guarantee that there exists a connected, strictly uphill (in terms of pixel intensity) path from each sample point to one of the original set of seed points which initiated the procedure. This property is what achieves the suppression of non-seeded high intensity regions.

The difficulty arises when a narrow element is encountered in a seeded region. Any signal drop-off along the narrow branch (due to noise or transitory signal reduction) can result in an undesirable attenuation of the intensity level along the entire remainder of the branch length. This is not an issue in the reconstruction of more compact regions as there will exist some convoluted high intensity path to carry the signal past the blockage. As the features in the region to be reconstructed become more and more narrow the chances of encountering a signal drop-off which can not be negotiated at the higher signal intensity level increase. In the case of fine branches, where the high intensity path is only one or two pixels wide the likelihood of undesirable signal suppression becomes extreme, leading to incomplete reconstruction of the desired objects.

In order to counter this difficulty we propose a non-geodesic extension to the reconstruction by dilation procedure aimed at bridging small gaps in the high intensity path while still effectively suppressing the signal intensity in neighbouring regions. The approach has the additional desirable property of preserving

*Corresponding author. *E-mail address:* kevin.robinson@eeng.dcu.ie

more fully the textural information in the reconstructed regions and suppressing the stepped contour effects which otherwise often manifest. These properties can be beneficial in terms of both the analysis and visualisation of the processed data.

The motivation for this work stems from a project whose aim is the segmentation of a ductal system called the biliary tree from a class of medical MRI scans of the abdomen. See Figure 1 for a maximum intensity projection (MIP) rendering of the three dimensional data from one such MRI scan. The ductal tree is clearly visible along with a number of occluding high intensity structures which we wish to suppress. Successful isolation of the finer branches towards the periphery of the tree is highly dependant on suppression of the high intensity proximal structures in the scene.

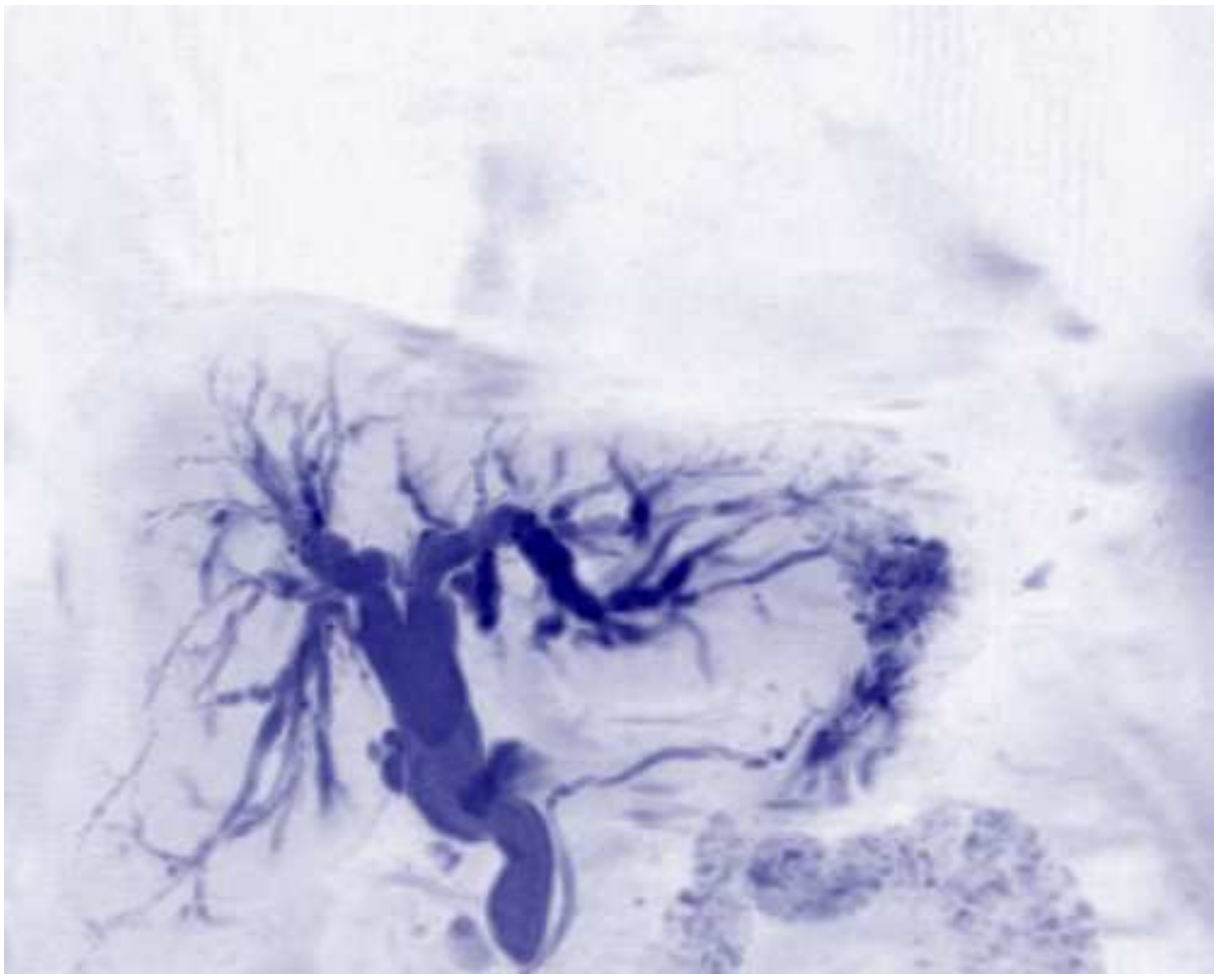


Figure 1: Illustration of the ductal tree whose segmentation is the ultimate goal. Neighbouring high intensity structures complicate the task.

2 Method

2.1 Morphological dilation operators

We first review the definitions of and the differences between dilation, conditional dilation, and geodesic dilation in greyscale data [7, 8]. Standard greyscale morphological dilation, $\delta^{(N)}$ is achieved where each sample point in the output is set equal to the maximum of it's own input intensity value and the values of

all sample points within a given neighbourhood in the source:

$$\delta^{(N)} I \;\widehat{=}\; \forall p : D \bullet \forall q : N_G(p) \bullet (q \leftarrow max(p, q)) \tag{1}$$

where D is the image domain and N signifies the 'size' of the dilation, represented in terms of N_G, the set of neighbouring samples constituting the structuring element to be applied in the dilation. Thus we can say, for all sample points p in the domain, for all sample points q in the neighbourhood of p, the output at q becomes the larger of p and q.

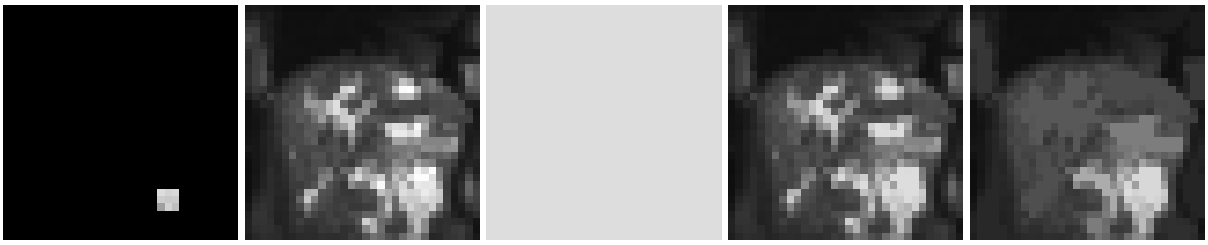
Conditional ($\delta_C^{\star(N)}$) and geodesic ($\delta_C^{(N)}$) dilations are then easily defined in terms of standard dilation as shown in Eqs. 2 and 3 respectively, where C represents the conditioning dataset, which must share the same domain as I, and $\wedge$ is the point-wise minimum operator.

$$\delta_C^{\star(N)} I \;\widehat{=}\; \delta^{(N)} I \wedge C \tag{2}$$

$$\delta_C^{(N)} I \;\widehat{=}\; \delta^{(1)} I \wedge C \ldots N \; times \tag{3}$$

Thus we can see that while in conditional dilation the point-wise minimum is applied only once after dilation to the full extent specified has been achieved, in geodesic dilation it is applied after each application of the fundamental dilation operator, with the two steps being repeated the necessary number of times. By applying the minimum at each iteration the procedure limits the growth of the dilated areas so as to avoid jumping over low intensity background regions in the conditioning mask and growing into unseeded neighbouring high intensity regions.

This behaviour is illustrated in Figure 2, where fig2c shows the standard (unconditional) dilation iterated until stability, which results in the entire domain arriving at the intensity level of the brightest sample point present in the marker (fig2a). Fig2d illustrates that the only difference between conditional dilation iterated until stability and the conditioning mask used to generate it, (fig2b) is that all sample points in the mask of a higher intensity than the highest intensity sample point in the marker have been capped at that maximum marker intensity level. Lastly fig2e shows geodesic dilations iterated until stability, of marker fig2a conditioned on mask fig2b, (the definition of reconstruction by dilation). The seeded region is retained while neighbouring regions are suppressed.



(a) Marker (b) Mask (c) Unconditional (d) Conditional (e) Geodesic

Figure 2: A comparison of standard, conditional, and geodesic dilation using an elementary two dimensional, eight connected structuring element, iterated until stability.

2.2 Hybrid reconstruction

As the caption in Figure 2 states the structuring element used in this example is an elementary two dimensional, eight connected structuring element. If the extents of the structuring element used in the dilation process reach beyond the innermost shell of sample points surrounding the origin the filter is no longer geodesic and cannot be used to perform reconstruction by dilation in the strictest sense of its definition. The procedure would amount to the application of more than one elementary dilation for

each application of the minimum operator (see Eq. 4). The manipulation of structuring elements is an important topic in this field [3, 5, 9], and proves valuable in the development of our procedure here.

$$\delta_C^{h(N,n)} I \;\widehat{=}\; \delta^{(n)} I \wedge C \ldots N \; times \tag{4}$$

This hybrid reconstruction of Eq. 4 has the potential to achieve the behaviour which we wish to utilise in our reconstruction approach, as it will allow the dilation to extend beyond small regions of intensity dropout, without breaching the more extensive low intensity valleys between disconnected neighbouring regions. The more dilations applied per application of the point-wise minimum operator, the wider the gaps which the reconstruction can cross. Thus we can see that there exists a family of reconstructions for any given starting data, where the optimal solution can be chosen in terms of how much physical separation exists at the point of closest proximity between seeded and unseeded regions in the data. So long as this measure allows sufficient scope to bridge the gaps in the fine branch components of the seeded regions, the reconstruction gaol can be successfully achieved.

2.3 Experimental procedure

We applied our approach to the isolation of a network of fine ducts in volumetric medical imaging data used to assess a region of the body in and around the liver. This network, called the biliary tree, collects bile produced in the liver, and delivers it to the small intestine where it is used in the digestive process. Figure 3 shows an example of one of the volumetric datasets under examination: the three dimensional data has been rendered in maximum intensity projection to illustrate the various regions visible, including the biliary tree which we wish to isolate and other structures which are to be suppressed. Note the many constrictions and signal voids visible in the branches of the tree. The ultimate goal of this work is to assist the radiologist in assessing the condition and operation of this ductal network. To this end we wish to achieve a clear and unobstructed reconstruction of the tree in order to facilitate its easy and effective examination and assessment.

We applied both standard reconstruction by dilation using 6, 18, and 26 connected structuring elements (the three fundamental three dimensional structuring elements leading to geodesic reconstructions), and we also applied a series of reconstructions utilising larger structuring elements. These larger elements were constructed so as to achieve approximately isometric reconstruction on the non-isometric volume data which we are analysing. The data is isometric in the x and y directions, with voxel dimensions of approximately 1.3mm each way, but in the z direction the voxel dimensions increase to 4.0mm. Thus in order to achieve dilation more consistently in all directions an anisotropic (in voxel terms) approach was preferred, so as to compensate for the non-cubic nature of the data. We found this to be the most effective approach, maximising the amount of unconstrained dilation we could use between applications of the minimum operator before the procedure starts to include unwanted structures in the reconstruction.

3 Results

We processed a number of datasets using both traditional geodesic reconstruction by dilation and our hybrid reconstruction approach applied at varying strengths, and assessed the reconstruction results achieved in each case. Figure 4 illustrates the superior intensity preservation characteristics of the hybrid reconstruction approach in the processing of objects of interest which include fine branching features. The level of retention achieved increases with the strength of the hybrid reconstruction applied.

We applied the series of reconstructions and then measured the degree of intensity suppression in the neighbouring unseeded regions and at the extreme ends of a number of target branches of varying widths within the seeded regions. Figure 4 shows the variations in signal drop-off observed at two different levels of our hybrid reconstruction, along with standard reconstruction by dilation. In this way we were able to demonstrate the enhanced level of reconstruction achieved using large anisotropic structuring

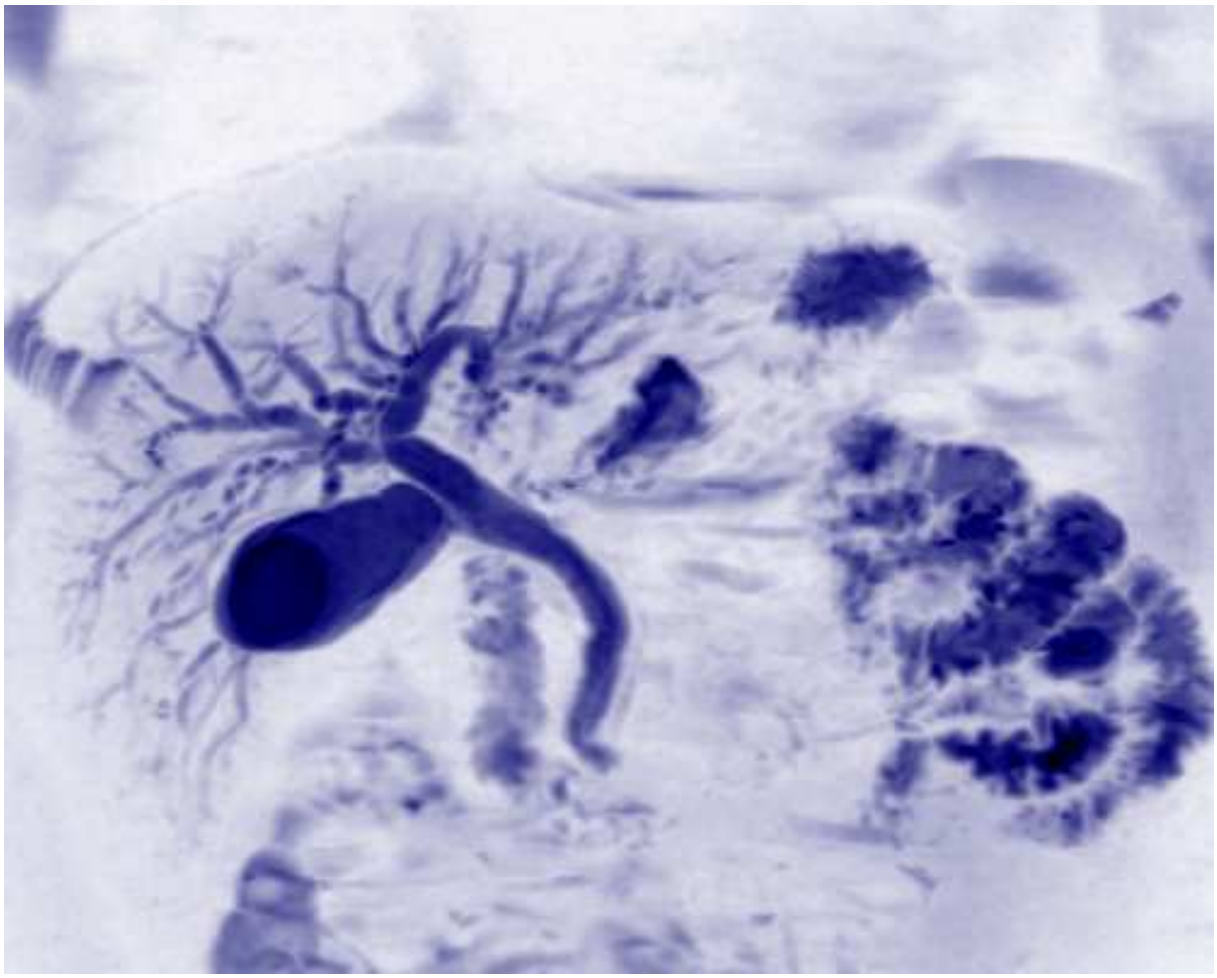
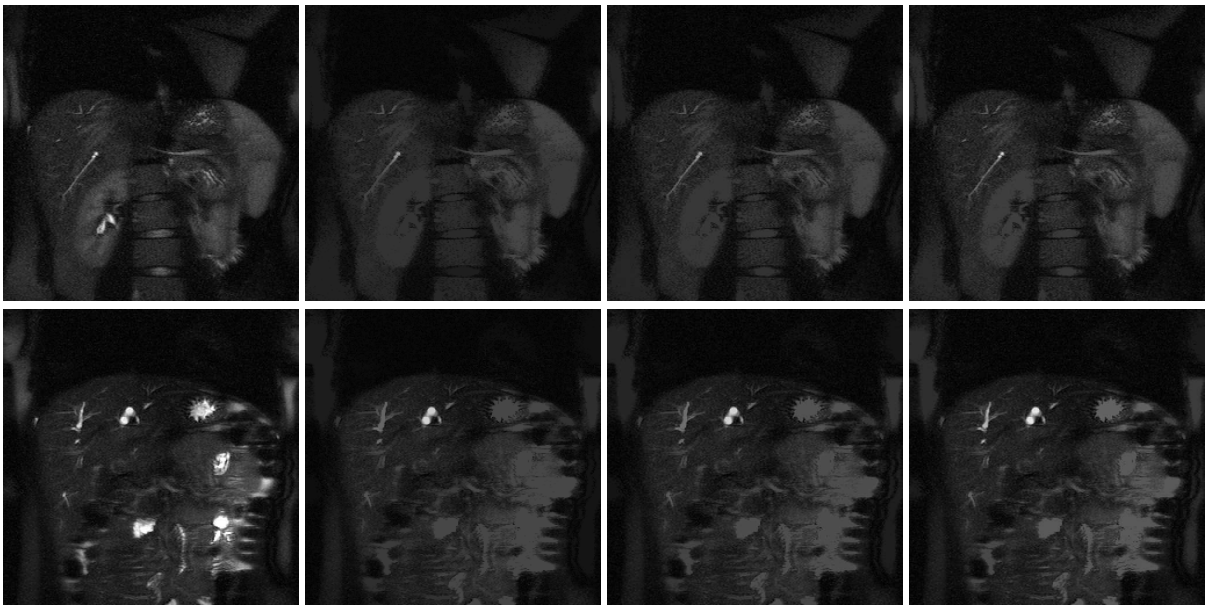


Figure 3: Maximum intensity projection of one of the datasets examined in the study demonstrating the biliary tree along with numerous unwanted high intensity regions.



(a) Original (b) Geodesic (c) 38 Anisotropic (d) 54 Anisotropic

Figure 4: Two sections through a volume dataset showing branch tips at various levels of reconstruction demonstrating both fine and course branches: a) original unfiltered data, b) 6-connected geodesic reconstruction by dilation, c) reconstruction using an anisotropic 38 element structuring element, d) reconstruction using an anisotropic 54 element structuring element

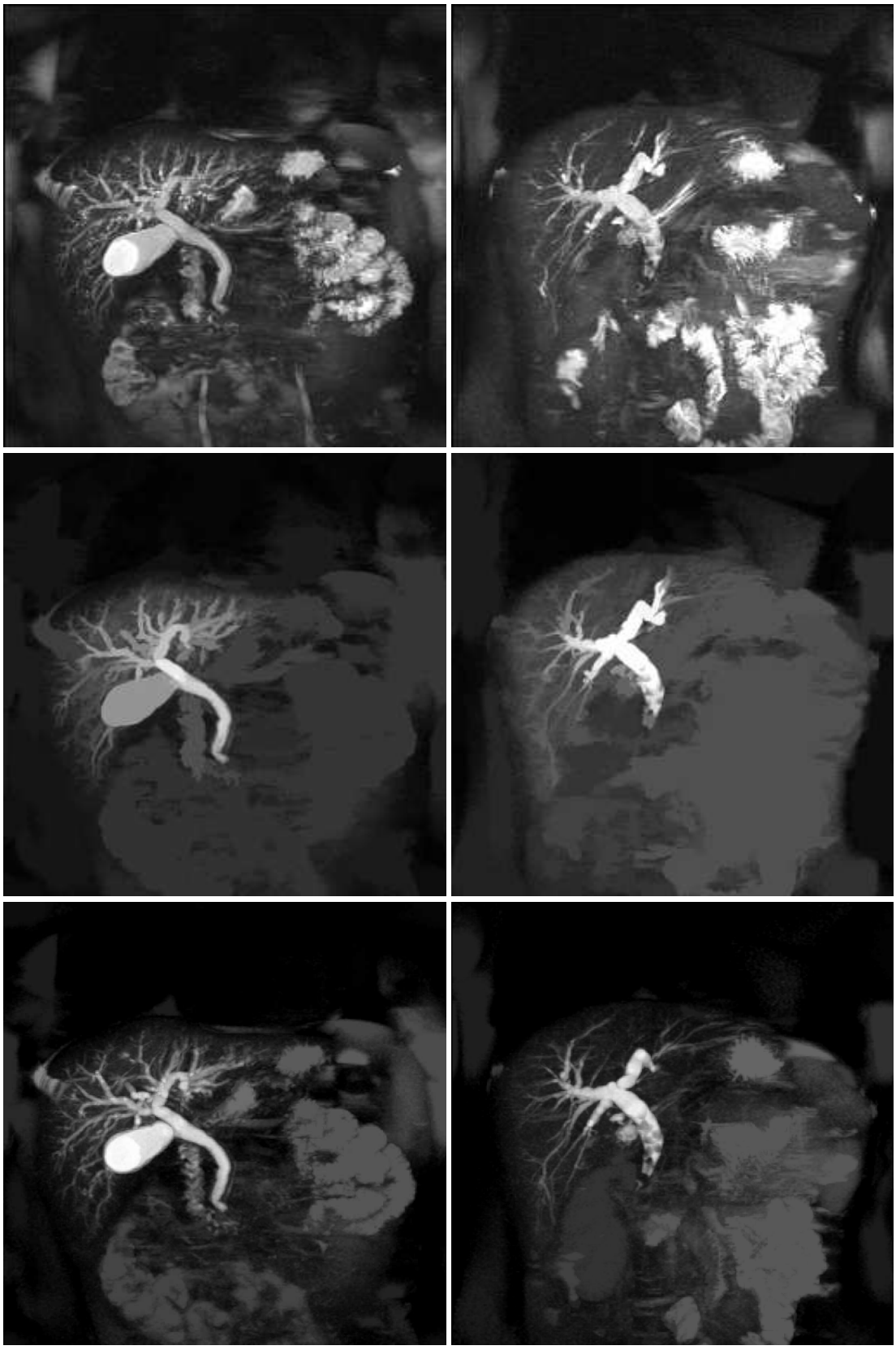
elements. Eventually as the size is increased beyond the optimal, the signal intensity in neighbouring regions begins to pick up until in the extreme the reconstruction approximates the original unfiltered data, with only the highest intensity peaks in the data being reduced to the level of the highest intensity sample points present in the original seed data.

In Figure 5 we can in addition observe the enhanced texture retention properties of our hybrid reconstruction approach, where the second row of images achieved using traditional reconstruction by dilation demonstrate excessive smoothing and the introduction of sharp graduations within the reconstructed tree, while the images on row three show superior preservation of the fine detail from the original data (shown on the top row). This can be of particular importance for the accurate interpretation of the final data by the radiologist.

We also observed the role that noise in the data plays in propagating the high intensity signal across background valleys. Once the approach departs from the geodesic scheme where a strict uphill intensity path is always retained between any point and an original seed region, isolated high intensity noise peaks in the background regions have the potential to piggyback the signal across the valleys like a series of stepping stones. This effect makes strong salt and pepper noise particularly unfavourable in the application of our technique. The nature of the noise distribution typical to our data makes the approach more applicable in this case as even with very strong dilations the degree of the unwanted propagation is kept to a manageable level due to the intensity and spatial spread present in the signal noise which means that the maintenance of a high intensity steppingstone path across valleys of any significant width becomes extremely unlikely.

4 Conclusions

By extending the basic principles of reconstruction by dilation beyond the geodesic case we have presented a hybrid reconstruction technique specifically designed to optimally reconstruct objects containing



(a) Volume Study 1 (b) Volume Study 2

Figure 5: Maximum intensity projections of two of the datasets from our study, preformed on the original (top row), geodesic reconstructed (middle row), and hybrid reconstructed (bottom row) data volumes.

fine branching structures in the source data while still effectively attenuating the signal from neighbouring unwanted high intensity structures.

Through the application of these techniques we have developed an effective and efficient image processing procedure which yields superior reconstruction results as a precursor to both further automated segmentation, classification, and analysis, and enhanced and simplified manual review of the data by the trained radiologist.

References

[1] J. Angulo and J. Serra. Automatic analysis of dna microarray images using mathematical morphology. *Bioinformatics*, 19(5):553–562, 2003.

[2] A. Araujo, S. Guimaraes, and G. Cerqueira. A new approach for old movie restoration. In *Proc. SPIE-High-speed Imaging and Sequence Analysis*, pages 67–77, 2001.

[3] L. Ji, J. Piper, and J-Y. Tang. Erosion and dilation of binary images by arbitrary structuring elements using interval coding. *Pattern Recognition Letters*, 9(3):201–209, 1989.

[4] V. Metzler, C. Thies, and T. Lehmann. Segmentation of medical images by feature tracing in a selfdual morphological scale-space. In *Proc. SPIE-Medical Imaging*, pages 139–150, 2001.

[5] H. Park and J. Yoo. Structuring element decomposition for efficient implementation of morphological filters. *IEE Proceedings: Vision, Image and Signal Processing*, 148(1):31–35, 2001.

[6] P. Salembier, P. Brigger, J. Casas, and M. Pardas. Morphological operators for image and video compression. *IEEE Transactions on Image Processing*, 5(6):881–898, 1996.

[7] J. Serra. *Image Analysis and Mathematical Morphology*. Academic Press, 1982.

[8] P. Soille. *Morphological Image Analysis: Principles and Applications*. Springer-Verlag, 1999.

[9] M. van Droogenbroeck and H. Talbot. Fast computation of morphological operations with arbitrary structuring elements. *Pattern Recognition Letters*, 17:1451–1460, 1996.

[10] L. Vincent. Morphological grayscale reconstruction in image analysis: Applications and efficient algorithms. *IEEE Transactions on Image Processing*, 2(2):176–201, 1993.

DISCRETE FOURIER TRANSFORM QUANTISATION TABLES FOR DIGITAL HOLOGRAMS OF THREE-DIMENSIONAL OBJECTS

Conor P. Mc Elhinney,[1] Alison E. Shortt,[1] Thomas J. Naughton,[1] Bahram Javidi[2]

[1]Dept. of Computer Science
National University of Ireland,
Maynooth
County Kildare, Ireland
email: tom.naughton@may.ie

[2]Electrical and Computer Engineering
University of Connecticut
371 Fairfield Road, Unit 1157
Storrs, CT 06269, USA
email: bahram@engr.uconn.edu

Abstract

We report on the creation of a general-purpose discrete Fourier transform (DFT) quantisation table that can be universally applied to digital hologram data of three-dimensional (3D) objects, with the aim of efficiently compressing the data. We captured digital holograms (whole Fresnel fields) of 3D objects using phase-shift interferometry. The complex-valued fields were decomposed into nonoverlapping blocks of 8×8 (or 16×16) pixels and transformed with the DFT. The relative importance of each of the blockwise DFT coefficients was traced throughout a digital hologram, and over multiple holograms. We used rms error in the reconstructed image to quantify importance in the DFT domain. We have found that DFT based quantisation gives one far more flexibility in choosing the quality/compression rate trade-off than a rigid uniform quantisation approach. This is the first blockwise DFT study to have been performed on digital holographic data and it has produced the first quantisation table that could be suitable for a JPEG-style compressor for complex-valued digital hologram data of 3D objects.

Keywords*: three-dimensional image processing, image compression, digital holography, discrete Fourier transform, JPEG*

1 Introduction

Holography is an established technique for recording and reconstructing three-dimensional (3D) objects. Digital holography [1, 2, 3, 4, 5, 6] has recently become feasible due to recent advances in megapixel CCD sensors with high spatial resolution and high dynamic range. A technique known as phase-shift interferometry [3, 5] (PSI) was used to create our in-line digital holograms [6, 7]. The resulting digital holograms are in an appropriate form for data transmission and digital image processing (noise removal, object recognition, and so on). A hologram encodes different views of a 3D object from a small range of angles [8, 9]. In order to reconstruct a particular 2D perspective of the object, the appropriate region of pixels must be extracted from the hologram and simulated Fresnel propagation applied [10, 5, 6]. It has also been proposed to stream digital holograms over a network to generate a form of 3D video [11]. The initial stages of such a proposal has involved the compression of individual holographic frames followed by object reconstruction [11, 12]. A real-time optical reconstruction method using the complex field of a digital hologram has also been demonstrated [13].

Given that each hologram is 65 Mbytes in size in its native format, real-time streaming of uncompressed digital holographic data is impractical. Lossless compression techniques, based on Lempel-Ziv,

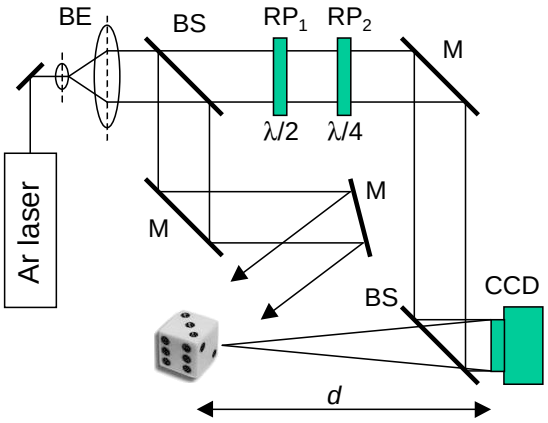


Figure 1: Experimental setup for PSI: BE, beam expander; BS, beam splitter; RP, retardation plate; M, mirror.

Huffman, and Burrows-Wheeler algorithms, have been shown to perform poorly on digital holographic data [11]. In order to facilitate more efficient transmission and storage of digital holograms, lossy techniques based on quantisation have been applied in the past [11, 13, 12, 14]. Wavelets have also been applied to digital holograms [15, 16], although not yet in the context of data compression. The performance of baseline JPEG [17] (the standard JPEG implementation) has been shown to suffer greatly in the presence of speckle noise [18, 19]. However, we believe that there is some potential in the JPEG-style approach if it is tailored to digital hologram data. In this paper, we investigate the possibility of creating a quantisation table that can be universally applied to digital hologram data in a future JPEG-style compressor. In future work we will look at a JPEG2000-like wavelet approach.

Hologram compression differs to image compression principally because our holograms store 3D information in complex-valued pixels [each pixel is a (8 byte, 8 byte) real-imaginary pair], and because of the inherent speckle content which gives the holograms a white-noise appearance. Holographic speckle is difficult to remove because it actually contains 3D information. Furthermore, a change locally in a digital hologram (introduced during lossy compression, for example) will, in theory, affect the whole reconstructed object. When gauging the errors introduced by lossy compression, we are not directly interested in the defects in the hologram itself, only how compression noise affects the quality of perspectives of the 3D object reconstructed through simulated Fresnel propagation. We therefore use a reconstruction plane metric to quantify the quality of our lossy compressed-decompressed holograms.

In Sect. 2, we describe how 3D objects are captured using phase-shift digital holography. We briefly summarise the JPEG algorithm in Sect. 3 and determine which of our DFT components produces the highest reconstruction-domain error in Sect. 4. The quantisation table is designed and evaluated in Sect. 5 and we conclude in Sect. 6.

2 Phase-Shift Digital Holography

We record Fresnel fields with an optical system based on a Mach-Zehnder interferometer (see Fig. 1). A linearly polarized Argon ion (514.5 nm) laser beam is expanded and collimated, and divided into object and reference beams. The object beam illuminates a reference object placed at a distance of approximately $d = 350$ mm from a 10-bit 2028×2044 pixel Kodak Megaplus CCD camera. The linearly polarized reference beam passes through half-wave plate RP_1 and quarter-wave plate RP_2. By selectively removing the plates we can achieve four phase shift permutations of 0, $-\pi/2$, $-\pi$, and $-3\pi/2$. The reference beam combines with the light diffracted from the object and forms an interference pattern in the plane of the camera. At each of the four phase shifts we record an interferogram. We use these four real-valued images to compute the camera-plane complex field $H_0(x, y)$ by PSI [3, 5]. We call this

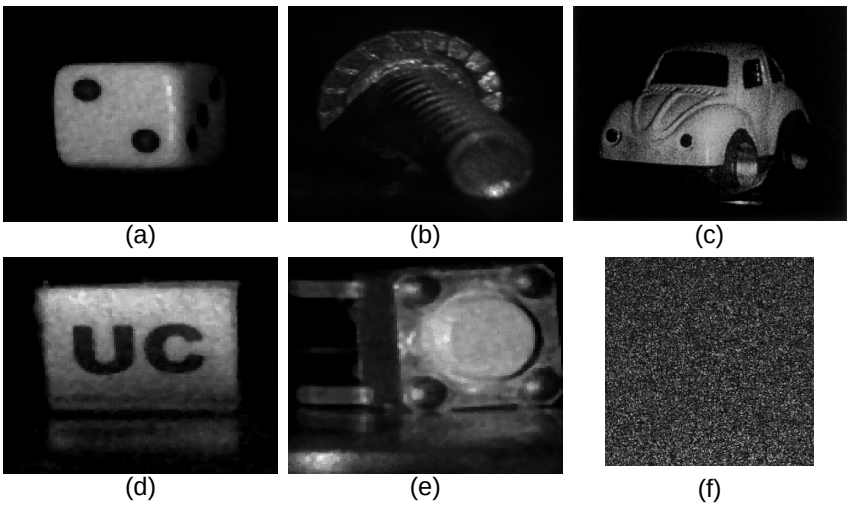


Figure 2: The set of holograms used in these experiments: (a)-(e), reconstructed objects from holograms no. 1 through no. 5, respectively; (f), the amplitude of an example 512×512 subset of digital hologram no. 1.

computed field a digital hologram.

A digital hologram $H_0(x, y)$ contains sufficient amplitude and phase information to reconstruct the complex field $U(x, y, z)$ in a plane in the object beam at any distance z from the camera [5, 6, 10]. This can be calculated from the Fresnel approximation [9] as

$$U(x, y, z) = \frac{-\mathrm{i}}{\lambda z} \exp\left(\mathrm{i}\frac{2\pi}{\lambda} z\right) H_0(x, y) \star \exp\left[\mathrm{i}\pi \frac{\left(x^2 + y^2\right)}{\lambda z}\right] \quad , \tag{1}$$

where λ is the wavelength of the illumination and $\star$ denotes a convolution operation. At $z = d$, and ignoring errors in digital propagation due to discrete space (pixelation) and rounding, the discrete reconstruction $U(x, y, z)$ closely approximates the physical continuous field $U_0(x, y)$.

Furthermore, as with conventional holography [8, 9], a windowed subset of the Fresnel field can be used to reconstruct a particular view of the object. As the window explores the field a different angle of view of the object can be reconstructed. The range of viewing angles is determined by the ratio of the window size to the full CCD sensor dimensions. Our CCD sensor has approximate dimensions of 18.5×18.5 mm and so a 1024×1024 pixel window has a maximum lateral shift of 9 mm across the face of the sensor. With an object positioned $d = 350$ mm from the camera, viewing angles in the range $\pm 0.74^\circ$ are permitted. Smaller windows will permit a larger range of viewing angles at the expense of image quality at each viewpoint. Five digital holograms of different 3D objects were used in our experiments. A reconstruction of each is shown in Fig. 2. Figure 2(f) shows the white-noise appearance of the holograms themselves.

3 JPEG

The baseline JPEG algorithm can be summarised as follows [17, 18]. The image is subdivided into 8×8 pixel blocks and each pixel value rescaled linearly to the range $[-128, 127]$. Each block is transformed with the discrete cosine transform (DCT) and the transformed values are stored with 12 bits/pixel. The DCT coefficients obtained are quantised to a lower precision with a user-defined 8×8 quantisation table of 8 bit values. Each DCT coefficient in the block is divided by its corresponding quantisation value and rounded to the nearest integer. This rounding operation essentially performs the lossy compression. The 64 DFT coefficients in each block are then coded losslessly. The values in the quantisation table determine how finely or coarsely to quantise each DFT coefficient. A more coarse quantisation (higher

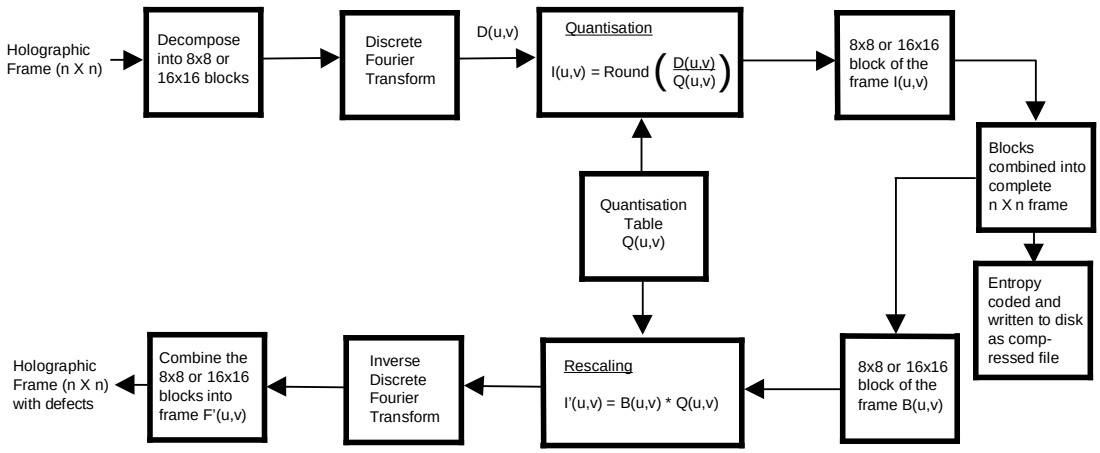


Figure 3: Sequence of steps used to evaluate our compression-decompression routine. The error is measured between reconstructions from the original holograms (at the input of the sequence) and quantised holograms (at the output of the sequence). Lossless compression (at the half-way stage) is used to get a meaningful measure of the size of the quantised digital hologram.

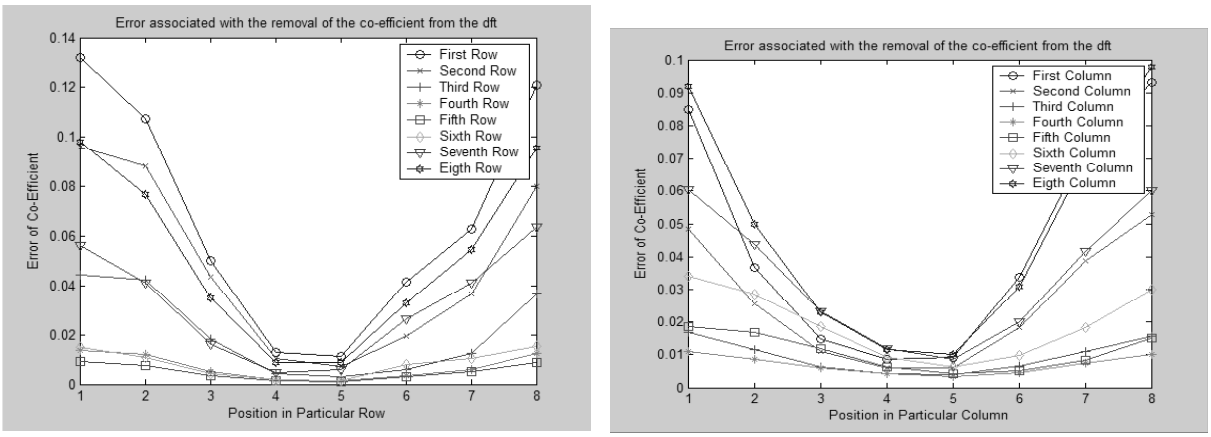
values) results in an increase in degradation in the image, but allows the lossess steps to be more effective. To reconstruct the image from this new form the lossless step is reversed, the implicit rescaling performed by the quantisation table is undone, and each 8×8 pixel block is inverse cosine transformed.

4 Blockwise analysis of discrete Fourier coefficients

Although digital holograms are complex-valued, the sequence of processes in baseline JPEG can still be used when modified for this richer data type. Central to the JPEG algorithm is the quantisation table. The table(s) in baseline JPEG were designed based on objective and subjective evaluation of compressed versions of a wide range of images. Our digital holograms look like white-noise functions and are not at all similar to the slowly spatially-varying continuous-tone images that JPEG was designed for. Therefore it could be expected that JPEG would not perform well on such data. Furthermore, the DCT is a simplification of the discrete Fourier transform (DFT) designed to output real-valued spectra on real-valued arguments. We replace the blockwise DCT with a blockwise DFT and design a quantisation table appropriate for digital hologram data. The application and evaluation of such a quantisation table is summarised in Fig. 3.

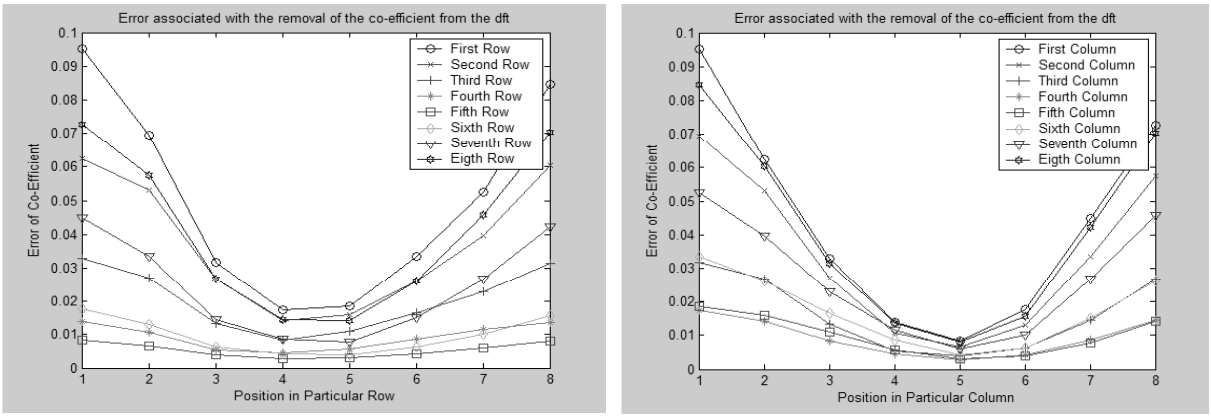
In order to design a suitable quantisation table, we analyse the (complex-valued) DFT coefficients in each block and rank these coefficients based on their relative influence on the final reconstructed object. We used separately an 8×8 pixel blockwise DFT and a 16×16 pixel blockwise DFT in our experiments. We used a 1024×1024 pixel window from each of our holograms so that the holograms could be divided evenly into 8×8 pixel or 16×16 pixel nonoverlapping blocks. We blockwise Fourier transformed the digital hologram, removed a particular coefficient from each DFT block, performed an inverse blockwise DFT, reconstructed the 3D object through simulated Fresnel propagation, and recorded the error in the reconstruction. The coefficient was removed by multiplying each transformed block of holographic data by a 8×8 pixel (or 16×16 pixel) mask of 1's that contained a single zero at the desired position. We repeated this for each of the coefficients in the block, and for each of several digital holograms. Error in the reconstruction U' was measured by a comparison with an equivalent reconstruction U_0 from the original digital hologram. The two reconstructions were compared in terms of normalised rms (NRMS) difference in their intensities, defined as

$$D(U') = \left(\sum_{m=0}^{N_x-1} \sum_{n=0}^{N_y-1} \left[|U_0(m,n)|^2 - |U'(m,n)|^2 \right]^2 \times \left\{ \sum_{m=0}^{N_x-1} \sum_{n=0}^{N_y-1} \left[|U_0(m,n)|^2 \right]^2 \right\}^{-1} \right)^{1/2}, \quad (2)$$



(a) (b)

Figure 4: Plots of the error resulting from removing one-by-one each coefficient of a blockwise 8×8 pixel DFT of hologram no. 2: (a) grouped into rows, and (b) grouped into columns.



(a) (b)

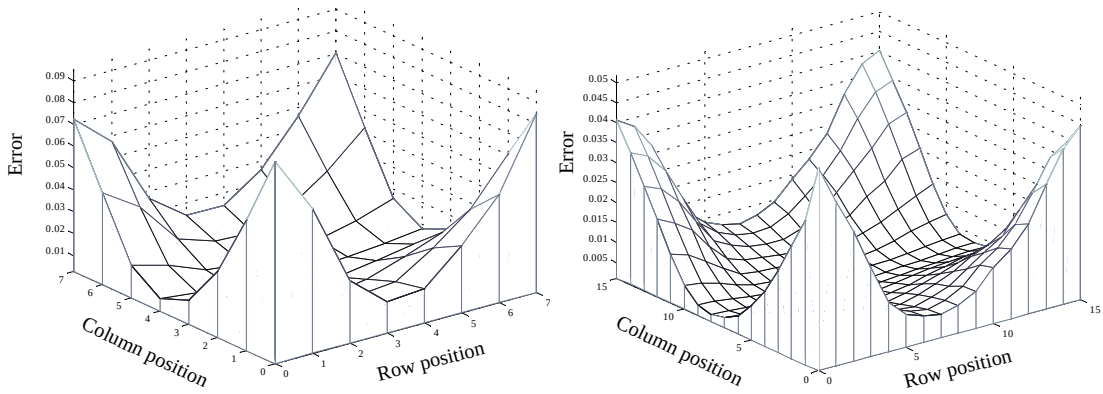
Figure 5: Plots of the coefficients in the 8×8 pixel mean error table: (a) grouped into rows, and (b) grouped into columns.

where (m, n) are discrete spatial coordinates in the reconstruction plane, and N_y and N_x are the height and width of the reconstructions, respectively. The rms differences for one such hologram are shown in Fig. 4. The larger the error introduced into the reconstruction, the more important that coefficient. The rms differences for the five digital holograms were averaged to construct the mean error table of DFT coefficients, shown in Figs. 5 and 6(a). The most interesting aspect of this error table is that the coefficients at each of the four corners retain important digital holographic information, whereas for standard continuous-tone images all of the important coefficients are located in a single corner close to the dc coefficient. A 16×16 mean error table was similarly created, and revealed the same characteristic structure [see Fig. 6(b)]. This meant that the manner in which we constructed the 8×8 pixel and 16×16 pixel quantisation tables could be the same.

5 Quantisation table creation

The values in the mean error table were rescaled to the range $[0, 1]$. The higher the value is in this table, the more important is that coefficient. The mean error table can be regarded as an indicator of the relative importance of each DFT coefficient, and can be used as the basis for a quantisation table.

Applying this mean error table directly as a quantisation table will introduce a fixed amount of loss



(a) (b)

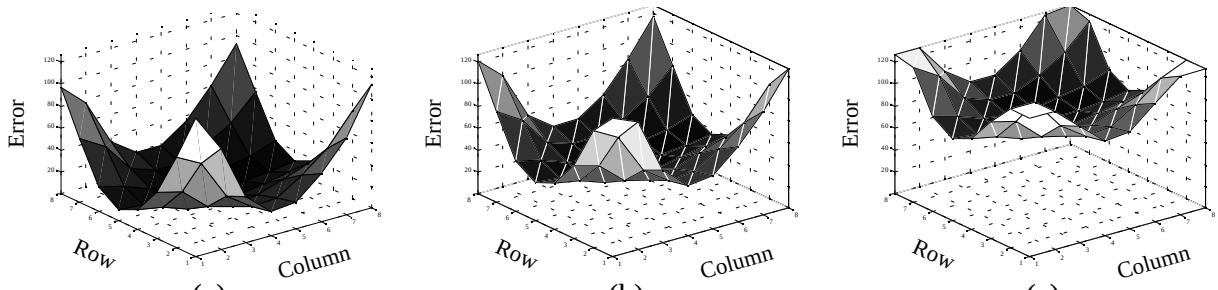
Figure 6: 3D plot of two mean error tables: (a) 8×8 pixel, and (b) 16×16 pixel.

127	93	43	23	25	45	71	113
83	71	35	19	21	35	53	81
43	37	17	11	15	23	31	41
19	15	7	7	7	11	15	19
11	9	5	3	5	5	7	11
23	17	9	5	5	9	13	21
59	45	19	11	11	21	35	57
97	77	35	19	19	35	61	93

Figure 7: 8 bit quantisation table for 8×8 pixel DFT compression.

into the digital hologram. In order to vary the loss, and indirectly vary the compression rate, we define a function f to generate quantisation tables that takes three arguments: the aforementioned mean error table $q : \{1, 2, \ldots, 8\}^2 \rightarrow [0, 1]$ (in the case of a 8×8 pixel blocks), a number of bits of resolution $b \in \mathbb{N}$, and an offset $t \in \mathbb{N}$. b exponentially varies the coarseness of the quantisation and t linearly varies the quality at a more fine level. f is defined as $f(q, b, t) = \lfloor q \times 2^{b-1} + 0.5 \rfloor + t$. The most straightforward 8 bit quantisation table for 8×8 pixel blocks is generated from $f(q, 8, 0)$ and is shown in Fig. 7. Figure 8 shows 3D plots of three tables generated from $f(q, 8, 0)$, $f(q, 8, 24)$, and $f(q, 8, 64)$.

The quantisation tables were applied to our digital holograms. In order to effect a final lossless compression stage typical in JPEG algorithms, the quantised blockwise Fourier transformed digital holograms were compressed with an implementation of the LZ77 algorithm [20]. Figure 9 shows plots of file size against reconstruction NRMS error for two digital holograms and two quantisation table sizes. Each curve (the curves labelled "nonuniform" in the legend) is a result of experimenting with several different values for offset t. For comparison we include the results for uniform quantisation (the single points labelled "uniform" in the legend). Although our DFT based quantisation does not outperform the com-



(a) (b) (c)

Figure 8: 3D plots of 8 bit quantisation tables for 8×8 pixel DFT compression with offsets of (a) 0, (b) 24, and (c) 64.

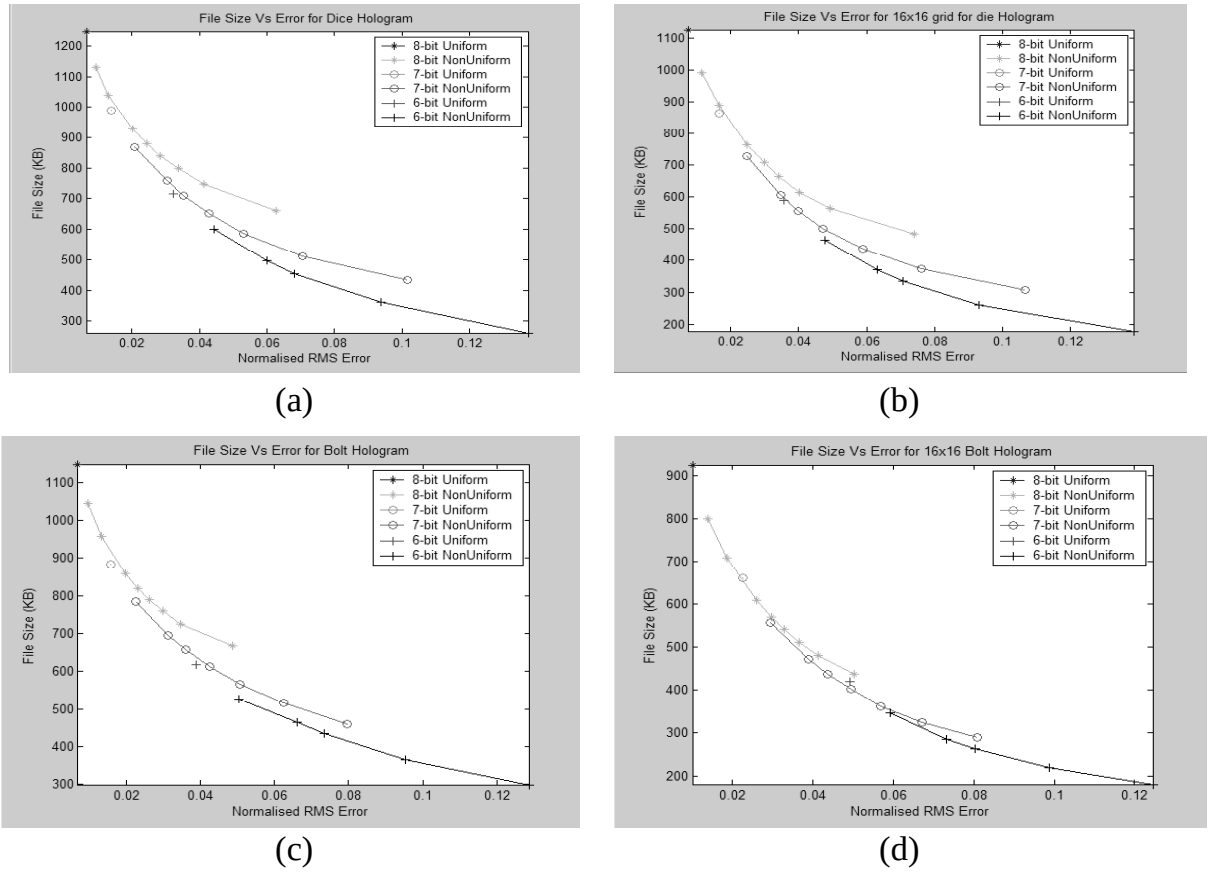


Figure 9: Results of applying 8 bit DFT quantisation to hologram no. 1 with (a) 8×8 pixel table, and (b) 16×16 pixel table; and to hologram no. 2 with (c) 8×8 pixel table, and (d) 16×16 pixel table.

bination of uniform quantisation and lossless encoding, it does give one far more flexibility in choosing the quality/compression rate than a rigid uniform quantisation approach. Furthermore, our knowledge of the distribution of values in each 8×8 pixel block (more significant values in the corners, for example) should allow us to perform a more efficient lossless stage in the future (similar to JPEG's zig-zag DCT ordering approach) than would be possible with uniformly quantised data.

The 1024×1024 pixel windows of each digital hologram that we used in these experiments required 16,384 Kbytes of storage space in uncompressed form, so a compressed size of 500 Kbytes (in Fig. 9) corresponds to a compression rate of over 30. When LZ77 is applied to unquantised digital holograms it achieves compression rates of less than 2.0 [11].

6 Conclusion

We have discussed the creation of a DFT based quantisation table that can be generally applied for the compression of digital holograms of 3D objects. Our quantisation table was based on a study that ranked each blockwise DFT coefficient in the hologram domain based on its importance to the reconstructed domain. We have found that DFT based quantisation gives one far more flexibility in choosing the quality/compression rate trade-off than a rigid uniform quantisation approach. Our quantisation table could form the basis for a future JPEG-style compressor for complex-valued digital hologram data of 3D objects. In future work we will look at a JPEG2000-like wavelet approach.

Acknowledgements

The authors wish to thank Enrique Tajahuerce and Yann Frauel for use of their digital hologram data. The third author wishes to acknowledge support from Enterprise Ireland.

References

[1] Joseph W. Goodman and R. W. Lawrence. Digital image formation from electronically detected holograms. *Applied Physics Letters*, 11(2):77–79, 1967.

[2] Leonid P. Yaroslavskii and Nikolai S. Merzlyakov. *Methods of Digital Holography*. Consultants Bureau, Plenum, New York, 1980. Translated from Russian by Dave Parsons.

[3] J. H. Bruning, Donald R. Herriott, J. E. Gallagher, D. P. Rosenfeld, A. D. White, and D. J. Brangaccio. Digital wavefront measuring interferometer for testing optical surfaces and lenses. *Applied Optics*, 13(11):2693–2703, November 1974.

[4] Ulf Schnars and Werner P. O. Jüptner. Direct recording of holograms by a CCD target and numerical reconstruction. *Applied Optics*, 33(2):179–181, January 1994.

[5] Ichirou Yamaguchi and Tong Zhang. Phase-shifting digital holography. *Optics Letters*, 22(16):1268–1270, August 1997.

[6] Bahram Javidi and Enrique Tajahuerce. Three-dimensional object recognition by use of digital holography. *Optics Letters*, 25(9):610–612, May 2000.

[7] Yann Frauel, Enrique Tajahuerce, Maria-Albertina Castro, and Bahram Javidi. Distortion-tolerant three-dimensional object recognition with digital holography. *Applied Optics*, 40(23):3887–3893, August 2001.

[8] H. John Caulfield, editor. *Handbook of Optical Holography*. Academic Press, New York, 1979.

[9] Joseph W. Goodman. *Introduction to Fourier Optics*. McGraw-Hill, New York, second edition, 1996.

[10] Levent Onural and P. D. Scott. Digital decoding of in-line holograms. *Optical Engineering*, 26(11):1124–1132, November 1987.

[11] Thomas J. Naughton, Yann Frauel, Bahram Javidi, and Enrique Tajahuerce. Compression of digital holograms for three-dimensional object reconstruction and recognition. *Applied Optics*, 41(20):4124–4132, July 2002.

[12] Thomas J. Naughton, John B. Mc Donald, and Bahram Javidi. Efficient compression of Fresnel fields for Internet transmission of three-dimensional images. *Applied Optics*, 42(23):4758–4764, August 2003.

[13] Osamu Matoba, Thomas J. Naughton, Yann Frauel, Nicolas Bertaux, and Bahram Javidi. Real-time three-dimensional object reconstruction by use of a phase-encoded digital hologram. *Applied Optics*, 41(29):6187–6192, October 2002.

[14] Thomas J. Naughton and Bahram Javidi. Compression of encrypted three-dimensional objects using digital holography. *Optical Engineering*, 43(10), October 2004.

[15] Levent Onural. Diffraction from a wavelet point of view. *Optics Letters*, 18(11):846–848, June 1993.

[16] Michael Liebling, Thierry Blu, and Michael Unser. Fresnelets: new multiresolution wavelet bases for digital holography. *IEEE Transactions on Image Processing*, 12(1):29–43, January 2003.

[17] Gregory K. Wallace. The JPEG still picture compression standard. *Communications of the ACM*, 34(4):30–44, April 1991.

[18] Rubeena Shahnaz, John F. Walkup, and Thomas F. Krile. Image compression in signal-dependent noise. *Applied Optics*, 38(26):5560–5567, September 1999.

[19] Takanori Nomura, A. Okazaki, Masashi Kameda, Yoshiharu Morimoto, and Bahram Javidi. Digital holographic data reconstruction with data compression. In Bahram Javidi and Demetri Psaltis, editors, *Algorithms and Systems for Optical Information Processing V*, Proceedings of SPIE vol. 4471, pages 235–242, San Diego, California, July 2001.

[20] Jakob Ziv and Abraham Lempel. A universal algorithm for sequential data compression. *IEEE Transactions on Information Theory*, IT-23(3):337–343, May 1977.

HAND GESTURE RECOGNITION VIA A NEW SELF-ORGANIZED NEURAL NETWORK

E. Stergiopoulou, N. Papamarkos* and A. Atsalakis

* Image Processing and Multimedia Laboratory
Department of Electrical & Computer Engineering
Democritus University of Thrace
67100 Xanthi, Greece
papamark@ee.duth.gr

Abstract
A new method for hand gesture recognition is proposed which is based on an innovative Self-Growing and Self-Organized Neural Gas (SGONG) network. Initially, the region of the hand is detected by using a colour segmentation technique that depends on a skin-colour distribution map. Then, the SGONG network is applied on the segmented hand so as to approach its topology. Based on the output grid of neurons, palm geometric characteristics are obtained which in accordance with powerful finger features allow the identification of the raised fingers. Finally, the hand gesture recognition is accomplished through a probability-based classification method.

Keywords: hand gesture recognition, colour segmentation, neural networks.

1 Introduction

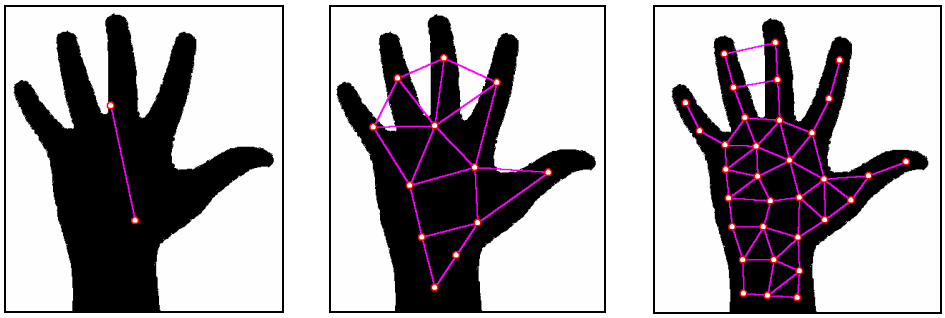
Hand gesture recognition is a promising research field in computer vision. Its most appealing application is the development of more effective and friendly interfaces for human–machine interaction, since gestures are a natural and powerful way of communication. Moreover, it can be used to teleconferencing and telemedicine, because it doesn't require any special hardware. Last but not least, it can be applied to the interpretation and the learning of the sign language.

Hand gesture recognition is a complex problem that has been dealt with many different ways. Huang et al. [1] created a system consisting of three modules: i) model based hand tracking that uses the Hausdorff distance measure to track shape–variant hand motion, ii) feature extraction by applying the scale and rotation invariant Fourier descriptor and iii) recognition by using a 3D modified Hopfield neural network (HNN). Huang et al. [2] developed also another model based recognition system that consists of three stages as well: i) feature extraction based on spatial (edge) and temporal (motion) information, ii) training that uses the principal component analysis (PCA), the hidden Markov model (HMM) and a modified Hausdorff distance and iii) recognition by applying the Viterbi algorithm. Yin et al. [3] used a RCE neural network based colour segmentation algorithm for hand segmentation, extracted edge points of fingers as points of interest and matched them based on the topological features of the hand, such as the centre of the palm. Kjeldsen et al. [4] suggested an algorithm of skin colour segmentation in the HSV colour space and used a back–propagation neural network to recognize gestures from the segmented hand images. Herpers et al. [5] used a hand segmentation algorithm that detects connected skin–tone blobs in the region of interest. A medial axis transform is applied, and finally, an analysis of the resulting image skeleton allows the gesture recognition.

In the proposed method, hand gesture recognition is divided into four main phases: the detection of the hand's region, the approximation of its topology, the extraction of its features and its identification. The detection of the hand's region is achieved by using a colour segmentation technique based on a skin colour distribution map in the YCbCr space [7-8]. The technique is reliable, since it is relatively immune to changing lightning conditions and provides good coverage of the human skin colour. It is very fast and doesn't require post–processing of the hand image. Once the hand is detected, a new Self-Growing and Self-Organized Neural Gas (SGONG) [9] network is used in order to approximate its topology. The SGONG is an innovative neural network that grows according to the hand's morphology in a very robust way. The positions of the output neurons of the SGONG network approximate the shape and the structure of the segmented hand. That is, as it can be viewed in Fig. 1(c), the grid of the output neurons takes the shape of the hand. Also, an effective algorithm is developed in

This work was partially supported by "PITHAGORAS-Development and implementation of techniques for optimal color reduction in digital images" project.

order to locate a gesture's raised fingers, which is a necessary step of the recognition process. In the final stage, suitable features are extracted that identify, regardless to the hand's slope, the raised fingers, and therefore, the corresponding gesture. Finally, the completion of the recognition process is achieved by using a probability–based classification method.



(a) (b) (c)

Figure1. Growth of the SGONG network: (a) starting point, (b) a growing stage, (c) the final output grid of neurons

The proposed gesture recognition system has been trained to identify 26 hand gestures. It has been tested by using a large number of gestures and the achieved recognition rate is satisfactory.

2 Description of the Method

The purpose of the proposed gesture recognition method is to recognize a set of 26 hand gestures. The principal assumption is that the images include exactly one hand. Furthermore, the gestures are made with the right hand, the arm is roughly vertical, the palm is facing the camera and the fingers are either raised or not. Finally, the image background is plain, uniform and its colour differs from the skin colour.

The entire method consists of the following four main stages:

- Colour Segmentation
- Application of the Self-Growing and Self-Organized Neural Gas Network
- Finger Identification
- Recognition Process

Analysis of these stages follows.

2.1 Colour Segmentation

The detection of the hand region can be achieved through colour segmentation. The aim is to classify the pixels of the input image into skin colour and non-skin colour clusters. This can be accomplished by using a thresholding technique that exploits the information of a skin colour distribution map in an appropriate colour space.

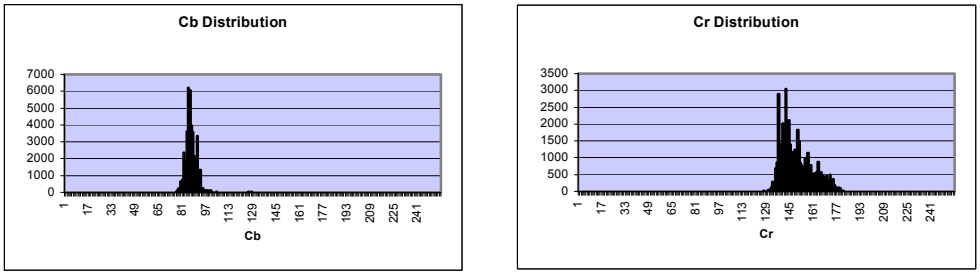
It is a fact that skin colour varies quite dramatically. First of all, it is vulnerable to changing lightning conditions that obviously affect its luminance. Moreover, it differs among people and especially among people from different ethnic groups. The perceived variance, however, is really a variance in luminance due to the fairness or the darkness of the skin. Researchers, also, claim that the skin chromaticity is the same for all races [6]. So regarding to the skin colour, luminance introduces many problems, whereas chromaticity includes the useful information. Thus, proper colour spaces for skin colour detection are those that separate luminance from chromaticity components.

The proposed colour space is the YCbCr space, where Y is the luminance and Cb, Cr the chrominance components. RGB values can be transformed to YCbCr colour space using the following equation [7-8]:

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -37.797 & -74.203 & 112 \\ 112 & -93.786 & -18.214 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1)$$

Given that the input RGB values are within range [0,1] the output values of the transformation will be [16, 235] for Y and [16, 240] for Cb and Cr. In this colour space, a distribution map of the chrominance components of skin colour was created, by using a test set of 50 images. It is found that Cb and Cr values are narrowly and consistently distributed. Particularly, the ranges of Cb and Cr values are, as

shown in Fig. 2, R_{Cb} = [80, 105] and R_{Cr} = [130, 165], respectively. These ranges were selected very strictly, in order to minimize the noise effect and maximize the possibility that the colours correspond to skin.



(a) (b)

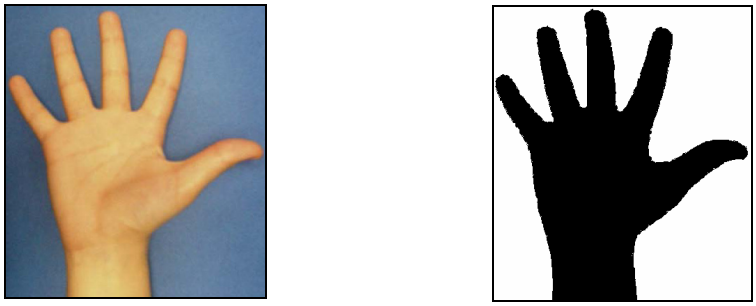
Figure 2. Distribution of: (a) Cb component, (b) Cr component

Let C_{bi} and C_{ri} be the chrominance components of the i-th pixel. If $C_{bi} \in R_{Cb}$ and $C_{ri} \in R_{Cr}$, then the pixel belongs to the hand region.

Finally, a thresholding technique completes the colour segmentation of the input image. The technique consists of the following steps.

- Calculation of the Euclidean distance between the C_{bi}, C_{ri} values and the edges of R_{Cb} and R_{Cr}, for every pixel. The distances are
- Comparison of the Euclidean differences with a proper threshold. If at least one difference is less than the threshold, then the pixel belongs to the hand region. The proper threshold's value is taken equal to 18.

The output image of the colour segmentation process is considered as binary. As illustrated in Fig. 3 the hand region, that is the region of interest, became black and the background white. The hand region is normalized to certain dimensions so as the system to be invariant of the hand's size. It is worth to underline also, that the segmentation results are very good (almost noiseless) without further processing (e.g. filtering) of the image.



(a) (b)

Figure 3. (a) Original image, (b) Segmented image

2.2 Application of the Self-Growing and Self-Organized Neural Gas Network

The next stage of the recognition process is the application of the Self Growing and Organized Neural Gas (SGONG) [9] on the segmented (binary) image.

The SGONG is an unsupervised neural classifier. It achieves clustering of the input data, so as the distance of the data items within the same class (intra-cluster variance) is small and the distance of the data items stemming from different classes (inter-cluster variance) is large. Moreover, the final number of classes is determined by the SGONG during the learning process. It is an innovative neural network that combines the advantages both of the Kohonen Self-Organized Feature Map (SOFM) and the Growing Neural Gas (GNG) neural classifiers.

The SGONG consists of two layers, i.e. the input and the output layer. It has the following main characteristics:

(a) Is faster than the Kohonen SOFM,

(b) The dimensions of the input space and the output lattice of neurons are always identical. Thus, the structure of neurons in the output layer approaches the structure of the input data,

(c) Criteria are used to ensure fast converge of the neural network. Also, these criteria permit the detection of isolated classes.

The coordinates of the output neurons are the coordinates of the classes' centers. Each neuron is described by two local parameters, related to the training ratio and to the influence by the neighbourhood neurons. Both of them decrease from a high to a lower value during a predefined local time in order to gradually minimize the neurons' ability to adapt to the input data. As it is shown in Fig. 1, the network begins with only two neurons and it inserts new neurons in order to achieve better data clustering. Its growth is based on the following criteria:

- A neuron is inserted near the one with the greatest contribution to the total classification error, only if the average length of its connections with the neighbor neurons is relatively large.
- A neuron is removed if no input vector is classified to its cluster for a predefined number of epochs.
- All neurons are classified according to their importance. The less valuable neuron is removed, only if the subsequent increase in the mean classification error is less than a predefined value.
- A neuron is removed, if it belongs to an empty class.
- The connections of the neurons are created dynamically by using the "Competitive Hebbian Learning" method.

The main characteristic of the SGONG is that both neurons and their connections approximate effectively the input data's topology. This is the exact reason for using the specific neural network in this application. Particularly, the proposed method uses the coordinates of random samples of the binary image as the input data. The network grows gradually on the black segment, i.e. the hand region and a structure of neurons and their connections is finally, created that describes effectively the hand's morphology. The output data of the network, in other words, is an array of the neurons' coordinates and an array of the neurons' connections. Based on this information important finger features are extracted.

2.3 Finger Identification

2.3.1 Determination of the Raised Fingers' Number

An essential step for the recognition is to determine the number of fingers that a gesture consists of. This is accomplished by locating the neurons that correspond to the fingertips. Observations of the structure of the output neurons' grid leads to the conclusion that fingertip neurons are connected to neighbourhood neurons by only two types of connections: i) connections that go through the background, and ii) connections that belong exclusively only to the hand region. The crucial point is that fingertip neurons use only one connection of the second type. Based on this conclusion, the determination of the number of fingers is as follows.

- Remove all the connections that go through the background.
- Find the neurons that have only one connection. These neurons are the fingertips, as indicated in Fig. 4.
- Find successively the neighbor neurons. Stop when a neuron with more than two connections is found. This is the finger's last neuron (root-neuron).
- Find the fingers' mean length (i.e. the mean fingertip and root neuron distance). If a finger's length differs significantly from the mean value then it is not considered to be a finger.

(a) (b)

Figure 4. (a) Hand image after the application of the SGONG network, (b) hand image after the location of the raised fingers

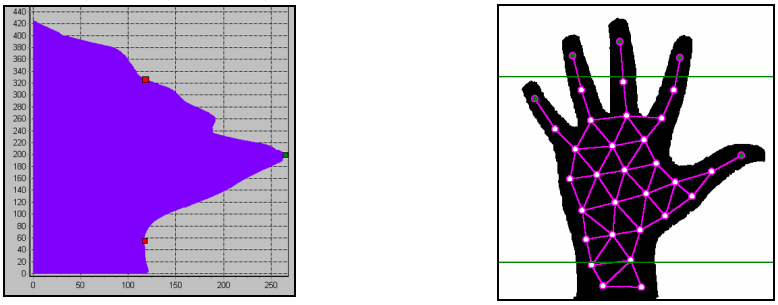
2.3.2 Extraction of Hand Shape Characteristics

Palm Region

Many images include redundant information that could reduce the accuracy of the extraction techniques and lead to false conclusions. Such an example is the presence of a part of the arm. Therefore, it is important to find the most useful hand region, which is the palm.

The algorithm of finding the palm region is based on the observation that the arm is thinner than the palm. Thus, a local minimum should appear at the horizontal projection of the binary image. The minimum defines the limits of the palm region as it is shown in Fig. 5. This procedure is as follows:

- Create the horizontal projection of the image *H[j]*:
- Find the global maximum $H\left[j^{\max}\right]$ and the local minima $H\left[j_i^{\min}\right]$ of $H[j]$.



(a) (b)

Figure 5. (a) Horizontal projection, (b) Palm region

- Calculate the slope of the lines segments connecting the global maximum and the local minima, which satisfy the condition $j_i^{\min} < j^{\max}$. The minimum j_{lower} that corresponds to the greatest of these slopes defines the lower limit of the palm region, only if its distance from the maximum is greater than a threshold value equal to ImageHeight/6.
- The point that defines the upper limit of the palm region is denoted as j_{upper} and is obtained by the following relation:

$$H\left[j_{upper}\right] \le H\left[j_{lower}\right] \quad \text{and} \quad j_{upper} > j^{\max} > j_{lower} \tag{2}$$

Palm Centre

The coordinates of the centre of the palm are taken equal to the mean values of the coordinates of the neurons that belong to the palm region.

Hand Slope

Despite of the roughly vertical direction of the arm, the slope of the hand varies. This fact should be taken under consideration because it affects the accuracy of the finger features, and consequently, the efficiency of the identification process. The recognition results depend greatly on the correct calculation of the hand slope.

The hand slope can be estimated by the angle of the left side of the palm, as it can be viewed in Fig. 6(a). The technique consists of the following steps:

- Find the neuron N_{Left}, which belongs to the palm region and has the smallest horizontal coordinate.
- Obtain the set of palm neurons N_{set} that belong to the left boundary of the neurons grid. To do this, and for each neuron, starting from the N_{Left}, we obtain the neighborhood neuron which has, simultaneously, the smallest vertical and horizontal coordinates.
- The first and the final neurons of the set N_{set} define the hand slope line (HSL) which angle with the horizontal axis is taken equal to the hand's slope.

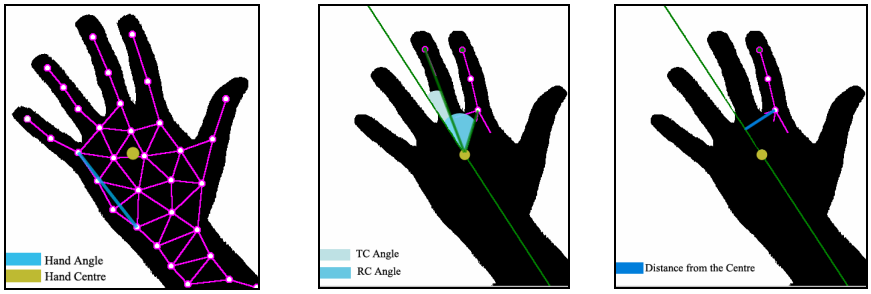
The hand slope is considered as a reference angle and is used in order to improve the feature extraction techniques.

2.3.3 Extraction of Finger Features

Finger Angles

A geometric feature that individualizes the fingers is their, relative to the hand slope, angles. As it is illustrated in Fig. 6(b), we extract two finger angles.

- RC Angle. It is an angle formed by the HSL and the line that joints the root neuron and the hand center. It is used directly for the finger identification process.
- TC Angle. It is an angle formed by the HSL and the line that joints the fingertip neuron and the hand center. This angle provides the most discrete values for each finger and thus is valuable for the recognition.



(a) (b) (c)

Figure 6. (a) Hand slope and centre, (b) Fingers' angles, (c) Distance from the centre

Distance from the Palm Centre

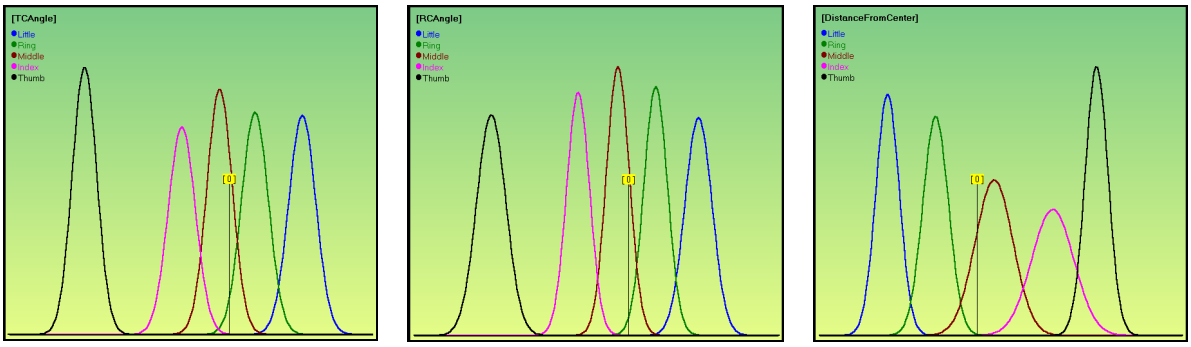
A powerful feature for the identification process is the vertical distance of the finger's root neuron from the line passing through the palm centre and having the same slope as the HSL. An example is illustrated in Fig. 6(c).

3 Recognition Process

The recognition process is actually a choice of the most possible gesture. It is based on a classification process of the raised fingers into five classes (thumb, index, middle, ring, little) according to their features. The classification depends on the probabilities of a finger to belong to the above classes. The probabilities derive from the features distributions. Therefore, the recognition process consists of two stages: the off–line creation of the features distributions and the probability based classification.

3.1 Features Distributions

The finger features are naturally occurring features, thus a Gaussian distribution can model them. Their distributions are created by using a test set of 100 images from different people.



(a) (b) (c)

Figure 7. Features distributions (a) TC Angle, (b) RC Angle, (c) Distance from the centre

If f_i is the i-th feature ($i \in [1, 3]$), then its Gaussian distributions for every class c_j ($j \in [1, 5]$) are given by the relation:

$$p_{f_i}^{c_j}(x) = \frac{e^{-\frac{\left(x - m_{p_i}^{c_j}\right)^2}{2\sigma^2}}}{\sigma\sqrt{2\pi}}, \qquad (3)$$

where, $j = 1,...,5$, $m_{f_i}^{cj}$ is the mean value and $\sigma_{f_i}^{cj}$ the standard deviation of the f_i feature of the c_j class. . The Gaussian distributions of the above features are shown in Fig. 7. As it can be observed from the distributions, the five classes are well defined and are well discriminated.

3.2 Classification

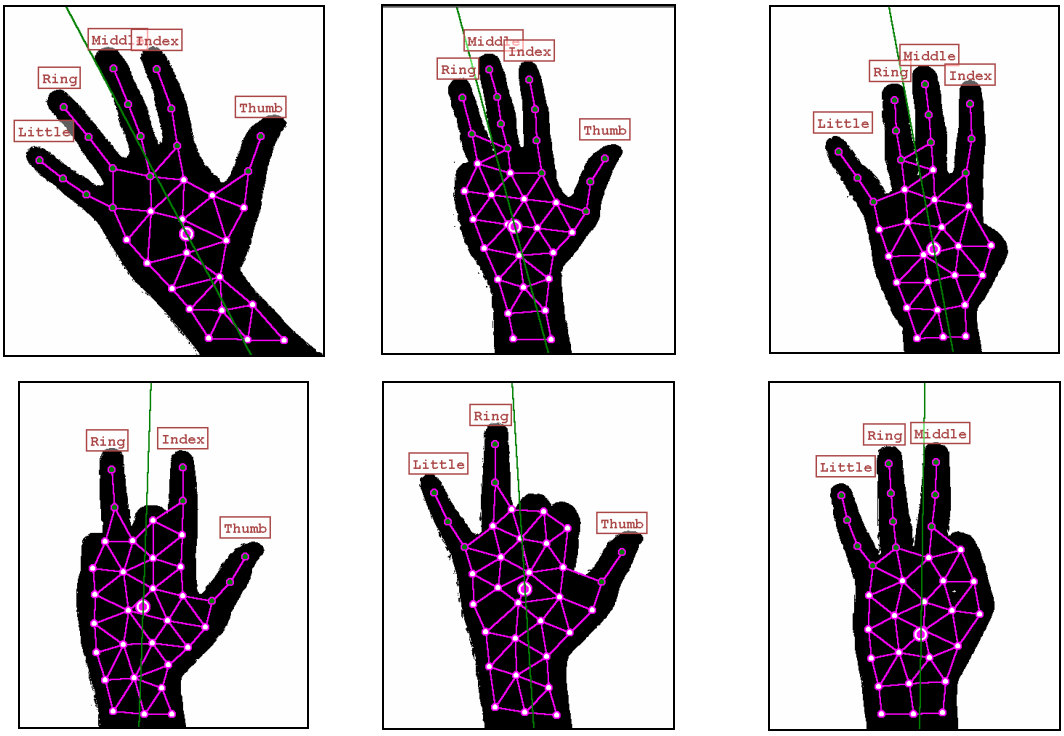
The first step of the classification process is the calculation of the probabilities RPc_j of a raised finger to belong to each one of the five classes. Let x_0 be the value of the i-th feature f_i. Calculate the probability $p_{f_i}^{cj}(x_0)$ for $i \in [1, 3]$ and $j \in [1, 5]$. The requested probability is the sum of the probabilities of all the features for each class

$$RPc_j = \sum_{i=1}^{3} p_{f_i}^{c_j} \qquad (4)$$

This process is repeated for every raised finger.
Knowing the number of the raised fingers, one can define the possible gestures that can be created. For each one of these possible gestures the probability score is calculated, i.e. the sum of the gesture' s each raised finger to belong to each one of the classes. Finally, the gesture is recognized as the one with the higher probability score.

4 Experimental Results

The proposed hand gesture recognition system, which was implemented in DELPHI, was tested with 158 test hand images 1580 times. It is trained to recognize up to 26 gestures. The recognition rate, under the conditions described above, is 90.45%. Fig. 8 illustrates recognition examples.



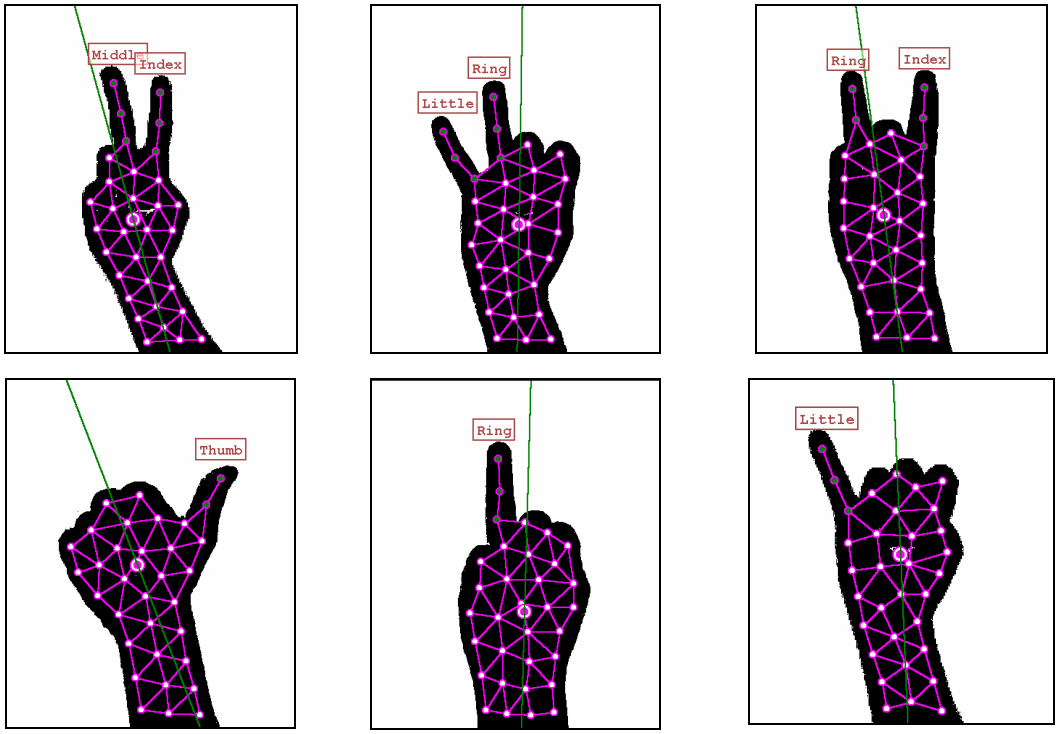


Figure 8. Gesture recognition examples.

5 Conclusions

This paper introduces a new technique for hand gesture recognition. It is based on a colour segmentation technique for the detection of the hand region and on the use of the Self-Growing and Self-Organized Neural Gas network (SGONG) for the approximation of the hand's topology. The identification of the raised fingers, which depends on hand shape characteristics and fingers' features, is invariant of the hand's slope. Finally, the recognition process is completed by a probability-based classification with very high rates of success.

6 References

[1] Huang Chung-Lin, Huang Wen-Yi (1998). Sign language recognition using model-based tracking and a 3D Hopfield neural network. Machine Vision and Applications, 10:292-307. Springer-Verlag.

[2] Huang Chung-Lin, Jeng Sheng-Hung (2001). A model-based hand gesture recognition system. *Machine Vision and Applications*, 12:243-258. Springer-Verlag.

[3] Yin Xiaoming, Xie Ming (2003). Estimation of the fundamental matrix from uncalibrated stereo hand images for 3D hand gesture recognition. *Pattern Recognition*, 36:567-584. Pergamon.

[4] Kjeldssen Rick, Kender John (1996). Finding skin in colour images. *Proceedings IEEE International Conference on automated face and gesture recognition*, 184-188.

[5] Herpers R., Derpanis K., MacLean W.J., Verghese G., Jenkin M., Milios E., Jepson A., Tsotsos J.K. (2001). SAVI: an actively controlled teleconferencing system. *Image snd Vision Computing*, 19:793-804. Elsevier.

[6] O' Mara David T. J. (2002). Automated Facial Metrology. Ph.D. Thesis, University of Western Australia, Department of Computer Science and Software Engineering.

[7] Chai Douglas, Ngan N. King (1999). Face segmentation using skin color map in videophone applications. *IEEE Transactions on Circuits and Systems for Video Technology,* 551-564.

[8] Chai Douglas, Ngan N. King (Apr. 1998). Locating facial region of a head–and–shoulders color image. Third IEEE International Conference on Automatic Face and Gesture Recognition, Nara, Japan, 124-129 .

[9] Atsalakis Antonis (2004). Colour Reduction in Digital Images. Ph.D. Thesis, Democritus University of Thrace, Department of Electrical and Computer Engineering.

Irish Sign Language Recognition Using PCA, Multi-scale Theory, and Discrete Hidden Markov Models

Wu Hai
Alistair Sutherland
School of Computing
Dublin City University
Glasnevin, Dublin 9, Ireland
wuhai@computing.dcu.ie

Abstract:
Hidden Markov Models (HMMs) have attracted increasing attention on dynamic gesture recognition. Different researchers use various features as the input to HMMs, hence the differences between their systems. Most of them make use of simple features which severely limit the system ability to deal with complex gestures. However, using complex features will increase the workload of the system, and thus slow down its real-time performance. This paper presents a novel method where a hand configuration extractor is constructed based on Hierarchical Principal Component Analysis (HiPCA), which can extract even very sophisticated hand shapes given a sequence of video. The hand shapes together with the trajectory of the hand centroid are then input into the Discrete Hidden Markov Models (DHMMs) to recognise dynamic gestures in the video sequence. Our experiments show that the method can achieve high performance in terms of both recognition rate and speed.

Keywords: Hierarchical PCA, DHMMs, Decision tree.

1. Introduction

Sign Language (SL) has been used by the Deaf people from all over the world. Most of the countries have their own SL which is different from the others. Unlike the spoken-language, where English is the major language in modern society in terms of science and business, there is not a dominant SL, which makes it hard for the Deaf from different countries to communicate. The ability to recognise gestures (static/dynamic) using computers will help to overcome this problem.

In recent years HMMs have shown great potential in the area of dynamic gesture recognition comparing to the other techniques and have attracted increasing attention. Since Starner and Pentland [1] applied them to the recognition of ASL sentences many other researchers have employed them in their systems, such as Vogler and Metaxas [2] and Wilson and Bobick [3]. In fact, HMMs are a rather general mathematical model that are able to handle the temporal variability of a dynamic process. Different researchers use various features as the input, hence the differences between their systems. Many of them only use basic geometric parameters of the hands or other simple features. For instance, Starner and Pentland use a simple feature set to describe the hand shape which consists of the x and y position of each hand, angle of axis of least interia, and eccentricity of bounding ellipse. Lee and Kim [4] took advantage of the hand centroid, divided the 2D plane in the image into 16 directions, and used the direction of the movement of the hand centre as the feature vector. Naturally, we can imagine that simple feature vectors could cause problems because different complex gestures could have very similar simple features. In this case, when the size of the vocabulary increases, the coincidence between features will become more severe. However, using complex features will increase the workload of the system,

and thus slow down the real-time performance. In this paper, we present a hand configuration extractor to deal with even very sophisticated hand shapes given a sequence of video, which gives it robustness against hand shape changing.

Our interest is in developing a hand gesture recognition system with a single camera that is able to run in frame rate without the aid of any other special hardware except a normal desktop computer. In the remainder of the paper, we describe the problem and our basic idea on how to solve the problem in section 2, then we introduce the details in our system in section 3. We evaluate our system using 35 dynamic gestures in section 4. Finally we summarise.

2. Problem Description

Irish Sign Language (ISL) is composed of two types of gestures: static gestures and dynamic gestures, for example, ISL contains 26 gestures corresponding to 26 English alphabet. Of which 23 are static shapes, while the rest have to be expressed by dynamic gestures. In this paper, we will concentrate on the task of dynamic gesture recognition. Let's start from a few dynamic gesture examples. Figure 1 gives three examples taken from ISL.

When a tutor teaches these gestures, he would use the following sentences to describe them [5]:

Gesture 1: Hold the hand in the "T" position at chest level, then move it to the right changing to the "V" position.

Gesture 2: Hold the hand in the "A" position beside left cheek, then move it to the right.

Gesture 3: Hold the hand in the "L" position, then swing it to the right.

("T", "V", "A", and "L" positions are in terms of the static shapes in ISL)

The above teaching method shows three key issues when describing a dynamic gesture:

Figure 1: Three dynamic gesture examples from Irish Sign Language. Pictures are taken from [5].

The hand configuration is important. For example: "T", "V", "A", or "L" position.

The global movement of the hand is important. For example: move it to the right, swing it to the right, and so on.

The relative position of the hand against other parts of body is important. For example: chest level, left cheek, and so on.

If we can devise a system that can fully integrate these points together, we should be able to achieve good results. Unfortunately, the third point is related to segmentation and recognition of other parts of body, which is beyond our current research. Thus, our current system attempts to deal with the first and second point. First, the system should be able to recognise the static hand shapes and deal with the change in static shapes during the performing of dynamic gestures. Second, the system should also have the ability to recognise the whole hand movement.

We notice it is harder to handle the first point than the second. People describe the hand movement in a rough way, such as move your left hand to the right. When designing a dynamic gesture, the designer will not

describe one gesture as "move your hand to the 63 degree direction up. Be careful, don't do it along the 70 degree, it means something else". As opposed to this, human hands show much more variety in term of its configurations, or shapes. For example, there are about 40 different hand configurations in ISL. Furthermore, when these 40 hand configurations are observed from different angles, more appearances will appear.

3. Solution Description

Based on the above thoughts, we designed our ISL recognition system which consists of three major components. The first one is a hierarchical decision tree based on the combination of multi-scale theory and Principal Component Analysis (PCA). Given a sequence of video containing ISL, this tree can extract the intermediate hand shapes fast and reliably. This component deals with the change of local hand shapes. In the meanwhile, the second component creates direction codes to record the movement of the hand centroid in the video sequence, which represents the hand's movement as a whole object. This component deals with the hand global movement. The third component is a recogniser based on the Discrete Hidden Markov Models (DHMMs) to deal with the dynamic characteristic in ISL.
In our system, the video camera is set up in front of the user. To segment the hand from the rest of the image the user wears a coloured glove whose colour is not likely to appear in the background of the streamed video images. A standard colour segmentation method is applied. We then compute the positions of the hand centroid from the segmented images whose trajectory represents the hand global movement and will be used to compute the direction code. At this stage, the area of the hand within the image varies greatly when the hand moves with respect to the camera. Thus we scale the segmented hand by area to a 32×32 grey-level centred on the centroid. Furthermore, given a 32×32 image I, its image vector f is constructed by concatenating the image pixels row by row. f will be use as input to the first component, i.e. the hierarchical decision, to compute the local hand shape. Now we describe the three components in details in the following sections.

3.1. Hierarchical Principal Component Analysis

As stated above, the first component constructs a hierarchical decision tree by utilising Hierarchical PCA (HiPCA), which combines the PCA with the multi-scale theory to build a hierarchical decision tree. Before we discuss its details, we first briefly review the multi-scale theory.
Given an image I, if we convolve it with a Gaussian kernel, a smoother version of it is obtained. Varying the blurring factor σ of the Gaussian kernel, the image I is then represented by a family of smoother versions of I: I(σ), where I(0) corresponds to the original image, and as the value of σ increases, more and more details in the original image are eroded an no spurious structures will be created, see figure 2. Formally, this procedure can be formalised by:

$$I(x, y, \sigma) = I_0 * G(x, y, \sigma) \tag{1}$$

where * denotes a convolution, I_0 represents the original image, σ is the scale parameter which is always non-negative, and G(x,y,σ) stands for a two-dimensional Gaussian kernel defined as:

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{2}$$

Figure 2. Images under different blurring factors: (1) is the original image, i.e. $\sigma = 0$. From (1) to (4), the blurring factors are 0, 0.5, 1.0, 1.5 respectively.

Hence, Given a set of gestures, if we blur them at different levels, different details will appear so that the same training set can be divided into different groups.
To utilise the above thoughts in practice, we give a detailed description of the algorithm: given a set of training image vectors X which is computed from the training video using the method introduced previously,

1. Every sample in the training set X is convolved with a two-dimensional Gaussian kernel whose blurring factor is σ, G(x,y,σ):

$$\mathbf{X}' = \{\mathbf{f}_1', \mathbf{f}_2', ..., \mathbf{f}_N'\} \quad (3)$$

where $\mathbf{f}_i'$ is given by:

$$\mathbf{f}_i' = \mathbf{f}_i * G(x, y, \sigma) \quad (1<i<N) \quad (4)$$

where * defines a convolution. This step blurs the differences between the images and reduces their separation in the PC space. This reduces the number of eigenvectors needed to describe the data as well.

2. A PC space of is computed from $\mathbf{X}'$:
   1). Computing the covariance matrix of $\mathbf{X}'$.
   2). A PC space is then computed whose basis are the eigenvectors of the covariance matrix of $\mathbf{X}'$.
   3). The dimensionality of the PC space is decided by retaining the first few PCs so that at least 95% of energy is retained.
3. The standard k-means algorithm is then applied to the data in the PC space, dividing them into C clusters according to what type of tree is wanted, i.e. for a binary tree C=2, for a quad-tree, C=4, and so on. The original training set X is then split into C groups: $X_1, X_2, \ldots, X_C$.
4. For each of the C clusters, check if the stop criterion is satisfied. If it is, mark it as a leaf, and if all the clusters at the current level are leaves, stop the splitting process. Otherwise, for each X_i (1<i<C), repeat step 1 to 4 with a smaller scale parameter σ' (σ'<σ).

The above procedure first blurring differences between images and reducing their separation in feature space by convolving all members of the training data with a Gaussian kernel, and then dividing the data in this space into clusters in the PC space computed from the convolved data. Then for each cluster, the same procedure is repeated but with a smaller σ so that more details in the gestures can be seen. We thus produce a hierarchical decision tree where each level of the tree represents a different degree of blurring. The decision tree is based on the multi-scale theory and PCA, hence we call the combination Hierarchical PCA (HiPCA). The search time is then proportional to the depth of the tree, which makes it possible to search hundreds of gestures with very little computational cost. The final output of the process is a decision tree, where each node is a PC space. See Figure 3.

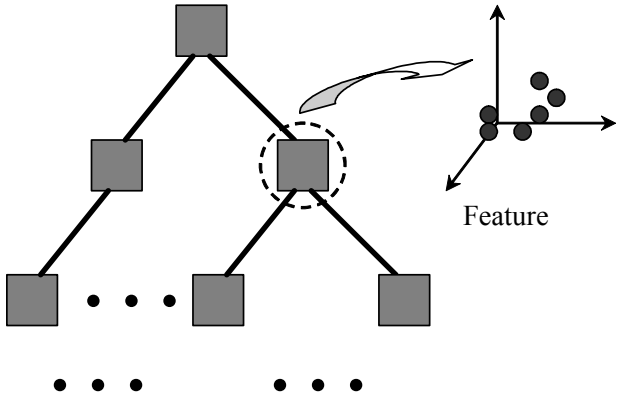


Figure 3: Illustration of a decision tree.

An important parameter during the construction of a decision tree is the termination condition. Currently we choose the data variance in the PC space as the termination condition. When the variance of image projections

in the PC space reduces to a certain level, 0.5 for example, the node will be marked as a leaf, and no further splitting operation will be done on this node. When all the nodes on the current level are leaves, the construction is stopped and the learning process is finished.
Obviously, the tree can separate a set of training images into many small groups according to their similarities. In other words, all the images in one leaf should contain similar gestures: not necessarily the same gesture, but similar in their two-dimensional images. If the training set is a sequence of videos which contains many different dynamic gestures, the leaves in the tree would contain different hand shapes, or configurations, that appeared in the videos. No matter how complex the hand configurations are, they will always be "extracted" from the training images. Hence, we call the tree a hand configuration extractor. We use gesture 1 (see Figure 1) as an example. 60 examples with the length varying from 5 frames to 12 frames are acquired continuously. We then compute a hierarchical decision tree based on these data. In total seven leaves are extracted out under the termination condition that the data variance in each leaf is smaller than 0.5. Figure 4 shows the mean images of these leaves. By looking at the seven images in the figure, one can have a basic idea of the dynamic procedure of gesture 1: starting with the shape of "T", then changing into the shape of "V". We label the leaves in the extractor by integers. Given a sequence of video, we input every image into the tree, which is then classified into one of the leaves. The number of the corresponding leaf hence can reflect an even very complex hand configuration. The change of the local hand configuration is then represented by a sequence of integers, for example, <34, 5, 4, 4, 93, 99, 30>.

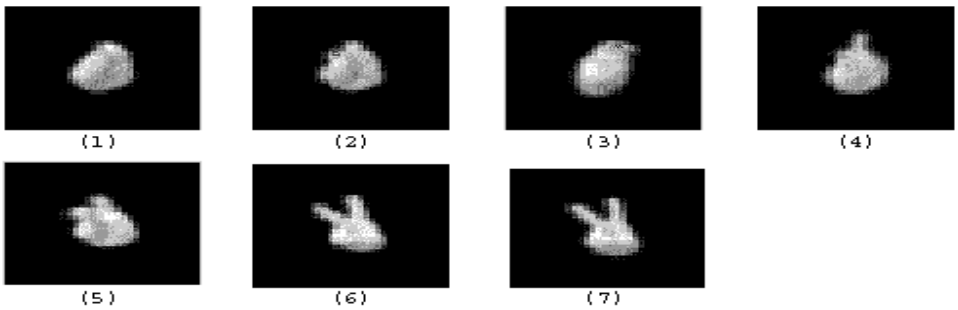


Figure 4: Mean images of the decision tree made from gesture 1 illustrated in Figure 1. The seven images show how one performs the gesture: starting with a shape of a fist in (1) and (2), hold hand in "T" position in (3), and change into "V" position. (4) and (5) are the intermediate steps of the change.

3.2. Direction Code

The hand configuration extractor only deals with the change of the local hand configuration, to record the hand global movement, we need to build the direction code.

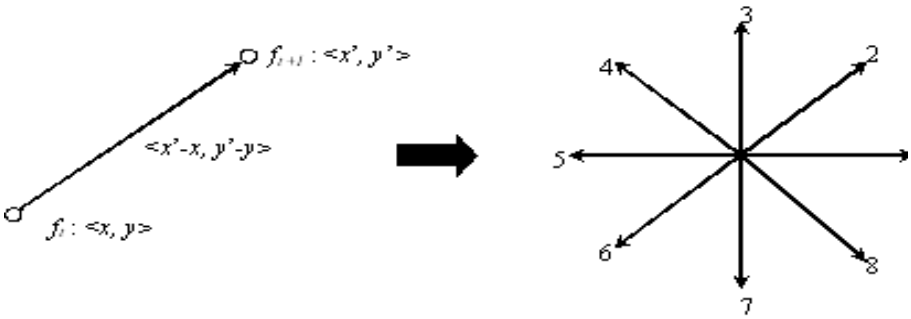


Figure 5: Building the direction code.

Given a sequence of images, we define the direction code based on every two consecutive images f_t and f_{t+1}. Assume the co-ordinate of the hand centroid in f_t is <x, y>, and in f_{t+1} <x', y'>. A vector is computed as <x'-x, y'-y>. Then we translate this vector into the direction code I_D that is one of 8 directions. See Figure 5.

3.3. Discrete Hidden Markov Models

To deal with the dynamic characteristic in ISL, we take advantage of DHMMs, which is the discrete format of Hidden Markov Models (HMMs). Many researchers have presented their systems based on HMMs. Most of them employed Continuous HMMs (CHMMs) or semi-continuous HMMs [1, 3], and use some geometric parameters of the hands as the input features[1, 4]. This brings a couple of disadvantages. On one hand, simple features can only separate gestures from a very small vocabulary, as when the size of the vocabulary grows, the coincidence between features will become more severe. On the other hand, comparing to DHMMs, CHMMs are slower and more difficult to train.
To achieve both low computational cost and robustness, we have used DHMMs in our system, and have developed our own feature vector, which is completely different from the simple features used in work by others.

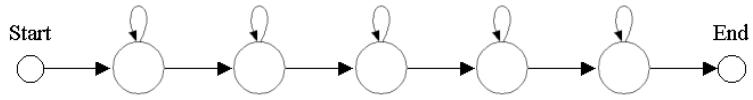


Figure 6: A sample DHMM topology which has been used in our evaluation.

Figure 6 shows the topology of the DHMM that has been used in our evaluation. It contains seven states including the *start* and *end* sates.
The input features we chosen to reflect the hand dynamics consists of two parts. The first part is used to handle the complex hand shapes which might happen in the dynamic gestures. It is obtained from the output of the hand configuration extractor. The second part is the direction code which reflects the global movement of the hand. A dynamic gesture is thus represented by a sequence of two-dimensional feature vectors. For example, one gesture might be represented by: < [366, 7], [509, 6], [509, 5], [509, 4], [359, 4], [148, 3], [23,3], [23,2] >. While another gesture might be represented by: <[355,1], [509,6], [441, 5], [441, 5], [441, 5]>.

4. Evaluation

Although the system has the potential on very large vocabulary, in the current stage, we only show the recognition result of 35 dynamic gesture in ISL. Our experiment was based on DELL OptiPlex GX1 P2 350 MHz and Creative Webcam 3. The training images were acquired under normal office illumination conditions. First, for each of the 35 gestures, we grab 60 samples, i.e. 2100 in total. Using these samples, we train the hand configuration extractor. The selection of blurring factors were all determined by trail and error and decreased in logarithmic order at different layers of the tree since it reflects the changing structure in the images [6]. The equation is given below [6]:

$$\sigma = \varepsilon * \exp(k / c) \tag{5}$$

where ε and c are constants. In practice, we use the values ε = 0.01 and c =20.
We first used the training set to construct the hand configuration extractor and to compute the direction codes. Once this had been done, the same set of training samples were fed into it whose output was then combined with the direction codes to build a sequence of 2-dimensional feature vectors. We then trained the DHMM recogniser with the feature vectors. The recogniser contains 35 individual DHMMs. That is, one for each gesture. The training of the DHMM recogniser was finished by using the HTK package.
For a fair test, we grabbed another similar group of samples, i.e. 60 for each of the 35 gestures, which were never used for any portion of the training. For every group, we fed it into the hand configuration extractor and

computed its direction codes in order to construct the feature vectors. The feature vectors were then sent into the DHMM recogniser to find out the possible existing dynamic gestures. The recognition rate is shown in Figure 7. No grammar was used.

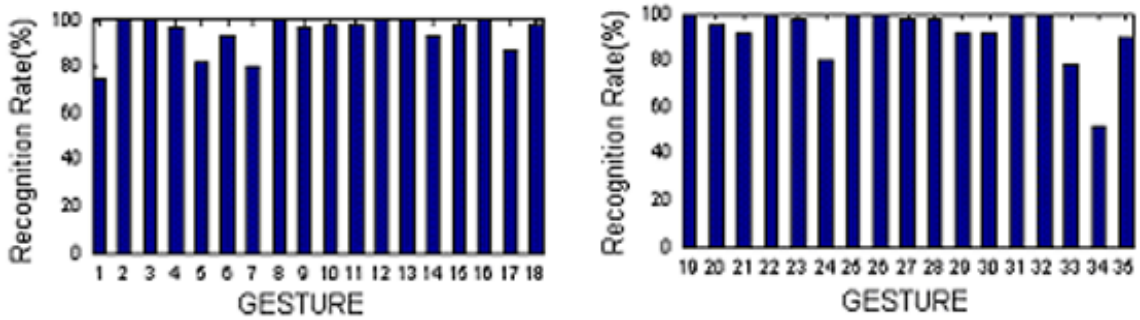


Figure 7: The bar chart of the evaluation results of 35 dynamic gestures

- **Error Analysis**

Errors are mainly caused by three different reasons:

1. The first type is caused by the visual similarity in hand shapes between the gestures. For example, most of the errors in gesture 1 were misclassifed into gesture 27. The two gestures have the same global hand movements. The only difference is in the local hand configurations. Gesture 1 holds the hand in the shape of "D" while Gesture 2 holds the hand in the shape of "W", illustrated in Figure 8. From a specific viewpoint, i.e. side view, the two shapes are similar. The ambiguity is essentially caused by using a single camera. No good solution is available in the current situation.

Figure 8: Two hand shapes in ISL. The left one corresponds to "D" and the right one "W".

2. The second type is caused by the visual similarity in global hand movements between the gestures. For example, most of the errors in gesture 7 were misclassified into gesture 34, and vice versa. Both gestures have the same change in terms of the hand configurations. The difference is that gesture 7 is moving the hand backwards as well as to the right, while gesture 34 only needs to move the hand to the right. This error shows the weakness of our system on handling 3D hand movements. Since no depth information was considered during the recognition, the backward movement sometimes was treated the same as moving to the right. The information on hand area is not much help either, because the backward movement is not very significant compared to the distance from the hand to the camera. 3D depth recognition is another open problem for our system, and more research has to be done in future.

3. The third type is caused by the lack of information of the relative position between the hand and other parts of body. For example, the error in gesture 33 is a different type: most of the errors were misclassifed into gesture 1. Both gestures hold the hand in the same hand shape of "D", and both of them perform similar hand movements: from the upright to down left. The difference between them is that the hand movement is a curve and is performed at chest level, while gesture 33 should perform as a straight line at face level. It could be improved effectively if the system had the ability to recognise the relative positions of the hand against other parts of body.

We also notice the overall recognition rates are high. Although partly it is because of the small vocabulary, this preliminary result does show our approach's potential.

5. Discussion

We presented a novel appearance-based system that is able to recognise dynamic gestures using HiPCA and DHMMs. It runs fast even on a cheap machine without help of any other special hardware except a webcam. This is because we employed a hierarchical tree to accomplish the search procedure. On the one hand, it reduces the search time significantly: from $O(n)$ to $O(\log_2 n)$. As the size of vocabulary gets larger, the reduction becomes even more significant. On the other hand, it reduces the dimension of the input feature vector while still remain the robustness to handle complex hand shapes. This allows us to use DHMMs instead of CHMMs, and hence speed up the recognition.The construction of a bigger vocabulary is now in progress.

Reference:

[1] T. Starner, "Visual recognition of American Sign Language Using Hidden Markov Models", Master's thesis, MIT Media Lab, USA, 1995.

[2] C. Vogler, D. Metaxas, "ASL Recognition Based on a Coupling Between HMMs and 3D Motion Analysis", in *Proceedings of IEEE International Conference on Computer Vision*, Bombay, India, 1998, 363–369.

[3] A. Wilson, A. Bobick, "Recognition and interpretation of parametric gesture", in *Proceedings of IEEE International Conference on Computer Vision*, Bombay, India, 1998, 329-336

[4] H. K. Lee, J. H. Kim, "An HMM-based Threshold Model Approach for Gesture Recognition", in *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 21(10), 1999, 961-973.

[5] Sign Language Association of Ireland, "Sign on: Basic signs used by Irish Deaf people", Dublin, 1995.

[6] Y. Leung, J. S. Zhang, and Z. B. Xu, "Clustering by Scale-space filtering", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, No. 12, 2000, 1396-1410.

GENERATING A MAPPING FUNCTION FROM ONE EXPRESSION TO ANOTHER USING A STATISTICAL MODEL OF FACIAL TEXTURE

John Ghent
Computer Science Department,
NUI Maynooth, Ireland
email: jghent@cs.may.ie

John McDonald
Computer Science Department
NUI Maynooth, Ireland
email: johnmcd@cs.may.ie

Abstract

We demonstrate a novel method of generating a mapping function which takes an image of a neutral face to an image of the same subject depicting an alternative expression. It is proposed that this mapping function can be used to automatically generate facial expressions from still images of never seen before faces. This technique draws on the work of Ekman's [8] Facial Action Coding System (FACS), which provides an anatomical basis for measuring facial movement. We use the FACS to generate a *Facial Expression Texture Model* (FETM), which is used in conjunction with several *Artificial Neural Networks* (ANN) to develop a mapping function. We describe this method in detail and provide results which demonstrate its effectiveness.
Keywords*: Facial expression synthesis, Facial Expression Texture Model (FETM), Facial Action Coding System (FACS), function approximation*

1 Introduction

The central goal of this paper is to describe the development of a mapping function which manipulates a neutral image of a subject to accurately display a desired expression. This paper builds upon the work described in [13] where a mapping function was created that manipulated contours depicting facial shape, this paper extends this idea by manipulating the texture of the face.

The development of this mapping function involves a comprehensive understanding of expression. Facial expressions have been studied by cognitive psychologists [5, 25], social psychologists [10], neurophysiologists [24], computer scientists [8] and cognitive scientists [6]. The model of facial expression described in this paper is Ekman's [10] Facial Action Coding System (FACS). This method of studying facial expressions and emotions depicted by facial expressions is based on an anatomical analysis of facial actions. A movement of one or more muscles of the face is known as an action unit (AU). All expressions can be described using one, or a combination of the AU's described by Ekman.

We achieve expression synthesis by building a statistical model of the AU in question from a number of subjects showing that expression in a training set. The change in texture of each face in the training phase is analysed and used to derive a mapping function, which takes their neutral face to one depicting the new expression.

To decrease the dimensionality of the mapping the variance in texture of each face in the training set is analysed using *Principal Component Analysis* (PCA). This approach can model a large amount of the variance in the training set by using only a few modes of variation or principal components. This representation of expression is known as the expression space. We use the expression space in conjunction with *Feedforward Heteroassociative Memory Networks* (FHMN) and *Radial Basis Functions* (RBF) to generate a subject independent mapping function, the results of which are presented in this paper.

2 Measuring expression

Few studies have measured how the face moves as an expression forms [19, 12, 10, 2, 29]. The central reason for this is the fact that research focused on facial expressions is limited due to the lack of adequate techniques for measuring the face. Knowledge of the muscles of the face allows us to characterise exactly what is happening as an expression is emerging. Since everyone's face is different it is difficult to characterise an expression any other way. For this reason a thorough understanding of the face is required prior to devising a scheme for the characterisation and measurement of facial expression.

According to Faigin [11], of the twenty-six muscles that move the face, only eleven are responsible for facial expressions. Although this description by Faigin provides a good basis for understanding the anatomy of facial expressions it does not provide an insight as to which muscles work together to create certain expressions.

The *Facial Action Coding System* (FACS) provides a method for studying facial expressions and emotions depicted by facial expressions based on an anatomical analysis of facial actions. A movement of one or more muscles of the face is known as an action unit (AU). Sometimes it is difficult to distinguish if one or a set of muscles is accountable for a facial movement. it is for this reason that the term action unit is used. All expressions can be described using the individual AU's described by Ekman or a combination of the AU's.

2.1 Facial Expression Texture Model (FETM)

To calculate the *Facial Expression Texture Model* (FETM) we warp all images to the mean shape. This is achieved using Delaunay triangulation to segment the mean shape into 214 separate triangles using 122 landmark points. We apply the affine transformation to the pixels within each triangle [8]. Suppose $\mathbf{x_1}$, $\mathbf{x_2}$ and $\mathbf{x_3}$ are three corners of a triangle. Any internal pixel can be written as

$$\mathbf{x} = \mathbf{x_1} + \beta(\mathbf{x_2} - \mathbf{x_1}) + \gamma(\mathbf{x_3} - \mathbf{x_1}) = \alpha\mathbf{x_1} + \beta\mathbf{x_2} + \gamma\mathbf{x_3} \tag{1}$$

where $\alpha = 1 - (\beta + \gamma)$ and $\alpha + \beta + \gamma = 1$. For $\mathbf{x}$ to be inside a triangle, $0 \leq \alpha, \beta, \gamma \leq 1$. Under the affine transformation, this pixel maps to

$$\mathbf{y} = \mathbf{f}(\mathbf{x}) = \alpha\mathbf{y_1} + \beta\mathbf{y_2} + \gamma\mathbf{y_3} \tag{2}$$

Each image is then represented as a vector.

Definition $\mathbf{x}_i^{k_j}$ *Let* $\mathbf{k}$ *be a vector of AU's where* $\mathbf{k} = \{k_0, k_1, k_2 ... k_{m-1}\}$ *and* m *is the number of AU's. Then* $\mathbf{x}_i^{k_j}$ *is a vector representing an image of subject* i *showing AU* k_j*.*

We use PCA to analyse how the vectors change with respect to each other. Before any significant analysis can be done on the shape of the faces, the mean must be computed. This is done using the equation below:

$$\overline{\mathbf{x}} = \frac{1}{Nm} \sum_{i=1}^{N} \sum_{j=0}^{m-1} \mathbf{x}_i^{k_j} \tag{3}$$

where $\overline{\mathbf{x}}$ is the mean image vector of every subject i portraying every AU k_j and N are the number of subjects in the training set . The difference vector is then calculated using

$$\delta\mathbf{x}_i^{k_j} = \mathbf{x}_i^{k_j} - \overline{\mathbf{x}} \tag{4}$$

where $\delta\mathbf{x}_i^{k_j}$ is the difference between $\mathbf{x}_i^{k_j}$ and the in the mean vector $\overline{\mathbf{x}}$. The covariance matrix is then calculated. In the experiments in this paper the n x n covariance matrix is very large, where $n = 65025$.

For this reason the eigenvectors and eigenvalues are calculated from a smaller s x s matrix derived from the data, where $s = N$ x m. Let $D = (\delta \mathbf{x}_1^{k_0} \cdots \delta \mathbf{x}_N^{k_m})$. The covariance matrix can be represented as

$$S = \frac{1}{s} D D^T \tag{5}$$

Let T be the s x s matrix

$$T = \frac{1}{s} D^T D \tag{6}$$

Let e_i be the s eigenvectors of T with eigenvalues λ_i. The s vectors De_i are all eigenvectors of S with eigenvalues λ_i. All remaining eigenvectors of S have zero eigenvalues. Texture parameters for $\mathbf{x}_i^{k_j}$ can be extracted and reconstructed using a similar technique used with the *Facial Expression Shape Model* (FESM) [13, 16].

3 Function approximation

ANNs have proven to be successful in many practical problems. It has been shown that ANNs can recognise handwritten characters [21], spoken words [20] and more relevantly human faces [9]. In this section we address the problem of facial expression synthesis and discuss ANNs that can be used for this task in conjunction with the FETM.

A Feedforward Heteroassociative Memory Network (FHMN) can be used to compute a mapping from x to y. This is a one-layer network that stores patterns and is the simplest type of network we consider.The Neural Network is trained by using the n principal components that represent a neutral face as input and the n principal components that represent a face depicting a specific expression as output. In this manner a mapping function is learned which maps the texture of a neutral face to that of a specific expression.

Radial Basis Function (RBF) networks are a form of ANN that are closely related to what is known as *distance-weighted regression*. The potential of RBF networks has been demonstrated several times [26, 23]. In a RBF network each hidden unit produces an activation determined by a radial function (usually a Gaussian) centred at a specific position. A diagram of a RBF network can be seen in Fig 1. Although Fig 1 suggests there is just one output, multiple output units can also be included. In RBF's

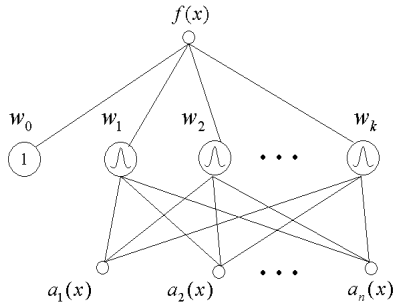


Figure 1: A radial basis function network

the learned hypothesis is a function of the form

$$\hat{f}(x) = w_0 + \sum_{u=1}^{k} w_u \mathbf{G}_u(d(x_u, x)) \tag{7}$$

where $\mathbf{G}_u(d(x_u, x))$ is the kernel function. It is common in practice to choose each function $\mathbf{G}_u(d(x_u, x))$ to be a Gaussian function centered at the point x_u. An overview of expression synthesis can now be be shown in Fig 2

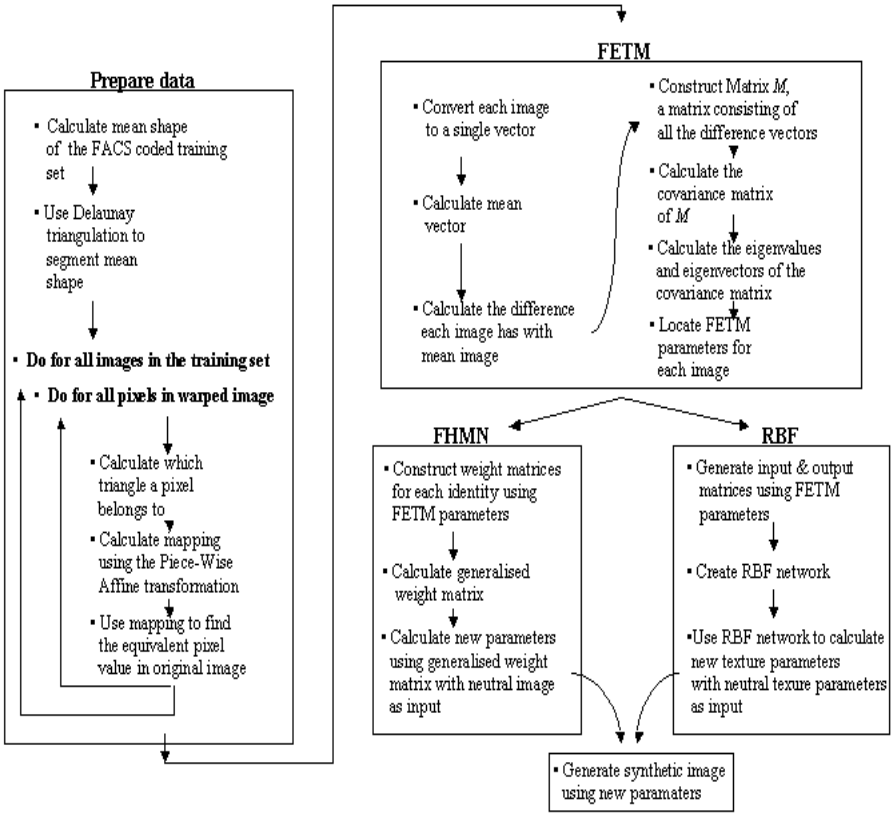


Figure 2: Texture Synthesis

4 Experiments and results

To create a FETM it is necessary to use a database that is consistent with the FACS description of an expression. For this reason we use the Cohn-Kanade AU-Coded Facial Expression Database [7]. The database includes approximately 2000 image sequences from over 200 subjects. All images used from the database are AU coded by certified FACS coders. The images used in the experiments described in this paper have been coded as AU 6 + AU 12 + AU 25. A short description of each is provided.

1. **AU 6:** Draws the skin from the temple and cheeks towards the eye. The outer band of muscles around the eye constricts.
2. **AU 12:** Pulls the corners of the lips back and upward, creating a smile shape to the mouth.
3. **AU 25:** Pulls the lips apart and exposes the lips and gums.

Forty people and 80 images from the Cohn-Kanade AU-coded facial expression database were used. Each image was acquired using a Panasonic WV3230 camera connected to a Panasonic S-VHS AG-7500 video recorder. The camera was located directly in front of the subject, and each image was digitized into 640 by 480 pixel arrays.

The mean shape was segmented using Delaunay triangulation and each image was warped to the mean shape using a piece-wise affine transformation. The mean image was then calculated (Fig 3). Each image was then represented as a single vector, subtracted from the mean image and the FETM was generated. The top 30 principal components of the FETM describe 95.60% of the total variance found in the training set. Fig 4 illustrates the effect of varying the top four principle components.

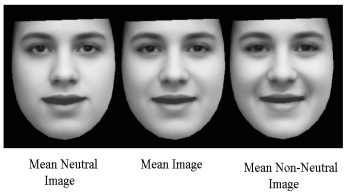


Figure 3: The mean images

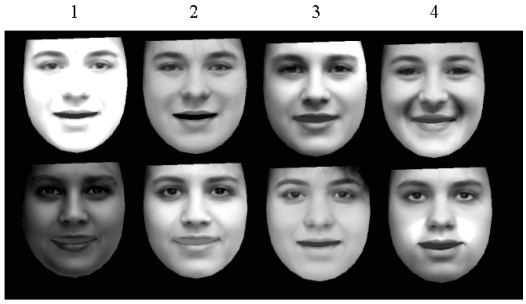


Figure 4: Top four principal components

A FHMN was used to generate a mapping from a neutral expression to one depicting the desired expression. Of the 40 subjects used to create the FETM, 37 subjects were used during the training of network.

This network failed to return convincing results with the FETM. Fig 5 illustrates the effect of passing an image through a mapping function created by a FHMN. It should be noted that the change in shape in Fig 5 is calculated using the *Facial Expression Shape Model* (FESM) [13, 15, 16].

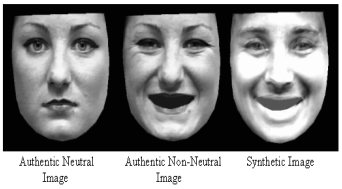


Figure 5: Expression Synthesis using a FHMN

To improve the mapping further we used a more sophisticated *Radial Basis Function Network* (RBFN) with the FETM. The top 30 principal components of the FETM were used to train the RBF. The training data consisted of 37 subject and 74 images. Three subjects were excluded from the training of each network to test each network with unseen data. The table below shows the correlation coefficients between the estimated and real principal components for the FETM in conjunction with a RBF network.

$Table_1$	$Experiment_1$	
$Subject$	$FETM$	RBF
1	0.9999	
2	1	
3	0.6017	
4	0.6771	
5	0.6208	
$Average$	0.7799	

Subjects one and two were used with 35 other subjects to train the network while subjects three, four and five are unseen test data. The test data for the FETM has a correlation coefficient of $t_{avg} = 0.6645$. Using a similar technique Yangzhou and Xueyin [28] showed how a $uniform\ function$ achieves results of $a_{avg} = 0.51$. This technique improves on this by computing a uniform function that achieves considerably better results. Fig 6 shows the error of the mapping within the FETM. The histogram on the left is the error of the mapping for all images in the training set and the histogram on the right shows the error for all the unseen images. Fig 7 illustrates the photo-realistic synthetic facial expressions of five different subjects. The first two rows consists of images of subjects that were used during the training of the RBF network while the next three individuals (rows 3, 4 and 5) were not used during the training of the network. Column one consists of shape free original images of individuals depicting neutral expressions. Column two consists of shape free original images of individuals depicting AU 6, AU 12 and AU 25 as described by the FACS. Column three consists of synthetic images of individuals portraying AU 6, AU 12 and AU 25 as calculated by the RBF network with neutral image parameters as input. Columns 4, 5 and 6 are the same as the first three columns respectively except with shape taken into consideration. The shapes in column 6 are calculated using a FHMN in conjunction with the *Facial Expression Shape Model* (FESM) [13, 16].

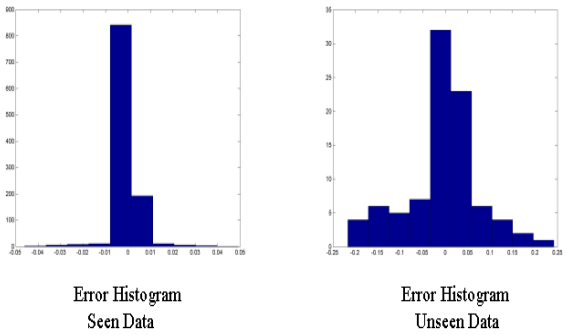


Figure 6: Error of the mapping

5 Conclusion and future work

This paper showed how a uniform mapping function was created which maps a neutral image of a face to one depicting a desired facial expression. This was achieved by the development of FETM and using this model several networks were trained to develop an accurate universal mapping function.

The FETM is based on the FACS, an anatomical analysis of facial actions. The FACS provides us with a universal method of analyzing facial expression and allowed for the generation of a texture model that is independent of subject (age, sex, skin colour etc.). The top 30 principal components of the FETM could describe 95.60% of the total variance found in the training set.

A FHMN was used to develop mapping functions which mapped an image of a neutral face to one depicting a smile (AU 6, AU 12, AU 25). This network over generalized the mapping and hence much of the identity of a subject was lost during the calculations. To improve the results a more sophisticated RBF network was used with the FETM. This networks greatly improved the results with a correlation coefficient between synthesized and authentic images of $t_{avg} = 0.6645$ was achieved. The results can be seen more clearly in Fig 7. The first two rows of this diagram show expression synthesis on data that was used during the training phase. This diagram shows how the technique is capable of dealing with changes skin colour. The images in the last three rows are images that were not present during the training phase. These images illustrate how this technique can generate a synthetic expression of a subject regardless of sex.

Figure 7: Original neutral, original non-neutral and synthesized images.

It is planned to use the FETM for expression classification. This could be done using similar neural networks to the ones detailed in this paper.

References

[1] Beinglass, A. and Wolfson, H. J. “Articulated object recognition”, Proceedings, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 461-466, 1991

[2] Birdwhistell, R.I “Kinesics and Context” Philadelphia: university of Pennsylvania Press, 1970.

[3] Balke, A and Isard, M, “Active Contours, The Application of techniques from graphics, vision, control theory and statistics to visual tracking of shapes in motion”,(Springer, 1998).

[4] Bozma, H. I. and Duncan, J. S. “Model-based recognition of multiple deformable objects using a game theoretic framework”, Information Processing in Medical Imaging-Proceedings of the 12th International Conference, pp. 358-372, Springer-Verlag, Berlin/New York, 1991

[5] Bruce, V. Young, A. “Understanding face recognition”. British Journal of Psychology, 77: 305-328. 1986.

[6] Brunelli, R. Poggio, T. “Face Recognition Features versus Templates” IEEE Transactions on PAMI, 15(10): 1042-1052, 1993.

[7] Cohn, J. Kanade “Cohn-Kanade AU-Coded Facial Expression Database”, Pittsburgh University, 1999.

[8] Cootes, T. F. and Taylor, C. J. “Statistical Models of Appearance for Computer Vision”, Wolfson Image Analysis Unit, Imaging Science and Biomedical Engineering, University of Manchester, Manchester M13 9PT, U.K. October 26th, 2001.

[9] Cottrell, G.W. Metcalfe, J. “Face, emotion and gender recognition using holons” Proceedings of the 1990 conference on advances in neural processing systems 3, 564-571, 1990.

[10] Ekman, P. and Friesen, W. V. “Facial Action Coding System”, Human Interaction Laboratory, Dept. of Psychiatry, University of California Medical Centre, San Francisco, Consulting Psychologists Press, Inc. 577 College Avenue, Palo Alto, California 94306, 1978.

[11] Faigan, G. “The Artist’s guide to Facial Expressions”, Watson-Guphill Publications, 1990.

[12] Fulcher, J.S. “Voluntary facial expressions in blind and seeing children. ” Archives of Psychology, ,38(272), 1942.

[13] Ghent, J. McDonald, J. “Generating a Mapping Function from one Expression to another using a Statistical Model of Facial Shape”, Proceedings of the Irish machine vision and image processing conference, 2003

[14] Ghent, J. McDonald, J and Harper, J. “A Statistical Model for Expression Generation using the Facial Action Coding System”, NUIM, NUIM-CS-TR2003-02, technical report, Jan 2003

[15] Ghent, J. McDonald, J. “An Overview of a Computational Model of Facial Expression”, NUIM postgraduate symposium, March 2004.

[16] Ghent, J. McDonald, J.“A Computational Model of Facial Expression”, NUIM-CS-TR-2004-01, technical report, Jan 2004.

[17] Grimson, W. E. L., “Object Recognition by Computer: the Role of Geometric Constraints”, MIT Press, Cambridge, MA, 1990

[18] Hill, A. and Taylor, C. J. “Model based image interpretation using genetic algorithms”, Image Vision Comput. 10, pp. 295-300, 1992

[19] Landis, C. “Studies of emotional reactions: II. General behavior and facial expressions” Journal of Comparitive Psychology, 4:447-509, 1924

[20] Lang, B. “The effects of processing requirments on neurophysiological responces to spoken sentences” PubMed 12191461 39(2): 302-318, 1990

[21] LeCun, Y. Boser, B. Denker, J.S. Henderson D. Howard, R.E. Hubbard, W. Jackel, L.D. “Backpropagation applied to handwritten zip code recognition” Neural Computation, 1(4): 541-551, 1989.

[22] Lispon, P. Yuille, A. L. O’Keeffe, D. Cavanaugh, J. Taaffe, J. and Rosenthal, D. “Deformable templates for feature extraction from medical images”, Proceedings of the first European Conference on Computer Vision (O. Faugers, Ed.), Lecture notes in Computer Science, pp. 413-417, Springer-Verlag, Berlin/New York, 1990

[23] Moody, J. Darken, C. “Fast learning in Networks of locall-tuned processing units” Neural Computation, 1:281-294, 1989.

[24] Perret, M. Hietanen, J.K. Oram, P. Benson, P. “The effects of lighting conditions on response of cells selective to face views in the macaque temporal cortex” Exp. Brain Res. 89: 157-71, 1992.

[25] Rhodes, G. Brake, S. and Atkinson, A. “Whats lost in inverted faces?” Cognition, 47: 25-57, 1993.

[26] Powell, J.D. “Radial basis functions for multivariate interpolation: a review” Clarendon Press, Oxford, UK, 1986.

[27] Staib, L. H. and Duncan, J. S. “Parametrically deformable contour models”, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, pp 427- 430, 1989

[28] Yangzhou, D. Xueyin, L. “Emotional facial expression model building”, Pattern recognition letters 24, pp 2923-2934, 2003

[29] Young. G and Decarie, T.G. “An ethology-based catalogue of facial/vocal behaviours in infancy” Archives of Psyhology. 37, No. 264, 1941.

[30] Yuille, A. L. Cohen, D. S. and Hallinan, P. “Feature extraction from faces using deformable templates”, Int. J. Comput. Vision 8, 99-112, 1992

Fast Iris and Pupil Localization and Eyelid Removal Using Gradient Vector Pairs and Certainty Factors

A. Ajdari Rad
Computer Engineering Department
Amirkabir University of Technology
Hafez av., Tehran, Iran
ali@itsi.ws

R. Safabakhsh
Computer Engineering Department
Amirkabir University of Technology
Hafez av., Tehran, Iran
safa@ce.aut.ac.ir

Navid Qaragozlou
Computer Engineering Department
Amirkabir University of Technology
Hafez av., Tehran, Iran
navid@itsi.ws

Maryam Zaheri
Electrical and Computer Engineering Department
University of North Carolina at Charlotte
Charlotte, NC, USA
maryam@itsi.ws

Abstract
Generally, the iris identification system is composed of three steps: acquiring the eye image including iris region, localizing the iris region and feature extraction, and decision making by means of matching. Localizing iris region is a very expensive task and it takes about 50% of the time of the process. Because of circular shape of the iris, circle detection methods are widely used for iris localization. In this paper, we present a fast circle detection method that uses the gradient vector pair and certainty factors concepts. Using this approach, iris boundary can be found fast, accurate, and robust against head tilts. Also a simple idea is used to remove eyelids. Results of the method evaluated with CASIA database and show a significant improvement in iris localization performance in comparison to the current methods.

Keywords: Biometric, Iris recognition, Iris and pupil localization, Circle detection.

1 Introduction

The traditional methods of human identity verification such as using keys, certificates, passwords, etc., can hardly meet the requirements of identity verification and recognition in the modern society. Biometric identification provides a convenient and reliable solution to this problem, and attracts extensive interests in the industry. Due to its various advantages, iris based identity verification is one of the most important biometric methods. Such advantages are persistency of the iris pattern over a long period of time, no need for direct contact with the subjects, automatic and rapid identification process. Also the reliability of iris-based identification is considerably high.

Iris identification includes three steps. The first step is acquiring the eye image including the iris region. Then the iris is localized and its features are extracted. The last procedure is making decision by means of matching. Localizing iris region is an expensive phase. In almost all iris recognition methods, it takes about 50% of the time of the process. Usually, the image acquisition step captures the iris as part of a larger image that contains other eye components, as well. Furthermore, if the eyelids cover parts of the iris, then that portion of the image above the upper eyelid and below the lower eyelid should be discarded. Also the contrast between eye components can be highly varied depending on the difference between pigmentation of the skin and the iris. Thus, iris localization must be insensitive to a wide range of unpredictable problems.

In this paper, we present a fast circle detection method that can find the iris boundary in an acceptable time and also is robust against head tilts. The results show a significant improvement in iris localization performance in comparison to current methods. Also a simple idea is used to remove eyelids. The paper is organized as follows. Section 2 describes pervious work on iris localization, focusing on one of the most famous approaches in detail. Section 3 presents details of the Fast Circle Detection (FCD) approach as a general method for finding circles that are brighter or darker than their background. How the FCD approach can be improved with certainty factors is

explained in section 4. Section 5 explains how the proposed approach can be applied to the iris localization problem. Section 6 presents experimental results of the proposed approach with two different data sets. Finally, in section 7, we describe the conclusion and our plans for future work.

2 Related Work

Since pupil is black, sclera white and iris gray, the simplest idea for iris localization is gray level thresholding of the eye image. This approach has disadvantages because of unpredictable color of eyelids and presence of eyelashes. Sometimes distinguishing between iris and skin gray levels is very hard. Also, the range of gray level of human eye components varies a lot. So, grayscale thresholding, by itself, cannot produce very good results.

Finding the iris based on its circular shape is another approach for iris localization. Some methods use Hough transform to detect a circle in the image or edge map of it [1]. The Circle Hough transform (CHT) is one of the best-known algorithms which aims at finding circular shapes with given radius within an image. In spite of its popularity, the CHT has some disadvantages. The major drawbacks of using CHT are the large amount of storage and computing power required by it in real-time applications. Also there were some approaches that have used cooperative modular neural networks [2] and cornea reflection [3] to find iris boundary.

The Wildes et al. [4] system performs contour fitting in two steps. First one is converting the image intensity information into a binary edge-map. The second one is voting the edge points to instantiate particular contour parameter values. The edge map is recovered via gradient-based edge detection. This operation consists of thresholding the magnitude of the image intensity gradient:

$$\left|\nabla G(x,y) * I(x,y)\right| \quad (1)$$

where:

$$\nabla \equiv (\frac{\partial}{\partial x}, \frac{\partial}{\partial y}) \quad (2)$$

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-((x-x_0)^2-(y-y_0)/2\sigma^2)} \quad (3)$$

while G(x,y) is a two-dimensional Gaussian with center (x_0, y_0) and standard deviation σ that it smoothes the image to select the spatial scale of edges under consideration. In order to incorporate directional tuning, the image intensity derivatives are weighted to favor certain ranges of orientation prior to taking the magnitude. For example, prior to contributing to the fit of the limbic boundary contour, the derivatives are weighted to be selective for vertical edges. The voting procedure is realized via Hough transforms on parametric definitions of the iris boundary contours. In particular, for the circular limbic or pupillary boundaries and a set of recovered edge points $(x_j, y_j), j = 1,2,...n$, a Hough transform is defined as:

$$H(x_C, y_C, r) = \sum_{j=1}^{n} h(x_j, y_j, x_C, y_C, r) \quad (4)$$

where:

$$h(x_j, y_j, x_C, y_C, r) = \begin{cases} 1, \ if \quad g(x_j, y_j, x_C, y_C, r) \\ 0, \ otherwise \end{cases} \quad (5)$$

with:

$$g(x_j, y_j, x_C, y_C, r) = (x_j - x_C)^2 + (y_j - y_C)^2 - r^2 \quad (6)$$

For each edge point (x_j, y_j), $g(x_j, y_j, x_C, y_C, r) = 0$ for every parameter triple (x_C, y_C, r) that represents a circle through that point. Correspondingly, the parameter triple that maximizes H is common to the largest number of edge points and is a reasonable choice to represent the contour of interest. In implementation, the maximizing parameter set is computed by

building $H(x_C, y_C, r)$ as an accumulator for $x_C, y_C,$ and r. Once populated, the array is scanned for the triple that defines its largest value. Contours for the upper and lower eyelids are fit in a similar fashion using parameterized parabolic arcs in place of the circle parameterization $g(x_j, y_j, x_C, y_C, r)$. Just as Daugman system relies on standard techniques for iris localization, edge detection followed by a Hough transform is a standard machine vision technique for fitting simple contour models to images. Both approaches have proven to be successful in the targeted application for localizing the iris. The histogram-based approach to model fitting should avoid problems with local minima that the active contour model's gradient descent procedure might experience.

By operating more directly with the image derivatives, however, the active contour approach avoids the inevitable threshold involved in generating a binary edge-map. Further, explicit modeling of the eyelids (as done in Wildes system) should allow for better use of available information than simply omitting the top and bottom of the image. However, this added precision comes with additional computational expense. More generally, both approaches are likely to encounter difficulties if required to deal with images that contain broader regions of the surrounding face than the immediate eye region. For example, such images are likely to result from image acquisition rigs that require less operator participation than those currently in place. Here, the additional image "clutter" is likely to drive the current, relatively simple model fitters to poor results. Solutions to this type of situation most likely will entail a preliminary coarse eye localization procedure to seed iris localization proper. In any case, following successful iris localization, the portion of the captured image that corresponds to the iris can be delimited. Figure 1 shows an example result of iris localization as performed by the Wildes system [4].

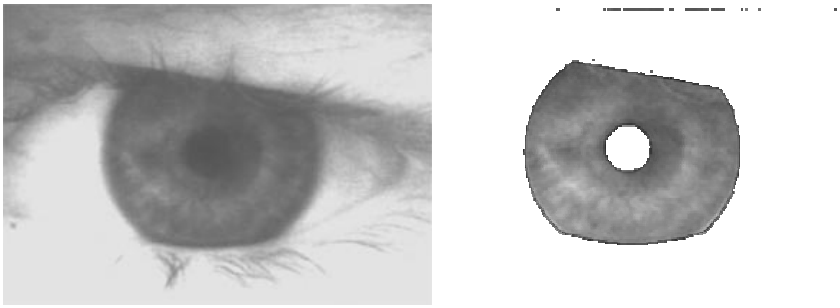


Figure 1. An example of iris localization according to Wildes method [4]

The Daugman approach [4] is the best-known algorithm for iris localization and recognition. This algorithm fits the circular contours via gradient ascent on the parameters (xc, yc, r) so as to maximize:

$$\frac{\partial}{\partial r} G(r) * \int_{r, x_c, y_c} \frac{I(x, y)}{2\pi r} ds \quad (7)$$

where:

$$G(r) = \frac{1}{\sqrt{2\pi}\sigma} e^{-((r-r_0)^2/2\sigma^2)} \quad (8)$$

is a radial Gaussian with its center at r0 and standard deviation σ that smoothes the image and * denotes convolution. In order to incorporate directional tuning of the image derivative, the arc of integration ds is restricted to the left and right quadrants (i.e., near vertical edges) when fitting the limbic boundary.

This arc is considered over a fuller range when fitting the pupillary boundary. However, the lower quadrant of the image is still omitted due to the artifact of the specular reflection of the illuminant in that region. In implementation, discrete equivalent of the above criterion is used.

More generally, fitting contours to images via this type of optimization formulation is a standard machine vision technique, often referred to as active contour modeling. Figure 2 shows two examples of Daugman's results [5, 6].

The above approaches have some benefits. They all rely on standard machine vision techniques for iris localization, and are relatively accurate and simple in user interactive applications. However despite these benefits, large amounts of calculation, high order of algorithm complexity, low performance for high resolution images and sensitivity to head tilts are some major disadvantages of them.

In the remaining of this paper, we present a fast and accurate approach for localizing the iris for recognition purposes.

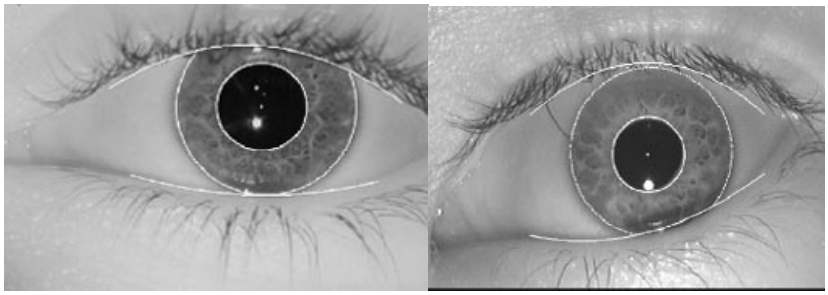


Figure 2. Examples of Daugman's method results [5]

3 Fast Circle Detection Using of Gradient Vector Pairs

In this section, we present a fast circle detection algorithm based on gradient vector pairs. Suppose that we have a dark circle on a bright background[1], as shown in Figure 3.a. The gradient vectors of the circle we search for are in the form shown in Figure 3.b. These vectors' directions are outward the circle, because the circle is darker than its background. Due to the symmetry of circle, for each gradient vector there is another gradient vector in its opposite direction. We call these vectors vector pair. As shown in Figure 4.a, a specific vector V1 is paired with a vector V2 if the following two conditions are satisfied:

(i.) Angle α, defined as the absolute difference between directions V1 and V2, should be nearly 180 degrees.

(ii.) Angle β between the line connecting P2 to P1 (the bases of V2 and V1) and the vector V1 should be nearly 0 degree[2] (This means that $\overrightarrow{P_2P_1}$ should be in the same direction as V1).

The second step of the algorithm is applied to find all vector pairs according to the above conditions in the gradient image. The second condition considerably removes noise by filtering useless vectors. As Figure 4.b shows, vectors V1 and V2 are not assumed as a vector pair due to condition (ii); however they satisfy condition (i).

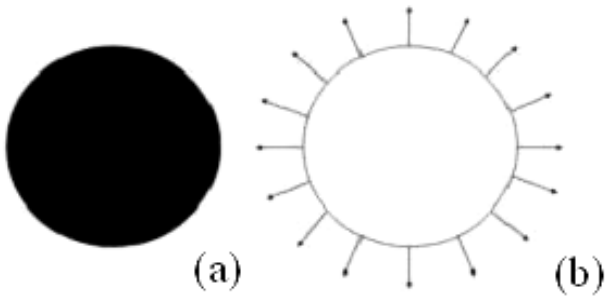


Figure 3. (a) A black circle in white background, (b) Gradient vectors of (a)

To increase the speed of pair matching, vectors are sorted according to their directions. So, for each specific vector, vectors with opposite direction can be found fast and easily.

[1] We can assume this without loss of generality because if the circle is brighter than its background, we can work on the negative image or simply reverse the direction of vectors.

[2] Or it should be in the opposite direction of V_2, because they are nearly parallel according to condition (i.)

In the third step, a candidate circle is considered for each pair of vectors. Such a circle has its center at the midpoint of P1 and P2, and its radius is equal to half of the distance between P1 and P2. Figure 4.a shows such a candidate circle in dashed lines. In special cases, if the approximate radius of the desired circle is known, a third condition can be used to filter out those vector pairs whose distances are outside the range of the expected values. This can improve the performance of algorithm significantly.

In the fourth and final step, the desired circles are extracted from the candidate circles produced in the previous step. There are two ways to do this. One way is employing a 3-dimensional accumulator matrix to count the occurrence of quantized circles. Then, the desired circles can be found by searching for local maxima in such a space. This is just like the classic CHT approach.

As the candidate circles are known, we use an easier approach to find the desired circles. Candidate circles are saved as a set of triples (Cx, Cy, r). These triples are then clustered using Euclidian distance between them. The means of clusters then specify the desired circles. The method reduces the space complexity and optimizes the entropy of the saved data. Also, prior knowledge about the number of the circles in the image can be used to get better results. The clustering method depends on problem attributes. If we know the number of circles to be found then we can use top hierarchical clustering to reach such numbers. Otherwise we can use variance minimizing methods to find proper clusters. In an application that aims to find one circle, clustering is replaced by averaging.

The FCD is a general circle detection method and can be applied to wide range of applications. Also if the definition of Pair Vectors is changed then it can be applied for some other shape detection approaches like arc detection, ellipse detection, and sphere detection. Because the FCD is presented for general circle detection adjusting parameters of algorithm is critical and depends on image features and statistics. The parameters give "adaptation to application" ability to the method. By increasing α and β, more pair vectors will be found. This may lead to better result (robustness against noise) or worse result (finding more wrong pair vectors). So adjusting these parameters is an art and significantly depends on problem features.

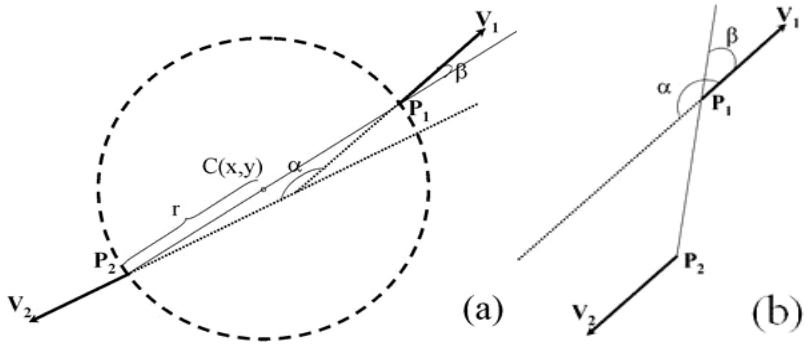


Figure 4. (a) Pair vectors and their candidate circle, (b) vectors rejected by condition (ii).

4 The FCD Improvement Using Certainty Factors (FCD-CF)

According to the original FCD, two vectors make a pair if they satisfy conditions i and ii of previous section. In real applications, these conditions are rarely exactly satisfied. So a range of acceptable values should be used for α and β. If the deviations of α or β from their ideal value (180 and 0 degrees) become more than a threshold value, the vector pair is omitted and if both parameters place in range, the candidate circle is considered. In contrast with this binary decision, two certainty factors can be considered according to mentioned angles:

$$C_\alpha = \exp(-\frac{(\alpha - 180)^2}{\sigma_\alpha^2}) \quad (9)$$

$$C_\beta = \exp(-\frac{\beta^2}{\sigma_\beta^2}) \quad (10)$$

These factors show the rate of satisfaction of i and ii conditions and can be used to control the behavior of the algorithm. Parameter σ can be used to adjust the effect of acceptable tolerance according to problem attributes.

Also, symmetry property of circle can be used to improve the performance of the FCD. By increasing the noise in the image the probability of matching of two random vectors as a vector pair increases. These wrong vector pairs increase error rate. Also, when the number of candidate circles is increased the clustering time is growth as well. Without loss of generality assume that the center of the candidate circle produced by v1 and v2 is placed at origin. According to figure 5, if a vector pair founded then six other points should be placed on the same circle as well.

To reduce the effect of noise, we can verify the existence of other six points in edge map of the image. If number of founded points is less than a specified threshold then the vector pair is omitted. In this way, random vectors that are not really placed on a circle are discarded. By omitting wrong candidate circles the number of candidate circles is reduced so the clustering step can be execute faster. If number of founded edge points is more than specified threshold then a candidate circle is considered just like the standard FCD. The number of founded points (n) also used to produce another certainty factor for the candidate circle denoted by C8:

$$C_8 = \exp(-\frac{(n-8)^2}{\sigma_8^2}) \quad (11)$$

In implementation, if each of Ca, Cb, or C8 becomes negative we omit the vector pair otherwise tree certainty factors used to make a final certainty factor according to the following formula that can be used to pair vector filtering and weighted clustering of final circles.

$$CF = iC_\alpha + jC_\beta + kC_8 \quad (12)$$

Parameters i, j, and k can be used to form different formulas according to problem features. For example for almost hidden circles, k parameter should considered near zero and for finding circles in noisy images, i parameter should be decreased. For the best adaptation to special problems, parameters, i, j, and k can be learned.

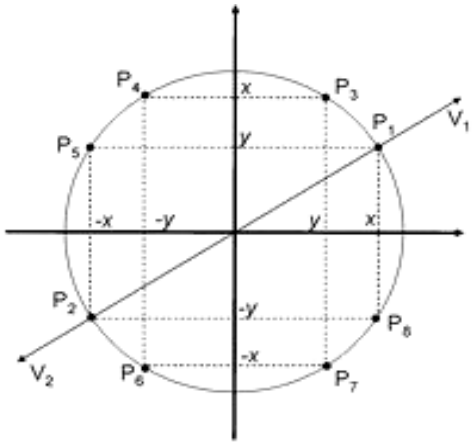


Figure 5. The pair vector and its related 6-points

5 Fast Iris Localization Using The FCD

This section explains the utilization of the FCD-CF method for iris localization. The method is faster than current algorithms and robust against head tilts. The algorithm has three major steps: pupil boundary detection, outer iris boundary detection and eyelids removal.

To increase the algorithm's performance, some preprocessing is suggested. A Gaussian filter can be used to smooth images. As pupil is a dark component, edge points corresponding to low gray level points can be considered to find pupil boundary. These preprocessing steps significantly reduce the number of the candidate gradient vectors so the algorithm can work faster.

As mentioned earlier, knowledge about radius can increase the performance of the proposed approach method. In iris recognition applications, the ranges of iris and pupil radii are predictable based on the image capturing device and the distance. Therefore this knowledge can be used to find boundaries more accurately and faster.

In the next step, after finding pupil boundary, iris outer boundary should be found. Since the location of the center point and range of the radius of iris can be predicted precisely, iris boundary can be easily found using the proposed approach. The proposed approach searches for gradient vector pairs in a limited area, and candidate circles are checked for having center and radius in valid ranges.

After finding pupil and iris circles, a post processing step can be applied to reach the better accuracy rate than before post processing. We use circle detector operator of Daugman in a very limited region around found circles.

To eliminate the eyelids, eyelashes and cornea reflections, other approaches remove the top and bottom 90-degree cones of the iris circle because head tilts are not acceptable [5, 7].
Contrary to current methods, before finding and removing eyelids we normalize iris using Daugman's approach:

$$\mathrm{I}(\mathrm{x}(\mathrm{r},\theta), y(r,\theta)) \rightarrow I(r,\theta)$$
$$\mathrm{y}(\mathrm{r},\theta) = (1-\rho)x_p + \rho\, y_s \qquad (13)$$
$$\mathrm{y}(\mathrm{r},\theta) = (1-\rho)x_p + \rho\, y_s$$

where I(x,y) is the raw image, (xp, yp) and (xs, ys) are centers of found pupil and iris respectively. The above formulas spread iris tube and present it as a 360*64 pixels rectangle. Figure 6 shows the result of normalization step.

Figure 6. Result of spreading of iris and normalization step

After normalization, eyelids appear as two semi-circles in predictable regions. Such semi-circles can be found easily using circle detection methods which adjust to find high contrast circles in predictable regions.

The proposed approach is an extremely fast and size-invariant method, so it is suitable for real-time and user interactive applications. The applications which use our approach as iris and pupil detection step can work in a user friendly manner. Due to the above advantages, distance of subject to image capture device can vary, pupil dilations and head tilts are acceptable and capturing a large number of images in time unit is possible.

6 Experimental Results

The proposed method has been tested for iris localization using the CASIA iris image database [8]. This database contains iris images of 108 individuals. There are 7 different 320x280 images for each subject. Radius of iris and pupil in each image are in the range of 28-75 pixels and 80-150 pixels respectively. In addition, a database of 100 higher resolution iris images (640x480) has been made, and the accuracy and speed of the FCD-CF approach has been tested with both data sets. In the second data set, there are some images with tilted subject heads. Radius of iris and pupil in each image are about 80 pixels and 250 pixels respectively.

For the purpose of comparison, the Daugman's localization algorithm and the original FCD have also implemented. All implementations have done in Matlab 6.1 environment using a system with 1.8MHz Pentium IV processor and 512MB RAM.

In pupil detection step, the result of the FCD and the FCD-CF are same because of very smooth edge of pupil. So we have only applied the FCD algorithm. In this step, both α and β parameters were set to 5 degree, and gradient vectors were calculated using Sobel operator and averaged in 5x5 windows and finally threshold by 30%. About 1755 pair vectors were found in average.

Significantly the FCD-CF has better result to find iris boundary than the FCD method. For iris outer boundary detection, both α and β parameters were set to 10 degree and gradient vectors were calculated using Sobel operator and averaged in 7x7 windows and finally threshold by 10%. The parameters of FCD-CF were adjusted as below:

$$\sigma_\alpha = 10, \quad \sigma_\beta = 10, \quad \sigma_8 = 4,$$
$$i = 1, \quad j = 2, \quad k = 0.5$$

There are about 327 pair vectors found in average in this step.
After finding circles, circle detector operator of Daugman was applied in 3 pixels around pupil found circle and 7 pixels around iris found circle. This post process improved the accuracy of the result about 3%. Also ranges of radius of iris and pupil in each image were given to both approaches. Upper of given ranges have been set with adding 50% of radius to actual radius, and lower bound of given ranges have been set with subtracting 25% of radius from actual radius.
To remove eyelids, the Canny operator was applied, and the result thresholded to omit edges that were below 0.3. Then Daugman's circle detector was applied in [45 135] and [245 295] columns of normalized image.

There is an important comment that should be mentioned here. Parameters of the proposed approach do not limit and weak the generality of approach. It can be observed that the adjusted parameters work for two different databases properly. As we mentioned before, parameters can be used to reach the best performance of the algorithm.

According to our experiments, Daugman's approach is able to localize about 83% of images. Our approach is able to localize more than 91% of CASIA images. Table 1 shows the average execution times of the proposed approach and Daugman localization algorithms.
From Table 1, for CASIA images, our approach is about seven times faster than Daugman's approach on the average. For higher resolution images, our approach has run near 14 times faster than Daugman's algorithm on the average.

Table 1. Average of execution times (in seconds)

	Proposed Approach	Daugman's Approach
CASIA (320x280)	0.84	6.37
Our dataset (640x480)	2.20	28.78

Figure 7 shows the response time histogram of both approaches. The standard deviation factor of the proposed approach and Daugman's approach have been calculated as 0.08 and 1.51 respectively. According to this figure, we find out that the response time of the proposed method is more deterministic and predictable which is a very important factor for real-time and human-interaction systems. According to our experiments, pupil localization takes 30% of computing time, iris outer boundary detection takes about 55% of computing time and image preparation and eyelids removal take about 15% of computing time.

Table 2. The results of accuracy tests of the proposed approach

	Centers Pupil/Iris	Radii Pupil/Iris	Overlap Pupil/Iris
CASIA dataset	0.03/0.05	0.04/0.07	0.98/0.94
Our dataset	0.01/0.02	0.01/0.04	0.99/0.96

Another test was done to estimate the accuracy of the proposed approach. One hundred different images were selected and the exact iris and pupil circle boundaries on them were determined manually by a human operator. The error rate was calculated in three ways: Distance of the estimated centers from actual centers, difference between estimated radius and actual radius and percentage of overlap of circle surfaces. Table 2 shows the final result of accuracy tests. Distance between centers and difference of the radii are normalized by the actual radius.

According to these results, our approach can find the pupil boundary significantly accurate and reliable. Also iris outer boundary detection shows very good performance. In almost all

experiments more than 97% of the visible part of iris is detected which is proper for recognition or other related applications. It should be mentioned that if the error rate of localization step passes about 10% the overall recognition system will fail. So the accuracy of Daugman's approach (and also other methods) is very close to the accuracy of our method. So, we could reach a seven times faster speed without losing the accuracy rate.

Some other localization methods can be robust against head tilts. For example, the CHT is rotation invariant and can find iris boundaries in any direction. Since other methods do not use edge directions, making them robust against head tilts (without losing accuracy) is expensive (heavy calculations and heuristics) and cannot be used in real-time applications.

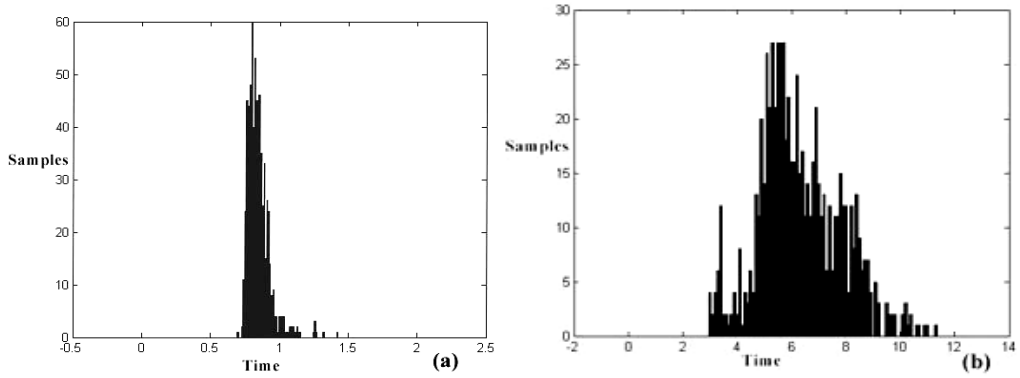


Figure 7. Histogram of response time for (a) propose method and (b) Daugman's method

Another evaluation is done to compare the resistance of the proposed algorithm against noise. The density of the pepper and salt noise was increased from 0 to 80 percent for each input image, and error rates (number of wrong localized images) of the proposed and Daugman's approaches were calculated. The results of this experiment shows that the proposed approach resists against noise until 30% noise rate but the Daugman's approach just resists until 17% noise rate. After the mentioned threshold, the error rate increases exponentially for each approach but with the lowest slope for the proposed approach.

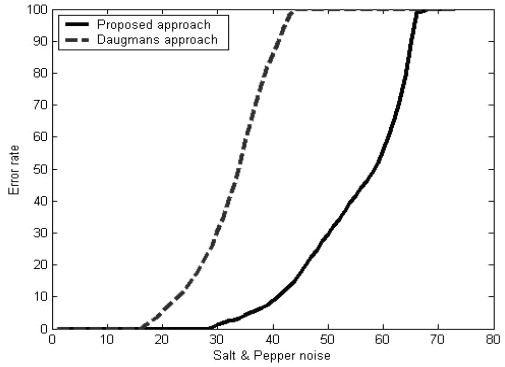


Figure 8. Resistance against pepper-salt noise

The Daugman's approach is totally failed after 41% noise but the proposed method resists until 68%. The surprising result is obtained according to nature of the FCD. When the noise increases it causes increasing of the unreal edges. Filtering nature of the FCD resists against this change but Daugman's approach count these unreal edges in its circle detector operator. Figure 8 draws the error rates versus noise density for each approach.

Iris localization using gradient vector pair approach has some drawbacks too. If more than half of iris is invisible, finding vector pairs will be impossible, so the FCD method fails in this case. Our approach fails for less than 9% of images of the CASIA database due to this reason. Fortunately, in such cases, the iris information is often not useful and the recognition system can

discard the image and try to capture another one. Figure 9 shows two output results of our approach.

7 Conclusions

In this paper, we presented a fast, size-invariant, application adaptable and accurate algorithm to find iris boundaries for recognition purposes. The algorithm consists of pupil boundary detection, iris outer boundary detection, eyelid removal, and boundary fitting. The method is based on using the symmetry of the gradient vector pairs on pupil and iris circle boundaries. The algorithm was implemented and its performance was compared with Daugman's localization algorithm using CASIA iris image database and 100 other high resolution images.

The experimental results show that the proposed approach is more than seven times faster than Daugman's for CASIA database and nearly 14 times faster for higher resolution images. As mentioned, iris localization takes about 50% of time and calculations in iris recognition applications, so using presented method can speed up the overall process significantly. Also a test on accuracy of our approach shows about 99% accuracy for pupil localization and about 97% for iris outer boundary detection. The accuracy rate is fitting for all current recognition approaches.

Persistency against noise is another advantage of our approach compare to the current methods. According to our experiments, if the input image is affected by 30% pepper and salt noise, there is no change on accuracy of proposed method. If the pepper and salt noise reaches near 50%, the error rate of the proposed method will be less than 25%.

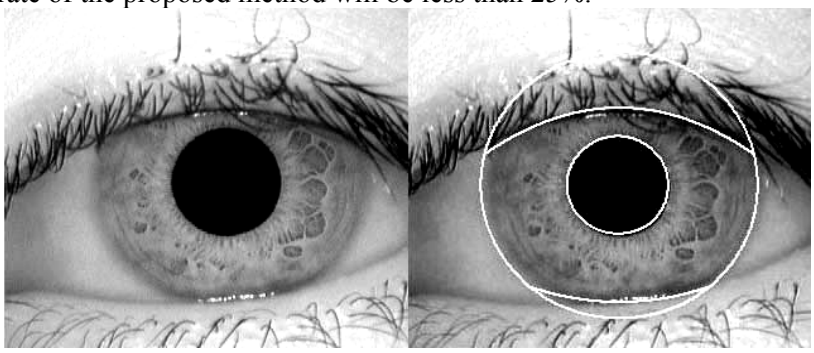


Figure 9. Output result of the proposed approach

References

[1] Zhu, Y., Tan, T., and Wang, Y., "Biometric Personal Identification System Based on Iris Pattern", *Chinese Patent Application*, No. 9911025.6, 1999.

[2] El-Barky, H., "Human Iris Detection Using Fast Cooperative Modular Neural Nets", Machine Graphics & Vision International Journal, Vol. 11, No. 4, pp. 499-512, 2002.

[3] Kong, W., and Zhang, D., "Detecting Eyelash and Reflection for Accurate Iris Segmentation", *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 17, No. 6, pp. 1025-1034, 2003.

[4] Wildes, R., "Iris Recognition: An Emerging Biometric Technology", *Proceeding of IEEE*, vol. 85, no. 9, Sep. 1997.

[5] Daugman, J.,"Biometric Personal Identifcation System Based", *U.S. Patent* No. 5, pp. 264-271, 1 March 1994.

[6] Daugman, J., and Downing, C. "Epigenetic Randomness, Complexity, and Singularity of Human Iris Patterns", *Proceedings of the Royal Society, Biological Sciences* 268: 1737-1740, 2001.

[7] Park, C, Lee, J., Smith, M., and Park, K., "Iris-Based Personal Authentication Using a Normalized Directional Energy Feature", *AVBPA 2003, Lecture Notes in Computer Science*, vol. 2688, pp. 224-232, 2003.

[8] CASIA Iris Image Database, http://www.sinobiometrics.com, 2003.

WAVELET-BASED FACE LOCALIZATION IN UNCONSTRAINED SCENES

J.-W. Wang
Institute of Photonics and Communications
National Kaohsiung University of Applied Sciences
415 Chien Kung Road, Kaohsiung, Taiwan 807, R.O.C.
jwwang@cc.kuas.edu.tw

Abstract
We formulate the human face detection task upon upright vertical frontal views faces in complex scenes as a wavelet-based problem and develop a novel approach using the extrema density aims to determine the image position of a single face. At first, face-of-interest (FOI) region is located and framed with an overlaid bounding box by finding facial edges using the inter-orientation wavelet subbands and the anthropometric measure. Then, a refining step is carried out to reposition and resize the bounding box for FOI to further improve the localization accuracy. Comparisons with two existing state-of-the-art face detection works and an on-line face demo system are presented, showing that our system has a comparable performance in terms of face localization rate and quality.

Keywords: Extrema density, face-of-interest (FOI) region, face localization.

1 Introduction

Face localization, which is a simplified detection problem and aims to determine the image position and size of a single face, is a fundamental stage in the process of face recognition. The accuracy of the detected face coordinate has a heavy influence on the recognition performance since most techniques (e.g. eigenfaces [1]) assume the face image normalized in terms of scale and rotation, their performance depends heavily upon the accuracy of the detected face position within the image. This makes face detection a crucial step in the process of face recognition. Recently, a sizable body of research in the area of face detection has been amassed. An excellent survey of the relevant literature can be found in [2]. A major technical challenge that needs to be addressed in various directions is the unsatisfactory performance of face detectors in rather unconstrained environments.

Recent works have proposed the use of wavelet functions as activation functions and have shown their powers in face detection problems [3]. Although wavelet decompositions can map the useful information content into a lower dimensional feature space, however, with the selected basis what feature is an efficient representation and how to develop a computationally efficient face localizing algorithm still deserve further study. In the current paper, based on dyadic discrete wavelet transforms (DWT) [4] we propose an efficient localization method to extract 2-D wavelet extrema density as feature from the three octave-width subbands decomposed. Then, a gradient-based boundary search algorithm is in turn used to find a coarse boundary of FOI. Finally, a refining step is carried out to readjust the previous located bounding box by using the head contour detection. The remainder of this paper is organized as follows. In Section 2, the characterization and the extraction of wavelet extrema density for facial edge detection is given. Section 3 describes the proposed face localization and head contour refining method. Experimental results and comparisons of the proposed scheme with the existing works are provided in Section 4. Conclusions are drawn in Section 5.

2 Facial Feature Extraction

Texture measure, which offers a means of detecting objects in background clutter that has similar spectral characteristics, is the visual cue due to the difference between human face and background [5]. To describe facial texture, one obvious feature is roughness. Since a face may exhibit different roughness over the decomposed wavelet subbands, it is proper in reality to detect face region by investigating the utility of feature derived from wavelet transform extrema. Roughness corresponds to the perception that our sense of touch will feel with an object and it can be characterized in two-dimensional scans by depth (wavelet coefficient strength) and width (separation between wavelet extrema). This interpretation prompted us to estimate the selected extrema as a particular signature of roughness, being very useful as a distinctive feature

of face texture measures. The properties of these extrema were studied in [6], and they turn out to be among the most meaningful features for signal characterization. The extremum at a point $f(x, y)$ of the horizontal channel component from its wavelet transform, Wf , is defined as:

$$Max_r : Wf(x, y) > max(Wf(x+1, y), Wf(x-1, y)), \quad (1)$$

$$Min_r : Wf(x, y) < min(Wf(x+1, y), Wf(x-1, y)). \quad (2)$$

In addition, similar definitions for the extrema in the vertical channel, Max_c and Min_c , are also defined. A pixel is a local extremum if it is both a local row extremum and a local column extremum. The operator of wavelet extrema for a 2-D image signal f is then defined as

$$\begin{aligned} Ef = \{ & Max_r Wf \cap Max_c Wf , \; Max_r Wf \cap Min_c Wf , \\ & Min_r Wf \cap Max_c Wf , Min_r Wf \cap Min_c Wf \}. \end{aligned} \quad (3)$$

This means Ef consists of coordinates of the wavelet extrema. Roughness of an image is not an absolute measure, but depends on the subbands at which the image processed. The LH wavelet subband makes a horizontal textured surface seem more remarkable, while the HL wavelet subband brings forward the rough structure of the surface at the vertical direction. To separate face-of-interest (FOI) region from the background, the extrema density $E_d f$ for an image of size M rows and N columns with extrema number $\#Ef$ is formulated as:

$$E_d f(\Gamma) = \# Ef / (M \times N), \quad (4)$$

where Γ is the threshold value for the wavelet coefficient, which can be quite critical in that it will affect the performance of successive steps such as face boundary localization. With the selected density range, which is usually obtained from the face databases by off-line learning, the threshold value Γ for wavelet coefficients is gradually increased in the meantime the thresholding step is repeated until a suitable facial texture representation for the task is reached. Extrema densities with various ranges, 0.1~0.06, 0.06~0.03, 0.03~0.015, 0.015~0.007, 0.007~0.0025, and 0.0025~0.001, are predetermined to investigate the exploitation of image attribute information such as edge. Using the 4-tap Daubechies [7] wavelet filter with two vanishing moments, an example of an original 384 × 286 face image as displayed in Fig. 1(a) being decomposed into four subbands for one level with extrema density 0.03~0.015 is shown in Fig. 1(b). As an illustration, consider Fig. 1(b), which is the result of applying a threshold value $\Gamma = 3$ to the subimage. We see that the subimages have been pleasingly sharpened by bringing out more of the facial texture details and the fine grain noise-like coefficients are less pronounced. These are much more acceptable results when compared to the rest of the other extrema densities, thus helping to localize the face region of interest. It is noted that the larger the extrema number the more information were the texture features found. However, the less accurate the boundaries between face and background become. This leads to a trade off between choosing a good facial region segmentation or good boundary between face and background.

3 Face Localization Algorithms

To develop an efficient localization method from an arbitrary uncontrived image, the face localization algorithm is proposed to precede expensive computations amidst three steps as followings:

Step 1: Wavelet extrema extraction at the LH, HL, and HH subbands, respectively.

Step 2: A thresholding process to adjust the extrema density for locating FOI by means of edge detection and anthropometric measure.

Step 3: The head contour detection is carried out to refine the previous located bounding box by comparing the detected FOI positions.

3.1 Locating Face-of-Interest (FOI) Region

As depicted in Fig. 1, we start by decomposing an image using one-level DWT and then extract the edge candidates for locating FOI. This could be very useful for the detection of facial directional textures, such as face boundary, since the separable sampling in DWT provides rectangular divisions of spectrum, with sensitivity to horizontal, vertical, and diagonal edges. Then, wavelet extrema density extractions are performed with the three inter-orientation decomposed subbnads, respectively, i.e. HL, LH, and HH. Consequently, the candidate edge segment is detected based on a measure of extrema discontinuity at a region, which is formed from the extrema number with values that exceed a preset threshold. Using the subband HL with size M/2 × *N*/2 and beginning from the two sides of the subimage, an approach detecting the vertical transition in extrema number associated with κ pixels region is given by

$$\sum_{\alpha=0}^{\kappa-1}\left|\sum_{y=0}^{M/2-1} Ef_{\mathrm{HL}}(x+\alpha,y) - \sum_{y=0}^{M/2-1} Ef_{\mathrm{HL}}(x,y)\right| \geq \varsigma, \tag{5}$$

where $0 \leq x + \alpha < N/4 - 1$ when starting from the left side of the image, $N/4 \leq x + \alpha < N/2\text{-}1$ when starting from the right side of the image, $\kappa = 4$ and $\varsigma = 3$. When the condition of the equation (5) occurs, the associated x coordinates are stored as candidate edge segments, and the procedure is performed before the search range is exceeded. Although attention thus far has been limited to a vertical edge, a similar task as well takes place at the LH subband to produce a candidate top edge of the horizontal orientation. In what follows, based on the aspect ratio of the face shape, which has been set to be [10/7, 6/4] in this work, a localization step is performed with the obtained candidate edges to search for the desired top and lateral boundaries of the FOI. With the located edges, thereafter, one can delimit the bottom edge of the FOI region.

3.2 Refining FOI by Using Head Contour Detection

As displayed in two examples of Fig. 2(a), due to different facial expressions, lighting conditions, and hairstyle, etc. the localization stage may raise a problem of imprecise localization of face, which can ultimately imply an erroneous face detection result. It would be unrealistic to hope for our framework that the bounding box for the previous FOI region has a pixel degree of precision. In order to tackle this problem, we present a refining approach to determine the up right and left contours of the head object, which are considered to be the most discriminative signature from the complex backgrounds. Constrained under around one and a third times the earlier located FOI region, by using the equation (5) for the three subbands a zig-zag scan procedure starting from the upper left and right corners downward diagonally, respectively. The head corner is determined by combining all the candidate segments among the HL, LH, and HH subbands because the desired signature is probably present at any one. The connected facial edge candidate with equal and above length of extrema number is then identified and located while the candidate segment of the head corner is detected. The final FOI location will be accordingly modified as illustrated in bold lines of Fig. 2(b), which are composed of the head corner and the facial edge. It is noted the FOI refining output will keep the same as the earlier framing result if there is no candidate segment available at the extreme case.

4 Face Databases and Experimental Results

In this section, we present the results calculated on BioID [8] and Visionics [9] databases, respectively. The first test set, a face database of mixed head inclination, gaze direction, hairstyle, gender, race, and age consisting of 1521 images (384 × 288 pixels, gray level) of 23 different persons, which has been recorded during different sessions and places at the BioID company headquarters. During the recording special emphasis has been laid on real world conditions. No restrictions on wear (clothes, glasses, etc), make-up, hairstyle, etc., were imposed to the participants. The second one is a commercial database, which comes from one of the leading pattern recognition systems available on the market. The database contains 120 color images with various sizes, each one showing the face of one out of 120 different test persons. For the purpose of determining system performance it is important to establish a clear definition of output classification. Successful face localization was defined as having at least FOI including both eyes, nose, and mouth located correctly. Localization rate is hereby defined as the ratio between the number of faces successfully localized and the number faces determined by a human. Some examples of face detection are shown in Figs. 3 and 4. Fig. 3 shows the examples of our results including correct and erroneous framing for the BioID test set, respectively, while Fig. 4 shows the examples of Visionics test set. Experimental results have demonstrated the effectiveness of the proposed method with localization rate 97.9% (1489/1521). However, it is also observed that the bounding rectangle (frame) for FOI may be larger than desired (background is not added to

the face) or improperly located when the faces are incomplete, with curl hairstyle, too dark, or too light, which complicate the face localization task considerably. The same is true for the framing when the head rotated excessively. In comparison to the results of the existing works, we would like to specially mention the results of [10] with 92.8% and [11] with 94.5% experimented on the same BioID data set as ours. It should be noted that it is hard to make a fair and an effective performance evaluation due to the lack of a common performance measurement for the face detection algorithm.

To further compare the performance of the proposed method with the currently available on-line face detector developed by Garcia [3] and Delakis at University of Crete, two data sets containing 100 random sample images taken from the BioID database and Visionics respectively are adopted to test the interactive demo system, which is located at *http://aias.csd.uch.gr:8999/cff/*. For the former dataset, our scheme performs slightly better than the Crete face detector with a successful detection rate of 97% while the quality of framing is also considered. No false dismissal is obtained using our scheme whereas three false dismissals (3%), for which the example images given no face found can be referred to Fig. 3, are obtained using the Crete face detector. It is interested to note that the last two of three false dismissals as shown in Fig. 3 from the Crete face detector are as well as failed to detect in Ref. [11], which implies that there are similar failure modes between them. Nevertheless, we have a completely different failure mode in the sense that the abovementioned images can be detected by our algorithm. Considering both our method and the Crete face detector, over-framed FOI cannot be totally avoided. In terms of speed, our system is faster, operating at an average processing time 1.6 ~ 2.1 sec per BioID image on a 1.0 GHz Pentium III PC. On the other hand, the Crete face detector processed at an average of 2.0 ~ 4.7 second per image on the same test data but on a different platform.

As presented in Fig. 4 for the Visionics data set, on the other hand, our algorithm detect 109 of the 112 faces which means a successful rate of 97.3%, whereas the Crete detector detects 98 faces of the 100 faces, leading to a successful detection rate of 98%. We observed that our detector failed mainly for faces of too dark. The main reason is that due to the dim lighting, which hides a significant part of the face, the number of extrema number in the LH subband is too few to detect the horizontal face boundary.

5 Conclusions

The presented framework led to fine face localization results, which did not involve sophisticated methods, is suitable for the application such as video telephony requiring the low-delay and limited computing power. A general face detection scheme may need to segment out the accurate face contours, however, thereby increasing the implementation cost.

Acknowledgements

The financial support provided by the NSC 92-2213-E-151-016 is gratefully acknowledged.

References

[1] Zhang J., Yan Y., and Lades M. (1997). Face recognition: eigenface, elastic matching, and neural nets. *Proc. IEEE*, 85:1423-1435.

[2] Yang M. H., Kriegman D. J., and Ahuja N. (2002). Detecting faces in images: a survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24:34-58.

[3] Garcia C. and Tziritas G. (1999). Face detection using quantized skin color regions merging and wavelet packet analysis. *IEEE Trans. Multimedia*, 1:264-277.

[4] Mallat S. G. (1989). Multifrequency channel decomposition of images and wavelet models. *IEEE Trans. Acous. Speech Signal Process*., 37:2091-2110.

[5] Craw I., Costen N., Kato T., and Akamatsu S. (1999). How should we represent faces for automatic recognition. *IEEE Trans. Pattern Anal. Mach. Intell*., 21:725-736.

[6] Mallat S. G. and Hwang W. L. (1992). Singularity detection and processing with wavelets. *IEEE Trans. Inform. Theo*., 38:617-643.

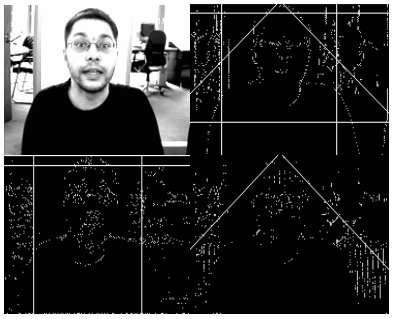
[7] Daubechies I. (1988). Orthonormal bases of compactly supported wavelets. *Comm. on Pure and Applied Math*., 91:909-996.

[8] http://www.humanscan.de/support/downloads/facedb.php

[9] http://www.identix.com/

[10] Kirchberg K. J., Jesorsky O., and Frischholz R. W. (2002). Genetic model optimization for Hausdorff distance-based face localization. *International ECCV 2002 Workshop on Biometric Authentication*, Copenhagen, Denmark, LNCS-2359:103-111.
[11] Wu J. and Zhou Z.-H. (2003). Efficient face candidates selector for face detection. *Pattern Recognition*, 36:1175-1186.

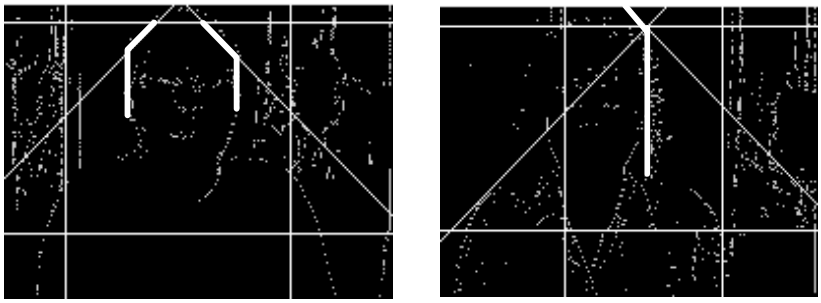
(a)



(b)

Fig. 1 (a) Original BioID image with located bounding box for FOI. (b) One-level decomposed wavelet subimages with extrema density 0.03~0.015 and detected facial edges.

(a)



(b)

Fig. 2 (a) BioID examples with improper localization and refined results. (b) The refining processes of (a) improved by using the head contour detection.

Fig. 3 Examples of correct and erroneous localization on BioID test images.

Fig. 4 Examples of correct and erroneous localization on Visionics test images.

Unsupervised Top-Down Object Segmentation: The Way for Image Information Content Assessment

Emanuel Diamant
VIDIA-mant
POB 933, Kiriat Ono 55100, Israel
emanl@012.net.il

Abstract
Effective image handling and processing are impossible without a proper assessment of image information content. A correct notion of the latter is not always well defined. Most often, it is used in the Shannon's sense, which deals only with an averaged (over the whole picture space) image information assessment. Human visual system rarely relies on such estimations. On the contrary, it is very effective in decomposing the observed scene into its particular meaningful constituents. That means, performing image objects segmentation in accordance with image information content. We argue that information content definition in Kolmogorov's sense is more suitable for such sort of tasks. Following the concepts of the Kolmogorov complexity theory, image information content can be defined as a set of descriptions of image data structures discernible (segmentable) within an image. We propose a technique for creating such image information content descriptions, which supposes a top-down unsupervised image segmentation procedure. We provide some illustrative examples, which demonstrate the effectiveness of this approach.

Keywords: Image understanding, image information content, image segmentation, image description.

1 Introduction

For humans, visual information was always the richest source of knowledge about their surrounding. However, despite of the growing use of various forms of imaging, the basic notion about what is visual information and what it essentially implies remain intuitive, uncertain and ambiguous. Most often, the expression "image information content" is used in the traditional Shannon's sense, which implies an average measure of uncertainty associated with an image generating process. But recently, it has become appreciated that measuring the randomness of a picture does not capture its inherent structure, that is, the intricate correlations between its constituents. It became generally agreed that information content is more adequately represented by the measure of image complexity, which reflects the regularities present in an object above and beyond pure randomness.

In the beginning, the idea to use complexity as a measure of information content was introduced (independently and approximately at the same time) by R. Solomonoff (1964) [1], A. Kolmogorov (1965) [2], and G. Chaitin (1966) [3]. It seems that a name like "Solomonoff-Kolmogorov-Chaitin Complexity" would be more suitable in this case (to give proper credit to all of the inventors). But over the time, the name "Kolmogorov Complexity" has become far more widely used. Following the general preference, we shall also use it in the subsequent discussion.

Following the theory of Kolmogorov's Complexity, we propose to define image information content as a set of descriptions of discernable image data structures perceived at different visibility levels. As such, three perceptual description levels can be generally distinguished: 1) the global level, where the coarse structure of the entire scene is initially outlined; 2) the intermediate level, where structures of separate, non-overlapping image regions usually associated with individual scene objects are delineated; and 3) the low level descriptions, where local image structures observed in a limited and restricted field of view are resolved. Assuming that the descriptions are created with a syntactically defined and fixed language, the total length of the descriptors may be considered as a quantitative measure of the image contained information.

2 Creating Information Descriptors

Kolmogorov's Complexity is a mathematical theory devised to explore the notion of randomness. Its basic concept is that information contained in a message (obviously, an image can be considered as a message) can be quantitatively expressed by the length of a program, that (when executed) faithfully reproduces the original message, [4]. Such a program is called the message description.

Various description languages can be devised and put to use for the purpose of description creation. Therefore, it is only natural to anticipate that a specific language will influence the length of the description and its accuracy. One of the important findings provided by Kolmogorov's complexity theory is the notion of language invariance, [5]. That is, the description language, of course, affects the length of object's description, but this influence can be taken into account by a language dependent constant added to the body of a language independent description, which actually is the Kolmogorov's complexity of an object. The latter determines the absolute amount of information in an individual object, and thus can be called the absolute Kolmogorov's complexity, [4]. The problem, however, is that this absolute Kolmogorov's complexity is (theoretically) unconstrained and, thus, it is practically uncomputable.

This topic would sound less discouraging if we will give up in advance the necessity of a perfect and accurate information description, if we would be pleased with its less complete and precise version. Practically that means that some part of image information would remain undiscovered and undescribed [5]. But essentially, we seldom use all the available information. Far more important for us is the insight of Kolmogorov complexity theory that in any case effective object description must commence with the simplest object structure delineation. An important equivalence between the shortest object description and the simplest object structure is established, [6]. The best way to achieve an object simplification is a some sort of object compression, when the existing object regularities are simply squeezed out from it. A hierarchical and recursive strategy for a description creation is thus emerged: Beginning with the simplified and course object structure, the description is subsequently augmented with more and more fine details unveiled at different hierarchical levels of object analysis and description.

3 Relevant Background

Traditional approaches, which deal with information content descriptions (like the recently introduced MPEG-7 standard), proceed with information features gathering (for the purpose of information descriptors creation) in a quite different manner. The widespread bottom-up information gathering approach is concerned, first of all, with processing of low-level elementary information pieces, which are initially searched and retrieved over the entire image space. Later they are grouped and aggregated into larger agglomerations, which are fed to the higher system levels for farther (higher-level) processing. To accommodate for external (user or system) requirements, that is, to incorporate the rules and principles by which disordered information pieces are combined and aggregated, a supervised top-down control flow is generally assumed. Its aim is to mediate the bottom-up information gathering. It is generally believed that this supervised intervention of a top-down conscious control leads to a more suitable and more task-fitting low-level information features acquisition, [7].

The roots of such preliminary bottom-up processing can be traced back to Treisman's Feature Integrating Theory [8] or Biederman's Recognition-by-components theory [9]. Relying on the evidence from human attentional vision studies, they were the first to propose the bottom-up manner of primary information gathering. However, the latest evidence put the correctness of the traditional approach in doubts. To properly understand the point, some words must be spent on the peculiarities of human vision: Human eye's retina has an odd structure – only a small fraction of its view field (approximately 2^0 out of the entire field of 160^0, [10]) is densely populated with photoreceptors. Just this small fragment of the retina (the so-called fovea) is responsible for our ability to see a sharp and clear picture of the surrounding world. The rest of the view field is a fast descending (in spatial density) placement of photoreceptors (from the fovea outward to the eye's periphery), which provides the brain with crude and fuzzy representation of the observed scene. To compensate for the lack of resolution over the entire visual field, continuous eye movements (also known as eye saccades) are performed, sequentially placing the high-resolution fovea over various (information rich) scene locations.

According to attentional vision theories the decision to make a saccade and to fix the fovea over a new image location ***precedes*** high-resolution (low-level) image information gathering, and hence, it can be yielded only by the coarse and poor information delivered by the peripheral vision. The flow of new evidence convincingly supports this suggestion: visual recognition/categorization tasks use "express", but comparatively imprecise and coarse-scale representations, before the fine-scale representations are acquired [11], the first signals reaching the

highest processing levels are from the eye's periphery, not from the fovea [12]. Not less surprising is the evidence that traditional assumptions about top-down intervention from the upper cognitive levels simply do not hold here. In most of the cases, saccadic movements are guided preattentively and unconsciously [13].

This flow of evidence from empirical studies of human attentional vision quite well support and come in agreement with the insights of the Kolmogorov's Complexity theory. Slightly twisted to fit the case of image information content exploration, the latter can be finally (and in brief) summarized as follows:

- Image information content is a set of descriptions of the observable image data structures.
- These descriptions are executable, that is, following them the meaningful part of image content can be faithfully reconstructed.
- These descriptions are hierarchical and recursive, that is, starting with a generalized and simplified description of image structure they proceed in a top-down fashion to more and more fine information details resolved at the lower description levels.
- Although the lower bound of description details is unattainable, that does not pose a problem because information content comprehension is generally fine details devoid.

4 Implementation Issues

Following the modern concepts of selective attention vision and the insights of Kolmogorov's Complexity theory, we propose a new way for unsupervised top-down image segmentation facilitating meaningful information content revelation and gathering. . Its architecture is shown in Figure 1, and it is comprised of three main processing paths: the bottom-up processing path, the top-down processing path and a stack where the discovered information content (the generated descriptions of it) are actually accumulated.

To facilitate the requirement for a top-down directed processing, we introduce a hierarchy of multi-level multi-resolution image representations called multi-stage image pyramid [14]. Such pyramid construction generates a set of compressed copies of the original input image. Each image in the sequence can be seen as an array that is half as large as its predecessor. The rules of this shrinking operation are very simple and fast: four non-overlapping neighbour pixels in an image at level L are averaged and the result is assigned to a pixel in a higher (L+1)-level image. This is known as "four children to one parent relationship".

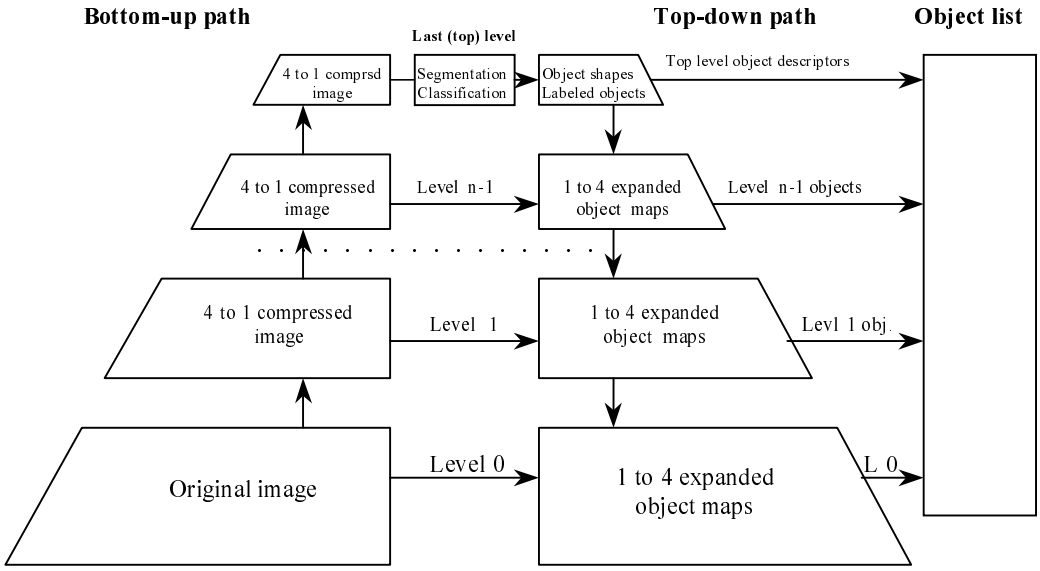


Fig. 1. The Block Diagram (Schema) of the suggested approach

At the top of the pyramid, the resulting coarse image undergoes a round of further simplification. Several image zones, representing perceptually discernible image fractions (visually dominated image parts,

super-objects) are determined (segmented) and identified by assigning labels to each of the segmented pieces. Since the image size at the top is significantly reduced and since in the course of the bottom-up image squeezing a severe data averaging is attained, the image segmentation/classification procedure does not demand special computational resources. Thus, any well-known segmentation methodology will suffice. We use our own proprietary technique that is based on a low-level (local) information content evaluation [15].

The technique first outlines the borders of the principal image fragments. Then similarly appearing pixels within the borders are aggregated in compact spatially connected regional groups (clusters). Afterwards, every cluster is marked with a label. Thus, a map of labeled clusters, corresponding to perceptually discernible image regions, is produced. Finally, to accomplish top-level object identification, for each labeled region its characteristic intensity is computed as an average of labeled pixels. This way, a second (additional) segmentation map is produced, where regions are represented by their characteristic intensities.

From this point on, the top-down processing path is commenced. At each level, the two previously defined maps are expanded to the size of the image at the nearest lower level. The expansion rule is very simple: the value of each parent pixel is assigned to its four children in the corresponding lower-level map (a reversed shrinking operation). Since the regions at different hierarchical levels do not exhibit significant changes in their characteristic intensity, the majority of newly assigned pixels are determined in a sufficiently correct manner. Only pixels at region borders (and seeds of newly emerging regions) may significantly deviate from the assigned values. Taking the corresponding current-level image as a reference (the left side, bottom-up path belonging images), these pixels can be easily detected and subjected to a refinement cycle. Here they are allowed to adjust themselves to the "proper" nearest neighbors, which certainly belong to one of the previously labeled regions (or to the newly emerging ones).

In such a manner, the process is subsequently repeated at all descending levels until the segmentation/classification of the zero-level (original input image) is successfully accomplished. It is clear, that the reconstructed image is not a "Just Notified Distortion" version of the original one. However, for most decision making purposes an exact detail-preserving information content description of an image is irrelevant. At every processing level, every image object/region (just recovered or an inherited one) is registered in the objects' appearance list, which is the third constituting part of the proposed scheme. (Notions of object and region are used in the paper interchangeably). The registered object parameters are the available simplified object's attributes, such as size, center of mass position (coordinates), average object intensity and hierarchical and topological relationship within and between the objects ("sub-part of…", "at the left of…", etc.). They are sparse, general, and yet specific enough to capture the object's characteristic features in a variety of descriptive forms.

This part of the processing scheme is (we suppose) the most suitable and natural place for external user interaction (a place for the "classical" top-down interference). User-defined task-dependent requirements can be easily formulated in human-friendly and human-accustomed forms, which are provided (supported) by the description implementations. The desired levels of description details are transparent (in the list) and are easily attended.

5 Experimental Results

To illustrate the qualities of the proposed approach we have chosen an unfamiliar (to most of the potential readers) picture 89072 from the Berkeley Segmentation Dataset [16]. By doing this we want to eliminate biasing of the observer's judgement about what is the right segmentation and to direct the observer's attention to the quality of segmentation pieces and their appropriateness to the final object segregation and decision making.

Fig. 2 represents the original image, Figs. 3, 4, and 5 are examples of the original image decomposition to regions of various detail complexity. Level 5 (Fig. 3) corresponds to the near-top-most hierarchical level (in this particular case, for the size of this particular image the algorithm builds a 6-level hierarchy). Level 1 (Fig. 5) is the lower-end-closest decomposition. For space saving, we provide only few examples from the image segmentation gallery, which for the reader's convenience are all expanded to the original image size. Extracted from the object list, the numbers of distinguished (segmented) at each corresponding level regions are also given in the capture of each figure.

Because real object decomposition is not known in advance, only the generalized intensity maps are presented here. But it is clear that even such simplified representations are sufficient to grasp the image concept. It is easy (for the user) now to define what region combination depicts the target object most faithfully.

Fig. 2. Original image, 324 x 480 pixels.

Fig. 3. Level 5 segmentation, 17 object-regions.

Fig. 4. Level 3 segmentation, 54 object-regions.

Fig. 5. Level 1 segmentation, 165 object-regions.

6 Conclusions

We presented a new technique for unsupervised image information content generation and top-down image decomposition to its constituent visual sub-parts. We rely on a hybrid bottom-up/top-down strategy which produces the simplest (the shortest, in terms of Kolmogorov's Complexity) description of image information

content. The level of unveiled description details is determined by the structures discernable in the image data and, thus, is independent from user intentions.

Despite a seeming similarity to the established multimedia content description standards, which (like MPEG-7 standard, e.g.) provide means and rules for image information content creation and Schemas for Object Description Design, our proposed approach is principally different:

- MPEG-7 description creation relies on a bottom-up process, [17]. This poses extreme difficulties for the initial object segmentation/identification. Therefore such a task is left beyond the standard's scope.
- MPEG-7 is not supposed to provide image reconstruction from the descriptions. Analogously designed descriptors can only be used for image comparison and similarity investigation purposes, (such as in Content Based Image Retrieval and other Web-related applications, [18]).

With respect to the standardized techniques, our approach has palpable advantages. We provide a technique that autonomously yields a reasonable image decomposition (to its constituent objects), accompanied by concise object descriptors that are sufficient for reverse object reconstruction with different levels of details.

References

[1] R. J. Solomonoff, "A Formal Theory of Inductive Inference", Part I, *Information and Control*, vol. 7, No. 1, pp. 1 – 22, March 1964.

[2] M. Li and P. Vitanyi, "An Introduction to Kolmogorov Complexity and Its Applications" (2nd ed), Springer-Verlag, New York, 1997.

[3] G. J. Chaitin, "Algorithmic Information Theory", *IBM Journal of Research and Development*, vol. 21, pp. 350-359, 1977.

[4] P. Grunwald and P. Vitanyi, "Kolmogorov Complexity and Information Theory", *Journal of Logic, Language and Information*, vol. 12, issue 4, pp. 497-529, 2003.

[5] N. Chater and P. Vitanyi, "The Generalized Universal Low of Generalization", *Journal of Mathematical Psychology*, vol. 47, issue 3, pp. 346-369, June 2003.

[6] N. Chater and P. Vitanyi, "Simplicity: A unifying principle in cognitive science?", *Trends in Cognitive Science*, vol. 7, issue 1, pp. 19-22, Jan. 2003.

[7] J. M. Wolfe, S. Butcher, C. Lee, and M. Hyle, "Changing your mind: On the contributions of top-down and bottom-up guidance in visual search for feature singletons", *Journal of Experimental Psychology: Human Perception and Performance*, vol. 29, pp. 483-502, 2003.

[8] A. Treisman and G. Gelade, "A feature-integration theory of attention", *Cognitive Psychology*, 12, pp. 97-136, Jan. 1980.

[9] I. Biederman, "Recognition-by-components: A theory of human image understanding", *Psychological Review*, vol. 94, No. 2, pp. 115-147, 1987.

[10] D. Van Essen, C. Anderson, D. Felleman, "Information Processing in the Primate Visual System: An Integrated Systems Perspective", *Science*, vol. 255, pp. 419-422, 24 Jan. 1992.

[11] A. Oliva and P. G. Schyns, "Coarse Blobs or Fine Edges? Evidence That Information Diagnosticity Changes the Perception of Complex Visual Stimuli", *Cognitive Psychology*, v. 34, pp. 72 – 107, 1997.

[12] S. Thorpe, "Ultra-rapid scene categorization with a wave of spikes", *Biologically Motivated Computer Vision: Second International Workshop, BMCV 2002*, In: LNCS vol. 2525, pp. 1-15, Springer-Verlag, Berlin, 2002.

[13] R. Rao G. Zelinsky,, M. Hayhoe, D. Ballard, "Eye movements in iconic visual search", *Vision Research*, vol. 42, Issue 11, pp. 1447-1463, May 2002.

[14] S. Tanimoto and T. Pavlidis, "A hierarchical data structure for picture processing", *Computer Graphics and Image Processing*, Issue 4, pp. 104-119, 1975.

[15] E. Diamant, "Single Pixel Information Content", *Proceedings of SPIE-IS&T 15th Annual Symposium on Electronic Imaging*, SPIE vol. 5014, pp. 460-465, 2003.

[16] Berkeley Segmentation Dataset – http://www.cs.berkeley.edu/projects/vision/grouping/.

[17] A. Jaimes and Shih-Fu Chang, "Automatic Selection of Visual Features and Classifiers", *IS&T/SPIE Conference on Storage and Retrieval for Images and Video Databases VIII,* SPIE vol. 3972, San Jose, CA, Jan. 2000.

[18] A. Jaimes and Shih-Fu Chang, "Model-Based Classification of Visual Information for Content-Based Retrieval", *IS&T/SPIE Conference on Storage and Retrieval for Images and Video Databases VII,* San Jose, CA, Jan. 1999.

AUTOMATIC VISUAL TRACKING FOR ANALYSIS OF LIFTING

Michael Wells
Mathematics
University of St. Thomas
Houston, Texas, USA

(Niels da Vitoria Lobo and Mubarak Shah)*
Computer Science
University of Central Florida
Orlando, Florida, USA

Abstract

This paper presents the tracking and analysis of lifting. Included in this paper is tracking along important regions of the body such as the torso. We propose a method which detects improper and proper lifting of large objects as seen in manufacturing and other industrial environments. We also propose the same method to be used as a tracking device for the torso and object. This paper includes a brief description of the ergonomics of lifting and those aspects of ergonomics that we included in the creation of our algorithm. Through proper classification of important regions of the body, general conclusions regarding the entities proper or improper method of lifting can be made. Along with the importance of ergonomics in the world, this paper also addresses the importance of tracking specific regions over time. By tracking the object that is lifted, a history can be developed on the movement of the object. In that history, the location of the object at specific times and how and where it was moved, dropped, etc. can be stored. Therefore this paper presents methods which can help ensure the safety of the object and the person through computer and camera monitoring.
Keywords*: Silhouette Extraction, Image Segmentation, Background Subtraction, Image Tracking*

1 Introduction and Related Work

According to the Bureau of Labor Statistics, one quarter of all injuries on the job are related to the back. Many companies have designed products like the Ergocube Ergonomic Container. These containers and others like them are designed to make lifting easier and better for workers. A system set up at a facility that could detect store workers lifting habits and point out lifting which may cause severe injury would be beneficial to workers and employers.

Tracking of objects is important to all areas of computer vision. Tracking of the torso is particularly important because of all the information it gives regarding the person's location and posture. Tracking the object lifted along with the torso of the person can together yield very important information on lifting for surveillance and monitoring purposes. Certain actions such as a lift can be classified as gestures along with other actions such as a wave or clap. See the following papers for more information on human action and recognition of gestures: [3], [4], [5], [6], [7], [8].

1.1 Components Used in our Algorithm

There are two components taken from other work that are used in our algorithm. Both of these components were modified slightly to be used in our application.

*A special thanks to my advisors who oversaw my work.

1. Rutgers Segmentation: The work done by [2] uses a simple nonparametric density estimation algorithm for feature space analysis. In this algorithm there are three methods for segmentation, under-segmentation, over-segmentation and quantization. For our method we used under segmentation which corresponds to the lowest segmentation resolution.

2. Knight System: The method implemented in the Knight System is used in our algorithm for its background subtraction algorithm as explained in [1]. The result of the difference between the background model and the subsequent frames in a sequence is a silhouette of the person. When an object in the background model is moved it becomes a part of the silhouette also, thus the object is in two places at once. In order to determine which pixels are valid foreground pixels the gradient and color based background difference are taken. The magnitude of the gradient in the background model at the object's original location is low but in the current frame where the object is moved the magnitude of the gradient is high because the location of the object has changed. A threshold, called the edge-ratio, is then applied to the product of these two gradient values. When the edge-ratio threshold is set very low the background model is not updated with the new uncovered background when an object is moved. When the edge-ratio threshold is set very high the old background model is then updated with the new background pixels when an object in the background model is moved. Taking the difference between the result of an image with a high edge-ratio and a low edge-ratio results in a region that is the objects original location that will be called the *candidate object region*. For a more detailed explanation of the Knight System and the use of gradient based subtraction see [1].

2 Silhouette Extraction

The first step to extracting a definite region for the torso and object is to use background subtraction to extract a silhouette of the person and objects moved. We use the background subtraction from [1]. In that approach we change the edge-ratio, as described in section 1.1.2, to 65%[1]. Because objects move from background to foreground when they are lifted in the scene, we do not want to include them in the bounded region after they are set down and no longer part of the silhouette. Using our connected components algorithm we accept only the connected regions of the background subtracted image.

In our connected components algorithm we first go to the background subtracted image with a 65% edge-ratio. At every white pixel in this binary image of the silhouette we dilate. Our connected components algorithm then finds the largest component of white pixels and keeps all other components within 10 pixels horizontally and 50 pixels vertically. We reason that components can be further away in the vertical and still be likely candidates of the silhouette whereas components that are far away in the horizontal are much less likely to be part of the silhouette. This conclusion is made due to the understanding that most people are taller than they are wide. The thresholds we chose may very as video sequences vary in complexity, however, these values worked best for the sequences that will be mentioned in the Results section.

Using the bounding box above, we then divide it into four different rectangular regions where the **head-region** is the top 16% of the bounding box and the **torso-region** is the next 33% of the region. The final 51% of the bounding box is then divided into two, where the top half is the **upper leg-region** and the bottom half is the **lower leg-region**. These individual rectangles in the bounding box give a region where most of each corresponding part of the body is likely to be found. These values were obtained through experimentation with video sequences we worked with and through work done by a previous REU[2] student who worked before me on carried bag detection at the University of Central Florida.

[1] This value allows for shadow removal, ghost object removal as well as large luminance changes which could result in erroneous background subtraction. This number was obtained by reading [1] and testing with many types of images with varying luminance and complexity.

[2] Research Experience for Undergraduates funded by the National Science Foundation

3 Bounding Box Segmentation and Torso Identification

The next step is to locate the torso in the bounding box. The torso is always found in the **torso-region** just below the **head-region**. Sometimes part of the torso is cut off by the **head-region**, therefore we look in both regions for the torso. By combining the result from the segmentation image and background subtraction image we find the segmented region with the highest percentage of background subtraction pixels. This method extracts the torso while the torso is in view. The method fails when the background is nearly the same color as the torso region of the foreground. In this extreme case even the background subtraction might fail. It is my goal in the future to use shape, contour[3], and gradient based information along with color information to locate and track the torso.

To solve the problem that may occur when separate regions are very close in color, we store the location of the torso pixels when identified for certain, and that location is not updated to a new position until a new confirmed torso region is found. The **torso-region** is tracked much like the object, which will be explained later on. Based on a history of the movement of the torso we can then predict approximately where the torso will be in the next frame by finding the average change in the horizontal and vertical components of the vector created by the movement of the torso from frame n to frame n+1. This same optical flow method is also used for minor occlusions when the torso region is not in complete view.

Once the torso is extracted we then find the second moment[4] line for the region to simulate the person's backbone. By tracking the orientation of this line with respect to the vertical we can determine when and how much their back is tilting. The torso algorithm depends greatly on the success of the detection of the object. Our algorithm confuses the object for the torso if our object tracker is turned off. Therefore they need to work together for the best results.

4 Visual Characteristics of a Lift

Our lifting detection algorithm is weighted by three parameters. When all three parameters are considered true the system concludes that a lift has occurred. In order to determine where the object is we assume that when a large object is picked up the person will descend over a certain threshold[5]. We use the first 0.6667 sec after the person has come into view to determine the average height of the individual. From that moment on after every lift the average height is recalculated over 0.6667 sec. As a person moves around lifting various objects the height of the person decreases as the person moves further away from the camera[6]. By recalculating the height throughout the sequence this ensures the accuracy of detecting a lift properly. If the person descends pass the threshold and then comes back over that threshold we conclude that the person has bent down and is now coming back up. This motion, called a **squat**, is the first parameter used to determine when to search for an object.

From the moment that the person is in view we calculate the silhouette's average width using the minor axis of a best fit ellipse. Before the person completes the **squat** we calculate the average width over 0.2 sec. Using the minor axis of the best fit ellipse for the silhouette we determine if the minor axis has increased by more than a certain threshold[7]. At this point, when the second parameter has been satisfied, we assume the person is holding something. Based on these two parameters we then go to the third parameter which will be explained below in the object detection section.

[3] my current research has been using the heat differential equation to find contours.

[4] The second moment of inertia is mathematically defined as $\iint_{Region} r^2 dxdy$

[5] This value can be set based upon the type of input images

[6] This phenomena is called parallax

[7] This value is set by the user

5 Object Detection

Once we know that the person has gone up and down, the **squat** parameter has been satisfied. We can then look for the object and begin tracking. We determine the candidate region for the object through the method described in section *1.1 Components Used in our Algorithm* item *2 Knight Method*. Based on the method used in the Knight System we chose to take the difference between the result of the image with the edge-ratio set at 40% with the result of the same image with the edge-ratio set at 80%. The difference between these two resulting images produces the *candidate object region*. We chose 80% and 40% because these values created the best results for the video sequences we tested, however these value may need to be adjusted based on varying background and luminance conditions in that may come to exist in an image.

The *candidate object region* obtained from some frame *n* is then tracked through the *k* frame history from the oldest fram in the history to *n* using Sum of Squared Differences, which will be explained in more detail in the next section. For our system we chose the value of forty for *k*. Through experimentation with video running at 30 fps, forty frames or 1.333 seconds of video is the best value for *k* because this value keeps track of a long enough history for the lifter to lift the object. If lifting takes longer than forty frames the value can be increased. I chose forty in particular to save computation time and space. As long as the history is shorter than the number of frames between two consecutive lifts the method will work. If the history is too large then the system will fail because it will overlap the lifting of more than one object. With more than one object lifted in the history sequence the system will not be able to distinguish which of the two objects is currently being lifted.

6 Object and Segment Tracking

Once the object is detected as explained above, we use our *k* frame history to find the current location of the object[8]. When the *candidate object region* is located, the object is then tracked through the *k* frame history up to its current location. Tracking is done through a method of Sum of Squared Differences (SSD). The *candidate object region* is used to generate a mask of pixels which is the maximum rectangle generated from the *candidate object region*. The (R,G,B) pixel values are then taken form the original color image based on the location of the rectangular mask obtained. Once the best fit mask is found for the object we then recursively fill the object region. Taking each neighboring pixel individually, if the pixel difference with its neighboring pixel is less than or equal to some threshold for each component in the color vector (R,G,B) then the new pixel is stored in the mask. The object pixels are then eroded and dilated and finally a new mask is generated[9].

Once this optimal mask O_{rgb} is found, the method looks to the next frame in the sequence and determines which possible mask N_{rgb} has the smallest difference.

$$\min \left[\sum_{x=-5}^{5} \sum_{y=-5}^{5} \left((O_{rgb} - N_{rgb})^2 \right) \right] \tag{1}$$

If the object is, for example, a backpack, and the person turns around, the object becomes occluded by the person's torso. In order to solve this tracking issue we first determine when the object is occluded. If the result from **equation 1** deviates more than some threshold value from O_{rgb} we place the bounding box and the pixel locations of the mask in the middle of the torso. While the new location of the mask is inside the torso, the original color values are still stored. We continue to search for the object in some neighborhood[10] while the object is occluded. When the object comes back in view after the occlusion

[8] The lifted object is not identified until the object has been significantly moved from its old location.

[9] In this new region, every other pixel is stored as the tracking mask to reduce the time it takes to search for the best translation of the mask that corresponds to the object.

[10] This neighborhood is defined by the user and varies based on the resolution of the image.

we then pick up the object again when the result from **equation 1** falls in the reasonable region of some threshold. This method for occlusion works only when the person is occluding the object while carrying or lifting. It is our goal to improve this method to handle all kinds of occlusion. The tracking thresholds were chosen by us because they worked well for our images but these values can be tightened or relaxed based on the detail of the video sequence and are independent of the design of the algorithm itself. The higher the resolution and more pixel information that exists the more strict the tracking thresholds can be when searching for the best mask translation.

7 Determining Proper and Improper lifts

The primary injury to the body in an improper lift is to the spinal column. When the back is arched or bent it puts uneven stress on the spine. The spine is a collection of vertebra stacked on one another in a column separated by discs that allow the spinal column of bones to bend without grinding each other and wearing away at the bone. When an object is lifted with the spinal column vertical or very near vertical the vertebra and discs compress evenly together. However, when the spinal column is bent or arched the line created by the force of the load on the spine downward compared to the line created by the spinal cord is increased. As the angle is increased the more unevenly the force is being dispersed over the vertebra. This uneven dispersal of force over the vertebra damages the spinal column causing back pain.

Based upon the extraction of the **torso-region** and **object-region** we can make many conclusions on proper and importer lifting. While the person is lifting the object, we can determine how many degrees their back has bent by finding the angle between the ground and the best fit line of the **torso-region**. This method works as long as the lift occurs in a reasonable profile view. Determined by our research, a reasonable view is maintained when the person is not turned more than 45 degrees from the profile position during a lift. Once they have lifted the object the system works with a fixed camera as long as the person stays in view and not in front or behind the object being lifted. The system fails under these two scenarios because the object is tracked based upon edge detection and when the person and the object are blended together in a lift the edge detection fails.

According to an Ergonomic Survival Guide for Laborers created by Cal/OSHA, lifting or carrying a 10 pound object 25 inches from your spine is equivalent to 250 pounds of force on your lower back whereas the same 10 pounds carried 10 inches from your spine is equivalent to 100 pounds of pressure exerted on your lower back. It is difficult to make conclusive measurements of the distance of the object from the spine unless a constant profile view is maintained. Therefore, for now we use the degree the back bends during a lift to detect an improper lift. For now, if the back bends more than 30 degrees during a lift we consider this improper and the person lifting may be putting their self in serious danger. This value can be changed based on more study done in the area of ergonomics.

8 Discussion and Future Work

The sequences in the table varied from lifting single objects including bags, boxes and trash cans to lifting up to three objects in one sequence. We also tested a sequence were a person bent down to lift an object but failed to lift.

Our algorithm works very well for these sequences. Our algorithm depends primarily on good background subtraction. Poor lighting is a primary cause to failure in our algorithm. Occlusion during lifting also causes our method to fail in detecting the object. It is important that the object be more than half in view for the lifted object to be properly tracked. Tracking the torso almost never fails. However, if the object detection fails and the object is brought close to the torso then our method has a high probability of confusing the torso with the object[11].

[11] In all of our sequences, if the object is detected with accurately then the torso is detected accurately as well.

sequence name	frames	lifts	correct	id torso	id object
LIFTLEAVE	600	1	1	100%	100
NOLIFT	720	0	N/A	100%	N/A
SITNOLIFT	288	0	N/A	90%	N/A
LIFTSET	780	1	1	91%	95%
LIFTBAGOCC	1140	1	1	93%	100%
THREELIFTOCC	2220	3	3	98%	93%
LIFTLEAVE2	222	1	1	100%	60%

For now it is very difficult to determine an improper and a proper lift if a reasonable profile view is not maintained.Therefore, we propose that in the future, if four synchronized cameras were placed at 90 degree intervals around the person this would eliminate the problem of not seeing enough of the person to distinguish between an improper and proper lift.

The system works very well for larger objects that range from about a quarter to half the size of a person. Once the object size begins to deviate from this optimal range it is difficult to locate the object. For small objects, sometimes a persons hand covers most of the object during a lift, and this can sometimes cause problems with the current method of tracking which is based primarily on color. Even though the object may be lost, the torso is still maintained and the system knows that an object has been lifted, however the location of the object in this case may be unknown. Judgments can still be made on the lift but not as well as having the exact location of the object with the location of the torso. When an object becomes too large there is not much displacement of the object during a lift and ours system fails here. Our algorithm depends on the nearly complete displacement of the object from its original position.

References

[1] JAVED, O., SHAFIQUE, K., AND SHAH, M. A hierarchical approach to robust background subtraction using color and gradient information. In IEEE Workshop on Motion and Video Computing (December 2002).

[2] COMANICIU, D., AND MEER, P. Robust analysis of feature spaces: Color image segmentation. In IEEE Conf. Computer Vision and Pattern Recognition (1997), pp.750-755.

[3] RAO, C., SHAH, M. AND SYEDA-MAHMOOD, T. Invariance in Motion Analysis of Videos. ACM Multimedia 2003: 518-527.

[4] Aggarwal, J. K. AND Cai, Q. Human Motion Analysis: A review. Computer Vision and Image Understanding: CVIU, 73(3):428-440, 1999.

[5] Bobick, A. Movement, Activity, and Action: The role of knowledge in the perception of motion. Philosophical Transactions of Royal Society of London, B-352:1257-1265, 1997.

[6] GAVRILA, D. M. The visual analysis of human movement: A survey. Computer Vision and Image Understanding: CVIU, 73(1):82-98, 1999.

[7] Ramanan, D. AND Forsyth, D. A. Automatic annotation of everyday movements. Technical report, UCB//CSD-03-1262, UC Berkeley, CA, 2003.

[8] HARITAOGLU, I., Harwood, D. AND Davis, L. S. Ghost: A Human Body Part Labeling System Using Silhouettes. In Proc. ICPR, Brisbane Australia, August 1998.

FEATURES VECTOR FOR PERSONAL IDENTIFICATION BASED ON IRIS TEXTURE

R. P. Moreno
Departamento de Engenharia Elétrica
EESC - USP
Av. Trabalhador Sãocarlense, 400
São Carlos / SP – Brasil
raphael@digmotor.com.br

A. Gonzaga
Departamento de Engenharia Elétrica
EESC - USP
Av. Trabalhador Sãocarlense, 400
São Carlos / SP – Brasil
adilson@sel.eesc.sc.usp.br

Abstract
This work presents a biometric method for identification vector building based on human iris features. The proposed work is based on iris texture features analysis and extraction. The work is divided in 3 steps. In the first, the eye image is preprocessed and Hough Transform for circles does the iris localization and segmentation. In the second step, the iris features information is extracted by a second order statistical approach, using the Haralick's texture features as classification parameters. Finally in the last step, the information is saved in a feature vector that can be used for iris recognition.

Keywords: Haralick, pattern recognition, biometrics, iris, texture.

1 Introduction

Biometry is the group of automatic methods used in people recognition, based in physiological or behavioral features. Examples of behavioral features are signature, gait, voice, etc. Examples of physiological features are fingerprint, face, iris, hand geometry, the veins in ocular retina, etc. One of the biometric advantages, if it is compared with conventional methods, is the possibility of identify, authenticate and localize people without requiring that they carry cards or memorize passwords [1].
Recently, the number of studies and researches in iris recognition has increased significantly. This crescent increase happens because, in identification systems, iris is more efficient, stable and accurate than the others biometric features [2].
The iris is the circular and retractile membrane, which is localized in the center behind the ocular globe. It's situated between the cornea and the anterior part of crystalline, and it has an orifice, the pupil. The fig. 1 shows the iris position in relation to a person eye.

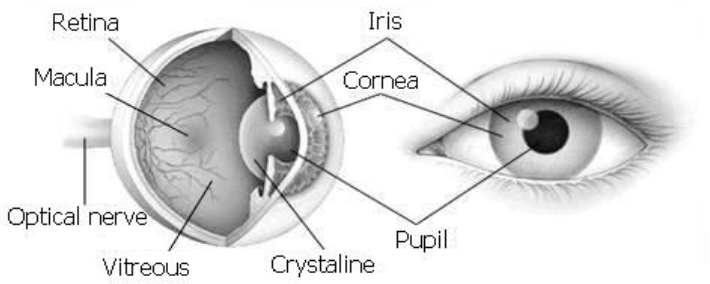


Fig. 1. Eye anatomy.

Formed by a multi-layer structure, the iris has a very complex color and shape pattern. It can be observed in fig. 2.

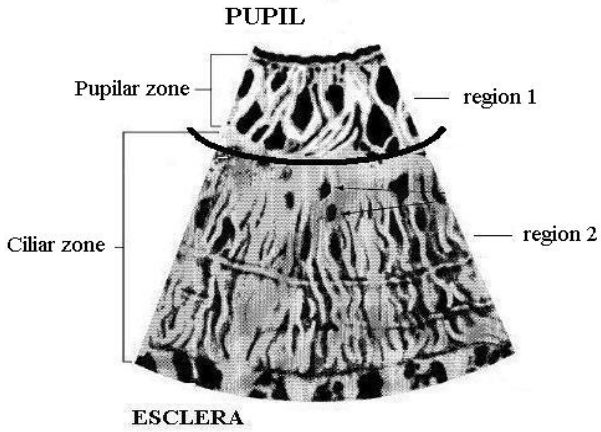


Fig. 2. Iris structure seen in a frontal sector.

The human iris possibility of been used as a biometric signature was first suggested by ophthalmologists [3]. They verified, through clinical experience, that each iris had a very detailed texture.

Recognition biometric systems based on iris study are possible because of some features. The most important of them is the iris uniqueness, which is a result of the chaotic organization of its patterns, established by the initial conditions in the embryonic genetic, [4]. The probability of two people having the same iris pattern is estimated in one in 10^{78} people. As written in [5], the right and left eyes of the same person have different texture patterns.

Another important feature is the iris stability. A normal iris is usually lubricated and preserved by the cornea and aqueous humor, becoming one of the most protected organs in a human body. Besides, the localization, size, shape and orientation remain stable and fixed from about one year of age throughout life [6].

2 Iris localization

The iris localization in an image is the task of find a ring situated between the pupil and the sclera. It is equivalent to finding non-concentric circles which determinate the internal and external borders of the ring. The method used in this work finds the center coordinate and the ray of the pupil, which is the internal border of the iris, through the Hough Transform (HT) for circles [7].

Compared with all others parts in the image, the pupil is much darker. So, after the application of a threshold, followed by an edge detector, the image will be ready to the Hough Transform technique.

The width of the iris ring used is fixed, separating just the iris region near the pupil.

Due to partial iris occlusion by the eyelid and eyelashes, the upper part of the iris ring was removed and it is not used in the algorithm sequence. The fig. 3 shows an original image (a) and the same image after the iris localization (b).

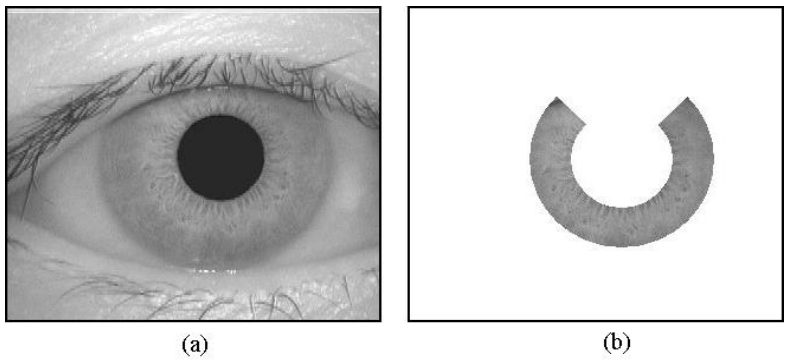


Fig. 3. (a) Original image. (b) Segmented iris.

After the pupil and consequently the iris localization, the system becomes robust to the pupil size and the position of the eye in the image.

3 Haralick's features

In this work, it is proposed an iris feature extraction methodology based in the Haralick's approach [8]. It uses second order statistics, by analyzing the relative position of the image pixels. Through this method, distinct images with equal first order histograms still can be differentiated.
The second order statistical measures are done in probabilities' distributions or co-occurrence matrixes. These matrixes (GLCM – *gray level co-occurrence matrix*) are bi-dimensional representations showing the spatial occurrence organization of the gray levels in an image. They represent a bi-dimensional histogram of the gray levels, where fixed spatial relation separates couples of pixels, defining the direction and distance *(d,θ)* from a referenced pixel to its neighbor. To build these matrixes, the couple of pixels' variation is done in the following angles: 0°, 45°, 90° e 135°, originating four distinct co-occurrence matrixes.
After computing the co-occurrence matrixes, several second orders statistical calculus can be calculated, including the Haralick's features. These are the features used in this work:

- **Second Angular Moment (SAM)**: measures the local homogeneity of gray levels in an image. The SAM equation is given by:

$$SAM = \sum_{i=0}^{n-1}\sum_{i=0}^{n-1}\left[P(i,j,d,\theta)\right]^2 \quad (1)$$

- **Contrast**: it measures the local quantity of gray levels in an image. The Contrast equation is given by:

$$Contrast = \sum_{i=0}^{n-1}\sum_{i=0}^{n-1}(i-j)^2 P(i,j,d,\theta) \quad (2)$$

- **Entropy:** also called as dispersion degree of the gray levels, it measures together with the SAM, the homogeneity in an image. The Entropy equation is given by:

$$Entropy = \sum_{i=0}^{n-1}\sum_{i=0}^{n-1} P(i,j,d,\theta)\log_2\left[P(i,j,d,\theta)\right] \quad (3)$$

- **Inverse Difference Moment (IDM)**: The IDM equation is given by:

$$IDM = \sum_{i=0}^{n-1}\sum_{i=0}^{n-1}\frac{1}{1+(i-j)^2}P(i,j,d,\theta) \quad (4)$$

- **Correlation**: it represents the linearity dependence of gray levels in an image. The Correlation equation is given by:

$$Correlation = \frac{\sum_{i=0}^{n-1}\sum_{i=0}^{n-1} i \cdot j \cdot P(i,j,d,\theta) - \mu_X \mu_Y}{\sigma_X \sigma_Y} \quad (5)$$

Where,

$$\mu_X = \sum_{i=0}^{n-1}\sum_{i=0}^{n-1} i \cdot P(i,j,d,\theta), \qquad \mu_Y = \sum_{i=0}^{n-1}\sum_{i=0}^{n-1} j \cdot P(i,j,d,\theta)$$

$$\sigma_X = \sqrt{\sum_{i=0}^{n-1}\sum_{i=0}^{n-1} i^2 \cdot P(i,j,d,\theta) - \mu_X}, \quad \sigma_Y = \sqrt{\sum_{i=0}^{n-1}\sum_{i=0}^{n-1} j^2 \cdot P(i,j,d,\theta) - \mu_Y}$$

And, μ_x and μ_y represent the mean in X and Y direction, and σ_y and σ_y represent the variance.

4 Feature vector

In this work, the segmented iris is divided into six sectors having the same size, as showed in fig. 4. The number of sectors was defined to increase the classifying method efficiency through the texture features.

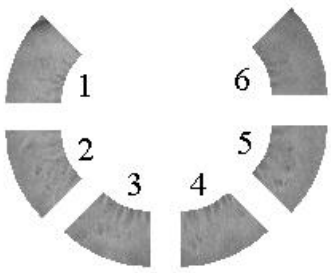


Fig. 4. *Segmented iris divided in six sectors.*

For each sector, the five Haralick's features are calculated resulting in a feature vector with 30 values. This vector will be saved in the database or used in an identification or authentication process.

5 Image database

The image database used to test the algorithm, CASIA version 1.0 [9], was developed by the *Iris Recognition Research Group - National Laboratory of Pattern Recognition* (NLPR) from the Institute of Automation, Chinese Academy of Sciences. The dataset has images with 256 gray levels, and resolution of 320x280 pixels, captured through a digital optical sensor also developed by the NLPR. There are 756 images of 108 eyes from 80 people.
In this dataset, seven images were taken from each iris, in two different moments. In the first one, three images were taken and in the second moment, one month later, more four images were taken.

6 Tests and results

The algorithm finds the proximity of two irises calculating the normalized Euclidian distance of the two features vectors, as described in equation 6.

$$D(A,B) = \sqrt{\sum_{i=1}^{30} \frac{\left(|A_i - B_i|\right)^2}{A_i}} \quad (6)$$

The fig. 5 shows an example of the distance calculus between an iris and the others 107 resting in the database.

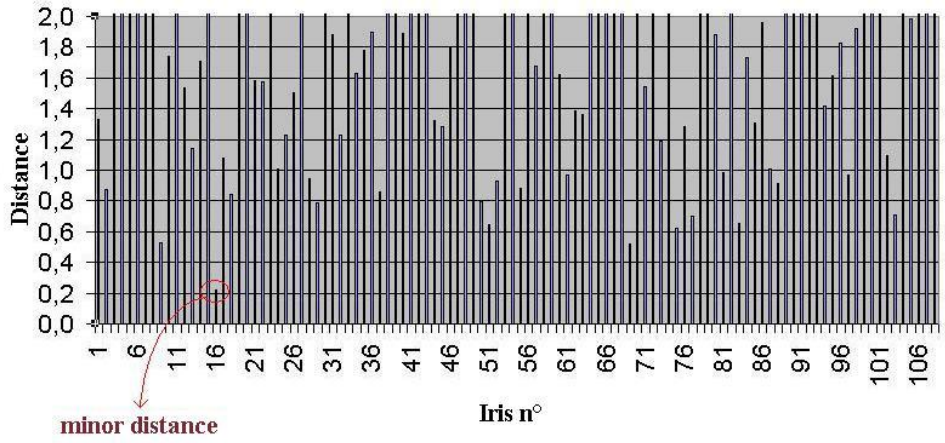


Fig. 5. Euclidian distance example.

For each comparison between two irises' images, the algorithm returns a number. To show if the vectors are from the same iris, the algorithm compare the value returned with a t (*threshold*) value, previous established. With this information, it's possible to evaluate the system accuracy varying the *t* value and building the ROC curve (*receiver operating characteristic*).
To build the ROC curve, it was generated a dataset with the mean feature vector taken from each one of the 108 irises of the database. The mean vectors were obtained calculating the means among the seven features vectors from each image of the same iris.
After that, for each *t* value it was done an authentication try between each database image and the others 107 irises. As the database has 756 images, 81648 authentication tries were done. During the authentication tries, the number of false accepted (FA) and false rejected (FR) were found. The Table 1 and the figure 6 show the FA and FR probability's distribution with the *t* variation.

Table 1. False accepted probability, $P_{(FA)}$ and false rejected probability, $P_{(FR)}$.

t	**$P_{(FA)}$**	**$P_{(FR)}$**
0,00	100,000%	0,000%
0,05	47,153%	3,571%
0,10	43,496%	3,704%
0,15	39,726%	4,101%
0,20	35,634%	4,762%
0,25	31,469%	6,085%
0,30	27,150%	7,011%
0,35	22,927%	7,804%
0,40	18,671%	9,392%
0,45	14,532%	11,243%
0,50	10,673%	13,889%
0,55	7,347%	17,989%
0,60	4,604%	22,354%
0,65	2,479%	27,910%
0,70	1,024%	35,185%
0,75	0,307%	47,222%
0,80	0,066%	62,963%
0,85	0,002%	78,704%

0,90	0,000%	94,577%
0,95	0,000%	100,000%

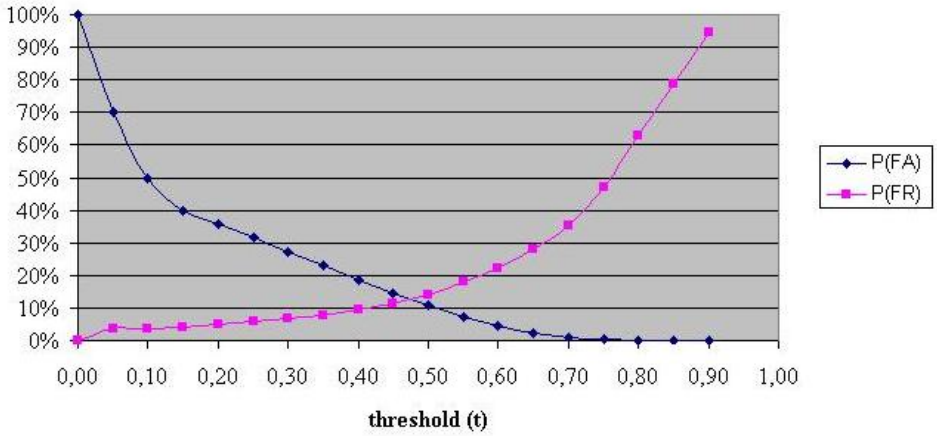


Fig. 6. False accept probability, $P_{(FA)}$ and false reject probability, $P_{(FR)}$.

The ROC curve, which represents the system accuracy, is showed in fig. 7 and was built with the $P_{(FA)}$ e $P_{(FR)}$ values showed in figure 6.

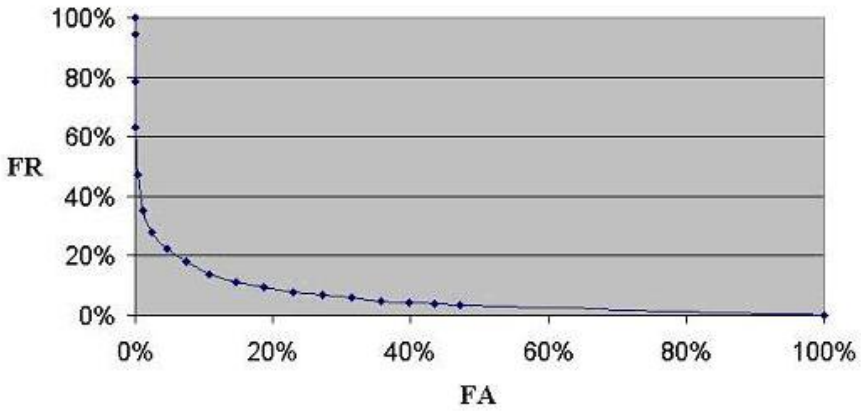


Fig. 7. ROC curve.

Conclusion

The ROC curve analysis validates the Haralick's features for using as a biometric feature extraction of human being, because they can reproduce the iris unique feature. Also, it is possible to conclude that the way chosen to divide the iris ring is an efficient method to obtain a uniform texture region. Another important point is that in the majority of the cases, the false accepted and the false rejected were obtained due to some kind of fail in the iris image. The partial occlusion and the lack of focus were the principal fail reasons. Fig. 8 shows an eye image with the iris very obstructed, what turns its identification a hard job.

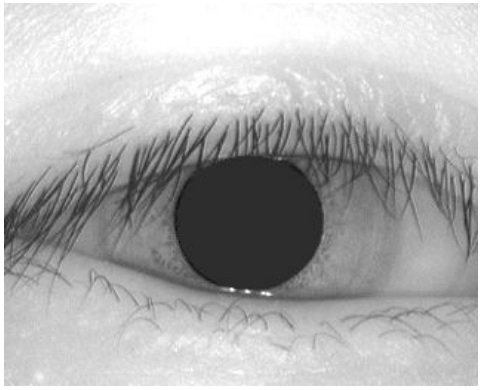


Fig. 8. Partial occlusion of the iris.

The identification also becomes difficult when the images, used to build the mean feature vector, have many differences.

References

[1] Negin, M. et al. (2000). An iris biometric system for public and personal use. IEEE, p.70-75.
[2] Jain, A.K. et al. (1999). Personal Identification in a network society. Norwell, MA - Kluwer.
[3] Adler, F.H. (1965). Physiology of the eye: Clinical application. The C. V. Mosby Company, 4a edição, Londres.
[4] Daugman, J. (1993). High confidence visual recognition of persons by a test of statistical independence. IEEE Transactions on Pattern Analysis and Machine Intelligence, v.15, no. 11, p.1148-1161.
[5] El-Balkry, H.M. (2001). Human iris detection using fast cooperative modular neural nets. Neural works, Proceedings of International Joint Conference on IJCNN'01, v.1, p.577-582.
[6] Willians, G.O. (1997). Iris recognition technology. IEEE AES Systems Magazine, abril 1997, p.23-29.
[7] Haralick, R.M.; Shanmugan, M.K. (1973). Computer classification of reservoir sandstones. IEEE Transactions on Geoscience Electronics, v.11, no. 4, p.171-177, Oct.
[8] Hough, P.V.C. (1962). Methods and means for recognizing complex patterns. U.S. Patent 3.069.654.
[9] CASIA. Iris Image Database versão 1.0, http://www.sinobiometrics.com.

Three-dimensional Reconstruction Using Silhouette Images From Random Angles

K. Kawasue
Department of Mechanical Systems Engineering, University of Miyazaki
1-1 Gakuen Kibanadai Nishi, MIYAZAKI
889-2192 Japan
kawasue@cc.miyazaki-u.ac.jp

N. Taguchi
Industrial Technology Center of Nagasaki
2-1303-8 Ikeda, Ohmura, NAGASAKI
856-0026 JAPAN
taguchi@tc.nagasaki.go.jp

Kerrison David
Tenchimon L.L.C.
3-7-39 Shimizu, MIYAZAKI,
880-0021 JAPAN
kerrison@tenchimon.com

Abstract
In general a three-dimensional measurement system is established by considering the geometric configuration of the system, such as CCD camera and structured light, etc., because the system measurement is based on the principle of triangulation. Once the configuration has been determined, it must be maintained until the measurements have been completed. The main disadvantage of such systems is the dead-angles that inherently exist in a single image taken from a fixed angle.

This paper proposes a new approach to three-dimensional reconstruction of an object shape using multiple silhouette images that are taken from arbitrary random angles. The technique can be realized by utilizing a magnetic sensor that is attached to the CCD camera. The distinctive feature of this system is that it reduces the limitation of the target in terms of size, shape and obstruction, etc. but more importantly it reduces the dead-angle problem of measurement.

Experimental results show the feasibility of our system.

Keywords: Silhouette, Magnetic Sensor, Measurement, Three-dimensional

1. Introduction

This paper addresses a new approach to 3D reconstruction of an object's shape using a simple setup. In general, the object to be measured is digitalized from many images that include structured light (slit-ray) information taken from different positions. Quantitative measurement is established by considering the geometric configuration between the CCD camera and the structured light [1]-[2]. Although 3D measurement methods have achieved a high level of satisfaction in limited fields, unmeasurable areas exist in many cases such as behind the object.

In order to reduce the unmeasurable area, a practical hand held laser scanning system has been developed by BC McCallum, etc [3]-[5]. An electromagnetic spatial locator determines the position and orientation of a hand-held assembly during the scan. Three-dimensional measurement can be achieved by just pointing the laser slit at the target.

The problem with systems that utilize a laser, e.g., a hand held laser scanner, is that the results can be affected by the color of the object. An object with a black or mirror surface cannot be measured since the CCD camera does not detect the laser light clearly enough from the surface.

It is well known, that the silhouette of an object also depicts the shape of object and can be determined without using a laser slit. Some researchers use the silhouette information for shape measurement [6]-[9]. Since this does not rely on the use of a laser, the measurement is not influenced by the color of the object, and the system is therefore simple and safe. Quantitative measurement can be achieved by knowing the relative configuration between the CCD camera and the object. In these systems, a turntable or rotating arm with an attached CCD is often used to

reduce the dead-angle. The use of a turntable or rotating arm maintains the relative configuration between the CCD camera and the object while taking quantitative measurements. However, the restriction of relative movements between the CCD camera and the object limits the size and shape of the target objects.

In this paper, a measuring system that enables arbitrary free movement of the CCD camera during measurement is introduced. Many silhouette images are recorded from arbitrary different angles. As an electromagnetic spatial locator is directly attached to the CCD camera, the sensor measures the three-dimensional position and the orientation of the CCD camera: at a frequency of 60Hz. Multiple images of the object are taken at arbitrary free angles, including measurement of the location and orientation of the CCD camera. The information from the different image views is then combined to reconstruct the 3D object on a computer display with minimum loss of data.

A typical application of this system, the measurement of agricultural products, is introduced in this paper. Recently in Japan, the quantitative measurement of agricultural products has been made a requirement to guarantee and maintain quality control. The proposed system has been successfully applied to quantify a product shape. Experimental results show the feasibility of the system.

2. System Set-up

The system set-up, shown in Fig.1, is composed of a CCD camera, a 3D magnetic spatial locator (Polhemus Fastrack) and a personal computer. The receiver of the spatial locator is fixed on the CCD camera. The location and orientation of the CCD camera is measured at a frequency of 60Hz by the 3D magnetic spatial locator. Each CCD pixel, at a particular row and column, on the outline of the silhouette can be represented in camera coordinates $\boldsymbol{p}_c$ (the origin of which is on the focal point of the camera) by considering the camera parameters such as the focal length, resolution, etc. As we know the relationship between the receiver coordinates and camera coordinates, the camera coordinates $\boldsymbol{p}_c$ can be converted into the corresponding receiver coordinates $\boldsymbol{p}_r$.

The spatial locator's transmitter, which is positioned in the vicinity of the CCD camera, is fixed relative to the object. The spatial locator's control unit returns the translation vector $\boldsymbol{t}$ and the rotation matrix $\boldsymbol{M}$ of the receiver coordinate system relative to that of the transmitter. The point in transmitter coordinates $\boldsymbol{pt}$ can then be computed using the spatial locator output data from

$$Pt = \boldsymbol{t} + M^t \boldsymbol{p}_r \qquad (1)$$

Where the $\boldsymbol{p}_r$ is the point in receiver coordinates [3].
The backboard positioned behind the object is used to increase the contrast between object and back-image.

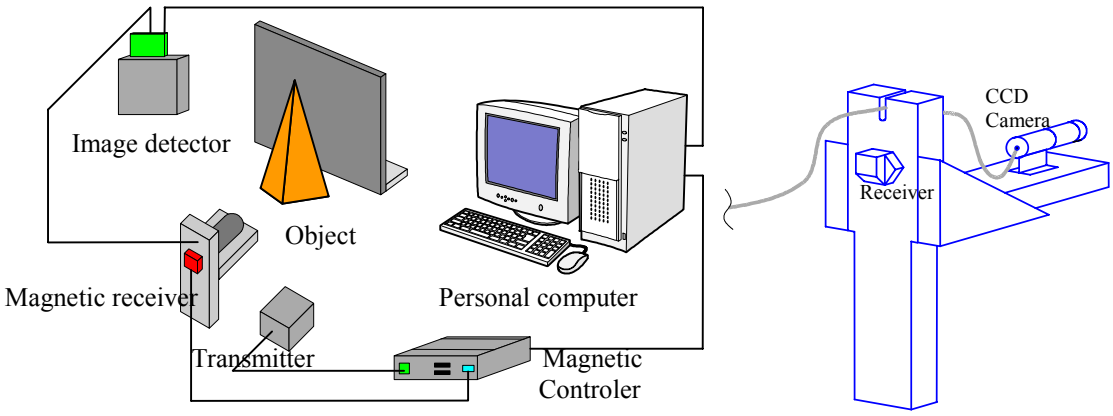
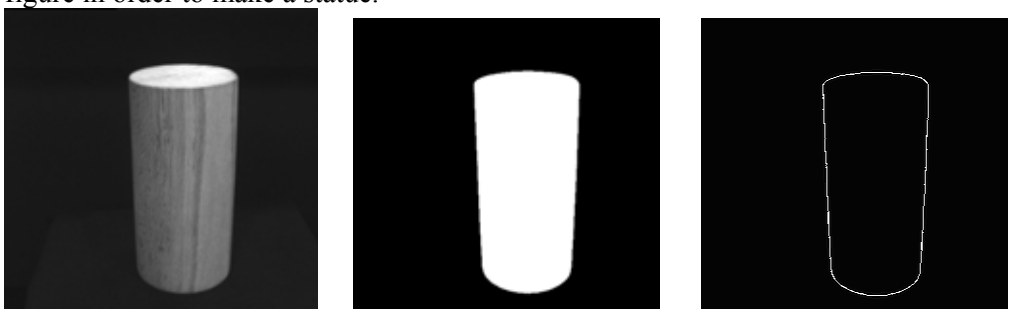


Fig.1 system setup

3. Object Reconstruction From Silhouette Images

The outline image of the object is used in the reconstruction method. An example of the image is shown in Fig.2c. Since the images are two dimensional, the real size of the image cannot be estimated without extra information such as depth/distance etc. However, the hatched area, see Fig.3, which has the same cross-section as the outline of the object, can be selected and used to estimate the volume of the object within the hatched area. The rate of expansion of this area is determined by the focal length of the camera. Our method reconstructs the object from multiple estimated volumes obtained at arbitrary angles. Then each estimated volume is converted into the transmitter's coordinates and the final volume can be extracted by deleting the outsides of the each estimated volume, see Fig.5. The technique can be likened to the way a sculptor would sculpt a 3D figure in order to make a statue.



a. Object b. binary image c. outline of object

Fig.2 Example of extracted silhouette image.

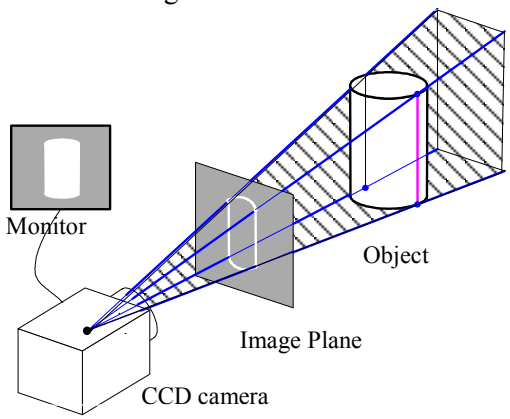


Fig.3 Hatched area used to estimate the volume of an object.

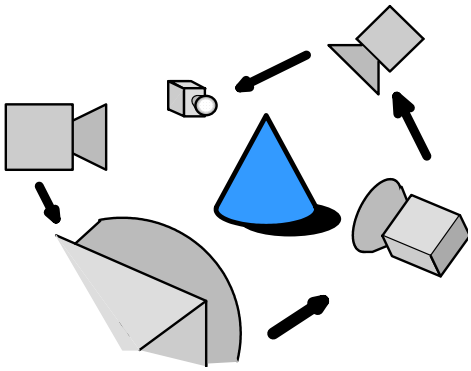


Fig. 4 Image acquisition from arbitrary free angle.

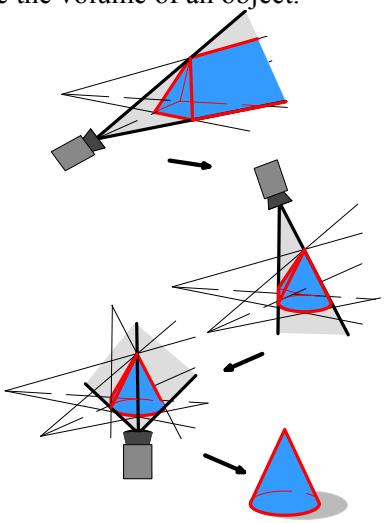


Fig.5 The extraction of final volume of the object

The outline of the object is used to determine the volume of the object. Fig.6 shows projection lines that pass through the outline image in the image plane and the object. The relationship between the point (u,v) on the outline in the camera coordinates and the relative coordinates (x,y,z) on the receiver is expressed as follows.

$$s\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} k_{11} & k_{12} & k_{13} & k_{14} \\ k_{21} & k_{22} & k_{23} & k_{24} \\ k_{31} & k_{32} & k_{33} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (2)$$

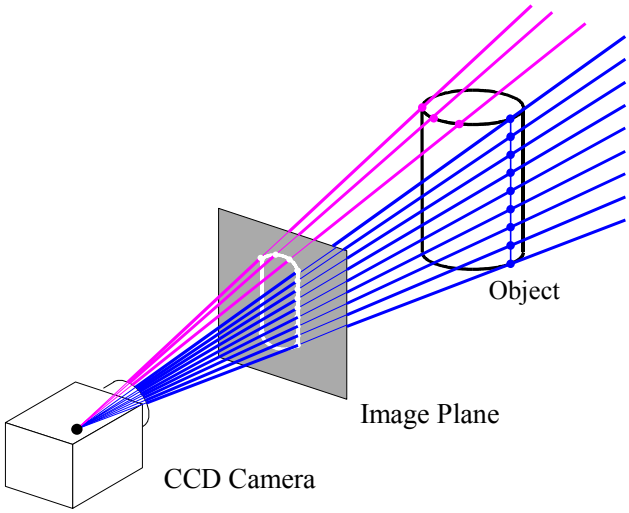


Fig.6 Lines pass the outline and Focus point

Parameters (k_{11}-k_{33}) are introduced by considering the rotation and displacement. This equation can be expressed also as following by expanding.

$$\begin{cases} u = k_{11}x + k_{12}y + k_{13}z + k_{14} - k_{31}ux - k_{32}uy - k_{33}uz \\ v = k_{21}x + k_{22}y + k_{23}z + k_{24} - k_{31}vx - k_{32}vy - k_{33}vz \end{cases} \quad (3)$$

The eleven parameters (k_{11}-k_{33}) are determined from the calibration procedure by feeding some corresponding coordinate pairs of positions that are already known. The calibration setup is shown in Fig.7. The image of the scale board is recorded by the CCD camera and is displayed on the computer display. A mouse device and a keyboard then used to register the positional pairs between the camera coordinates and the receiver coordinates, respectively.

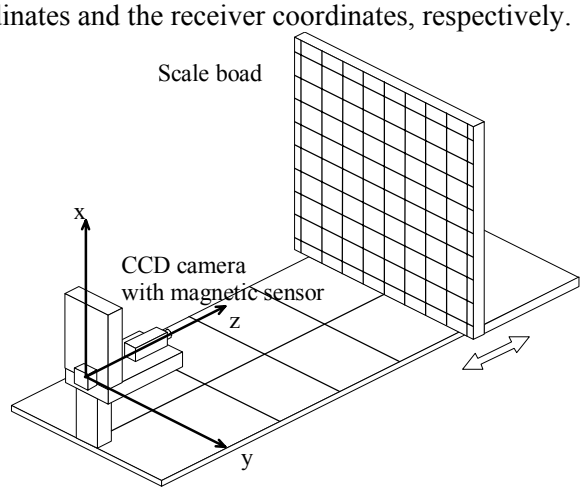


Fig.7 Calibration setup

Equation (3) indicates the two planes, and the intersectional line between these planes indicates the line that passes through focal point and the outline of the object. The line can be expressed as follows, in receiver coordinates.

$$\begin{cases} x = f \cdot t + x_1 \\ y = g \cdot t + y_1 \\ z = h \cdot t + z_1 \end{cases} \quad (4)$$

Where (f,g,h) is a direction vector and (x_1,y_1,z_1) is the translation vector. The receiver coordinates can be converted to the transmitter coordinates using the following formula.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = RPY(\Psi,\Theta,\Phi) \begin{bmatrix} f \cdot t + x_1 \\ g \cdot t + y_1 \\ h \cdot t + z_1 \end{bmatrix} + \begin{bmatrix} w_x \\ w_y \\ w_z \end{bmatrix} \quad (5)$$

Where (w_x,w_y,w_z) is the translation vector of the receiver coordinate system and $RPY(\Psi, \Theta, \Phi)$ is the rotation matrix of the receiver coordinate system relative to that of the transmitter. The resultant lines produced by (5) construct the outside of the estimated volume on the transmitter's coordinates. Fig.8 shows the cross-section of the estimated volume on the horizontal plane z_1. The cross section can be determined by finding the intersection points between the lines from focal point to the outline and the plane z_1. The cross-section can be determined by combining the estimated volume obtained at different angles. Fig.9 shows the cross section of the measured cylinder. The whole volume of the object can be reconstructed by accumulating the cross-sections on different planes z_n. Fig.10 shows the whole volume of the measured cylinder. The average error of the measurement for this object was 1.5 mm.

4 Shape Measurements Of Agricultural Products

Generally, in the evaluation of agricultural products, a human inspector judges the quality of the product by comparing them to a known standard sample. In this manual method, however, individual differences occur in the evaluation. For this reason, the quantification of agricultural products is required.

The purpose of this experiment is to demonstrate that the quantification of a product shape using our computer vision system. Fig.11 shows the result of reconstruction of a banana on the computer using the measured data. It takes about 1 minute to measure the outline of the banana by an inexperienced operator. However, with training and optimized computer programming this time could be greatly reduced.

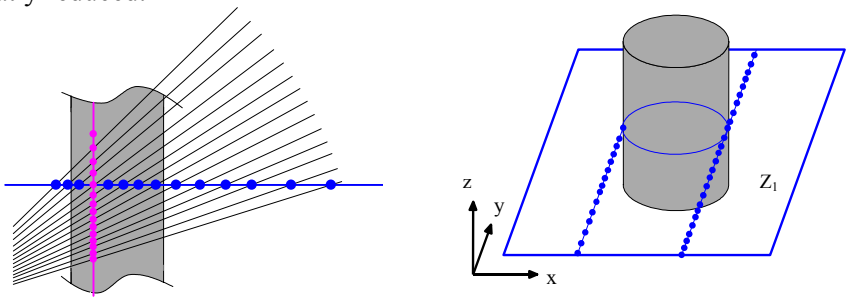


Fig.8 Determination of the cross-section of estimate volume

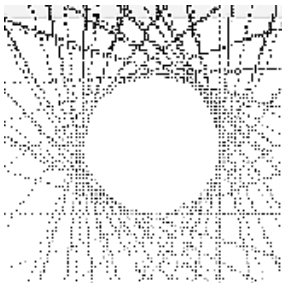


Fig.9 The estimated cross-section of the cylinder

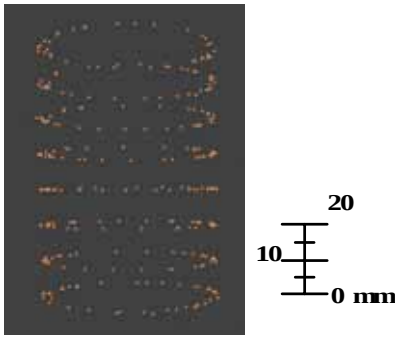


Fig.10 Whole volume of cylinder

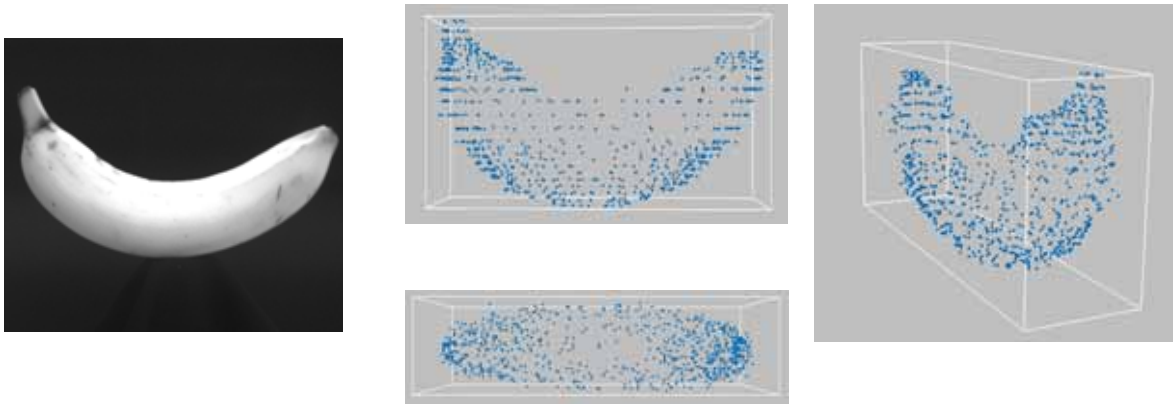


Fig.11 Measurement of banana

5 Conclusion

A three-dimensional measurement system which enables a three dimensional reconstruction on a computer using silhouette images recorded from arbitrary free angles has been introduced. Since measurement device consists only of a CCD camera with a magnetic locator, the system is compact and flexible for many different kinds of objects to be measured. An operator can easily change the position and angle of the CCD camera according to the shape and the size of the object. The CCD camera should to be positioned closer to the object when the size of the object is small and it to be farther away for larger objects. To a degree the results depends on the operator's experience, but once they have become accustomed to using it, it would be very good tool for measurement.

The measurement of agricultural products was introduced by way of example of the type of applications suitable for our system. However, the system has possibilities that can be applied in various other fields.

Reference

[1] Three-dimensional computer vision(1996), A geometric viewpoint, Olivier Faugeras

[2] Computer vision: Theory and industrial application(1992), Springer-Verlag, 1992

[3] B. C. McCallum, W. R. Fright, M. A. Nixon and N. B. Price(1996), A Feasibility Study of Hand-held Laser Surface Scanning, *Proc. of Image and Vision Computing NZ*, Lower Hutt, pp. 103–108.

[4] M. A. Nixon, B. C. McCallum, W. R. Fright, and N. B. Price(1998). The effects of metals and interfering fields on electromagnetic trackers. Presence, *MIT Press*, Volume 7, Number 2, pp. 204-218.

[5] B. C. McCallum, M. A. Nixon, N. B. Price and W. R. Fright(1998), Hand-held Laser Scanning In Practice, *Proc. of Image and Vision Computing NZ*, The University of Auckland, pp. 17-22.

[6] H. Baker(1977), Three-dimensional modeling. *In Fifth International Joint Conference on Artificial Intelligence*, pages 649–655.

[7] B.G. Baumgart(1987), Geometric modeling for computer vision. *Technical Report AIM-249*, Artificial Intelligence Laboratory, Stanford University

[8] W. N. Martin and J. K. Aggarwal(1987). Volumetric description of objects from multiple views. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5(2):150–158.

[9] P. Srivasan, P. Liang, and S. Hackwood(1990). Computational geometric methods in volumetric intersections for 3d reconstruction. *Pattern Recognition*, 23(8):843–857.

A PERFORMANCE CHARACTERISATION IN ADVANCED DATA SMOOTHING TECHNIQUES

Michael Lynch*, Kevin Robinson, Ovidiu Ghita, Paul F. Whelan
Vision Systems Group
Dublin City University, Ireland

Abstract

A comparison paper is presented to evaluate the results from five smoothing filters. The filters are linear, nonlinear isotropic and nonlinear anisotropic designed to smooth homogeneous areas while preserving the higher moments in the data. The methods are outlined and then evaluated on the extent to which edge information is preserved and unwanted noise is suppressed.
Keywords*: Smoothing, adaptive, Savitzky-Golay, diffusion*

1 Introduction

The noisy images presented in medical imaging has led researchers to investigate methods of smoothing image noise while maintaining important information such as edges.

There are two main types of smoothing, linear and non-linear. Both of these types have being extensively studied in literature. Traditional linear filters such as mean, average and Gaussian attempt to remove noise by replacing each pixel by an average or weighted average of its spatial neighbours [2]. While this reduces the amount of noise present in the image, it also has the disadvantage of removing or blurring the edges. Nonlinear filters, the most common being the median filter, modifies the value of the pixel by some nonlinear function of the pixel value and its spatial neighbours. Nonlinear filters maintain the edges but the filtering results in a loss of resolution by suppressing fine details. More recently, the use of edge-based diffusion has emerged [1, 3, 4, 5, 6]. These filters require a trade-off between smoothing efficiency, preservations of discontinuities and the generation of artifacts. In short the diffusion or smoothing term is a variable over space and time and in [3], this term is a function of the magnitude of the gradient intensity at the point in question. Gerig [4] extended this case to 3D and performed the diffusion on medical volumes. Perona and Malik's [3] diffusion has the disadvantage that it stopped the diffusion at edges, this was advanced by [7] by permitting diffusion along the direction of the edges making it anisotropic.

This paper compares five filters. The linear Savitzky-Golay filter is a convolution of the image with the least-squares fitting of a polynomial. The basic Gaussian filter is a convolution with a Gaussian mask, nonlinear adaptive filtering which filters the image but smooths less in areas of local discontinuities and high spatial variance. Nonlinear diffusion, which again smooths with an exponential with no smoothing occurring where the gradient has high values. Finally anisotropic Gaussian smoothing which uses a scaled and shaped Gaussian mask to smooth along the direction of high gradients and never across the gradients.

*Corresponding author. *E-mail address:* lynchm@eeng.dcu.ie

2 Savitzky-Golay Filter

The Savitzky-Golay [8] smoothing filter was introduced for smoothing data and for computing the numerical derivatives. The smoothed points are found by replacing each data point with the value of its fitted polynomial. The process of Savitzky-Golay is to find the coefficients of the polynomial which are linear with respect to the data values. Therefore the problem is reduced to finding the coefficients for fictitious data and applying this linear filter over the complete data. The size of the smoothing window is given as $N \times N$ where N is odd, and the order of the polynomial to fit is k, where $N > k + 1$.

$$f(x_i, y_i) = a_{00} + a_{10}x_i + a_{01}y_i + a_{20}x_i^2 + a_{11}x_iy_i + a_{02}y_i^2 + ... + a_{0k}y_i^k \tag{1}$$

We then want to fit a polynomial of type in Eq. 1 to the data. Solving the least squares we can find the polynomial coefficients. We start with the general equation, $A \cdot a = f$, where a is the vector of polynomial coefficients $a = (a_{00} \quad a_{01} \quad a_{10} \quad .... \quad a_{0k})^T$ We can then compute the coefficient matrix as follows, $(A^T \cdot A) \cdot a = (A^T \cdot f)$, which in least squares can be written as $a = (A^T \cdot A)^{-1} \cdot (A^T \cdot f)$.

Due to the linear-squares fitting being linear to the values of the data, the coefficients can be computed independent of data. The general coefficient matrix becomes $C = (A^T A)^{-1} A^T$. C can then be reassembled back into a traditional looking filter of size $N \times N$. In order to smooth the image the first coefficient is used, higher order coefficients are used to calculate derivatives. The advantage of the Savitzky-Golay filter has the ability to preserve higher moments in the data and thus reduce smoothing on peak heights. In more homogeneous areas the smoothing approaches an average filter.

3 Adaptive Smoothing

The algorithm for adaptive smoothing implemented in this paper is adapted from Chen [5]. The technique measures two types of discontinuities in the image, local and spatial. From both these measures a less ambiguous smoothing solution is found. In short, the local discontinuities indicate the detailed local structures while the contextual discontinuities show the important features.

In order to measure the local discontinuities for each pixel the average of the ∇'s around the pixel in the horizontal, vertical and diagonal is calculated to be E_{xy}. In order to measure the contextual discontinuities, a spatial variance $\sigma^2_{xy}(R)$ is employed in a square kernel $N_{xy}(R)$.

This value of sigma is then normalised to $\tilde{\sigma}^2_{xy}$ between the minimum and maximum variance in the entire image. A transformation is then added into $\tilde{\sigma}^2_{xy}$ to alleviate the influence of noise and trivial features. It is given a threshold value of $\theta_\sigma = (0 \leq \theta_\sigma \leq 1)$ to limit the degree of contextual discontinuities.

Finally, the actual smoothing algorithm runs through the entire image updating each pixels intensity value I^t_{xy}, where t is the iteration value.

$$I^{t+1}_{xy} = I^t_{xy} + \eta_{xy} \frac{\Sigma_{(i,j)\in N_{xy}(1)/\{(x,y)\}} \eta_{ij}\gamma^t_{ij}(I^t_{i,j} - I^t_{x,y})}{\Sigma_{(i,j)\in N_{xy}(1)/\{(x,y)\}} \eta_{ij}\gamma^t_{ij}} \tag{2}$$

where,

$$\eta_{ij} = \exp(-\alpha\Phi(\tilde{\sigma}^2_{ij}(R), \theta_\sigma)), \tag{3}$$

$$\gamma^t_{ij} = \exp(-E^t_{ij}/S) \tag{4}$$

The variables S and α determine to what extent the local and contextual discontinuities should be preserved during smoothing. If there are a lot of contextual discontinuities in the image then the value of η_{ij} will have a large influence on the updated intensity value. On the other hand, if there are a lot of local discontinuities then both γ_{ij} and η_{ij} will have the overriding effect, as η_{ij} is used for gain control of the adaption.

4 Nonlinear Diffusion Filtering

The standard blurring operation involving Gaussian filtering attempts to remove the noise at the expense of poor edge preservation and is given as;

$$S_{x,y} = I_{x,y} \circ Gauss(x, y, \sigma) \tag{5}$$

where S is the filtered image, I is the input image, $\circ$ implements the 2D convolution, $Gauss()$ is the 2D Gaussian function where σ is the scale parameter. The smoothing becomes more pronounced for higher values of the scale parameter but we can notice a significant attenuation of the signal at image boundaries. This result is highly undesirable for many applications like image segmentation and edge tracking where a precise identification of object boundaries is required.

To alleviate the problems associated with the standard Gaussian smoothing technique, Perona and Malik [3] proposed an elegant smoothing scheme based on non-linear diffusion. In their formulation the blurring would be performed within homogeneous image regions with no interaction between adjacent or neighbouring regions that share a common border. The non-linear diffusion procedure can be written in terms of the derivative of the flux function, $\phi(\nabla I) = \nabla I \cdot D(\|\nabla I\|)$, where ϕ is the flux function, I is the image and D is the diffusion function. This equation can be implemented in an iterative manner and the expression required to implement the non-linear diffusion is illustrated in Eq. 6.

$$I_{x,y}^{t+1} = I_{x,y}^{t} + \lambda \sum_{R=1}^{4} [D(\nabla_R I) \nabla_R I]^t \tag{6}$$

where I^t represents the image at iteration t, R defines the 4-connected neighbourhood, D is the diffusion function, ∇ is the gradient operator that has been implemented as the 4 connected nearest-neighbour differences and λ is a parameter that takes a values in the range $0 < \lambda < 0.25$.

The diffusion function $D(x)$ should be bounded between 0 and 1 and should have the peak value when the input x is set to zero. This would translate with no smoothing around the region boundary where the gradient has high values. In practice, a large number of functions can be implemented to satisfy this requirement and in our implementation we were using the exponential function proposed by Perona and Malik [3], $D(\|\nabla I\|) = e^{-(\frac{\|\nabla I\|}{k})^2}$,where k is the diffusion parameter. The parameter k selects the smoothness level and the smoothing effect is more noticeable for high values of k.

5 Anisotropic Gaussian Smoothing

An anisotropic filter based on the familiar Gaussian model was implemented in order to provide edge enhancing, directional smoothing. The goal was to develop a versatile smoothing filter based on a straightforward and highly adaptable form. The approach reduces to convolution with a scaled and shaped Gaussian mask, where the determination of the mask weights becomes the key step governing the performance of the filter. By calculating the local greyscale gradient vector and favouring smoothing along the edge over smoothing across it we can achieve an effective boundary preserving filtering approach, where regions are homogenized while edges are retained.

The weight $wt(\vec{pq}, \nabla u)$ at each location in the mask is a function of the local gradient vector at the centre of the mask and the distance of the current neighbour from that centre. There is a large number of possibilities for the formulation of the mask weight calculation, based on the desired form for the non-linear and anisotropic components of the filter. The weight for some neighbour q is calculated as a function of the gradient of point p, at the mask origin, and the distance from the origin to the neighbour q. The relationship used in our approach is given in Eq. 7, where $\vec{pq}$ is the vector from the mask centre point p to some neighbour q, ∇u is the gradient vector at p, λ is the scale parameter, controlling smoothing strength, and μ is the shape parameter, controlling anisotropy. When μ equals zero the anisotropic term

$(\frac{\vec{pq}\cdot\nabla u}{\lambda})^2(2\mu+\mu^2)$ disappears and the filter reduces to the non-linear, isotropic form, where smoothing decreases close to strong edges but is applied equally in all directions, at any given location in the image.

$$wt(\vec{pq},\nabla u) = e^{-((\frac{\|\vec{pq}\|\|\nabla u\|}{\lambda})^2+(\frac{\vec{pq}\cdot\nabla u}{\lambda})^2(2\mu+\mu^2))} \tag{7}$$

The images in Figures 1 and 2 illustrate the operation of the anisotropic filter. As the smoothing strength and the number of iterations is increased more noise and small features are eliminated, but even in extreme cases the most important edges in the image are well preserved in both location and strength.

6 Experiments and Results

The results of each filter are assessed by their ability to smooth homogeneous areas while preserving the areas with higher moments. Smoothing of homogeneous areas is measured using the standard deviation while the preservation of edges is measured using the strength and spread of the edge in the filtered images. The filters are tested on two images, see figures 1 and 2, the first image of a laboratory having a high SNR (signal-noise-ratio) and high CNR (contrast-to-noise-ratio) with a high density of edges. The second medical image has a much lower SNR and CNR. Parameters were chosen to give the optimal results on visual inspection. Visual results are presented in figures 1 and 2. The standard deviation is

Figure 1: Results from each of the smoothing filters, top-row, l-to-r is the original image, image after Savitzty-Golay and Gaussian. Bottom-row, Adaptive, Nonlinear Diffusion and Aniostropic Gaussian.

measured in a 7×7 window over the entire original image. From these values 25% of the highest values were eliminated as belonging to edges in the image and 25% of the lower values as having no significant texture to smooth. The standard deviation for each of the filtered images is then taken at the same positions. The results are presented in Table 1. For the laboratory image, Adaptive smoothing gives the best results followed by the two other non-linear filters. Both linear Savitzky-Golay and Gaussian filters have the highest deviation after smoothing. In the medical image there are more significant differences

	Laboratory Image			*MR Image*		
	SD	Edge height	Edge width	SD	Edge height	Edge width
Original	57.4	31	2.26	277.7	219	2.04
Savitzky-Golay	40.8	23	2.5	61.23	158	2.48
Gaussian	41.0	15	4.4	102.8	196	2.16
Adaptive	24.2	26	2.13	42.99	211	2.00
Diffusion	27.7	25	2.17	69.63	214	2.00
Anisotropic	31.9	30	2.17	35.05	219	1.99

Table 1: Shows the standard deviation (SD), edge strength and edge spread on both images after each filtering. The edge strength and edge spread are taken from histograms in figure 3.

with the anisotropic and adaptive smoothing operators giving the best results while the Gaussian performs worst in the low SNR image.

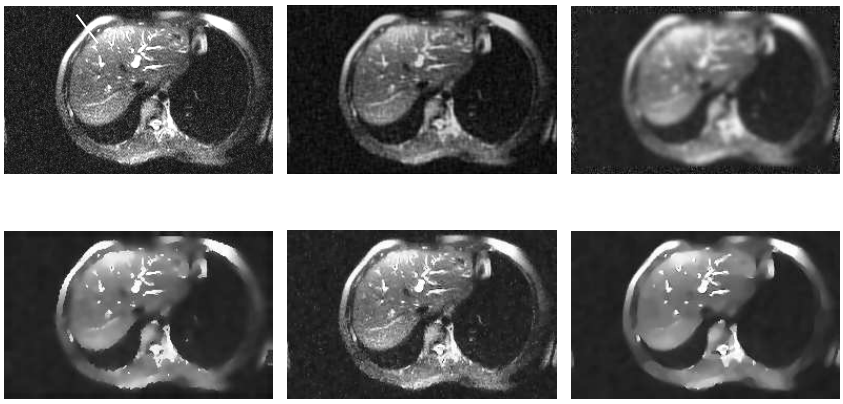


Figure 2: Results from each of the smoothing filters, top-row, l-to-r is the original image, image after Savitzty-Golay and Gaussian. Bottom-row, Adaptive, Nonlinear Diffusion and Anisotropic Gaussian.

The strength, shift and spread of the edge is evaluated on each of the images. Histogram plots across two edges are shown in figure 3 showing both the image pixels and the gradient across the edge. For the lab image the results are similar for all filters with more significant differences between filters in the medical image. Two measurements are taken from these histograms which indicate edge strength and spread. These results are compiled in Table 1. While Savitzky-Golay and Gaussian filters spread the edge, the other three maintain and even enhance the edge characteristics.

7 Conclusion

Five filters were evaluated using two criteria, texture smoothing and edge preservation. The filters consisted of two linear filters, two non-linear isotropic and one non-linear anisotropic. The filters were tested on two images with high and low SNR and the results show that, particularly in the low SNR case, the anisotropic and adaptive filters performs much better than the linear filters at smoothing out the noise in homogeneous areas while still maintaining the edge strengths with minimum blurring across the edge.

The Gaussian performs the worst of all the filters. The Savitzky-Golay deals better at preserving the edges but again suffers in the lower SNR image. The anisotropic and adaptive smoothings preservation of edges allows for more agressive smoothing on homogeneous areas.

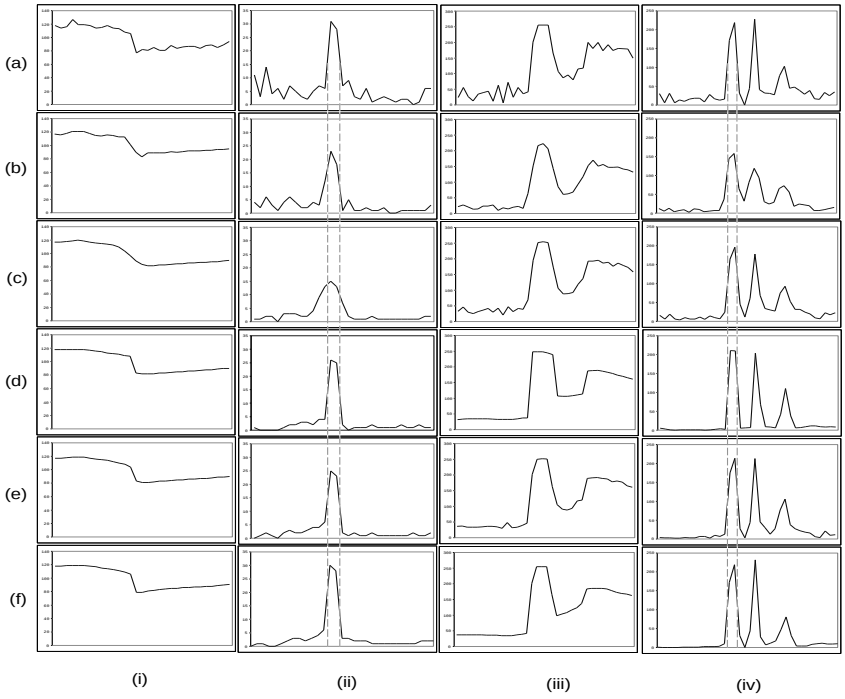


Figure 3: Pixel intensities and gradients along white lines from images *figure 1(a)* and *figure 2(a)*. *(i)* and *(iii)* show the pixel intensities and *(ii)* and *(iv)* show the gradient values from the lab image and the medical image respectively. *(a)* is the original image, *(b)* image after Savitzty-Golay, *(c)* Gaussian, *(d)* Adaptive, *(e)* Nonlinear Diffusion and *(f)* Anisotropic Gaussian.

References

[1] J.J. Koenderink The structures of images. *Biological Cybernetics*, 50:363–370, 1984.

[2] M. Petrou and P. Bosdogianni. *Image Processing: The Fundamentals*. Wiley Publishing, Inc., 1st edition, 1999.

[3] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(7):629–639, 1990.

[4] G. Gerig, O. Kubler, R. Kikinis, and F.A. Jolesz Nonlinear anisotropic filtering of MRI data. *IEEE Transactions on Medical Imaging*, 11(2):221–232, June 1992.

[5] K. Chen A Feature Preserving Adaptive Smoothing Method for Early Vision. Technical report, National Laboratory of Machine Perception and The Center for Information Science, Peking University, Beijing, China, 1999.

[6] G. I. Sanchez-Ortiz, D. Rueckert, and P. Burger Knowledge-based tensor anisotropic diffusion of cardiac magnetic resonance images. *Medical Image Analysis*, 3(1), 1999.

[7] J. Weickert A review of nonlinear diffusion filtering. *Scale-Space Theory in Computer Vision*, 1252:3–28, 1997. Springer, Berlin. B. ter Haar Romeny, L. Florack, J. Koenderink and M. Viergever (Eds.).

[8] A. Savitzky and M.J.E Golay Smoothing and differentiation of data by simplified least squares procedures. *Analytical Chemistry*, 36(8):1627–1639, 1964.

COPLANAR CAMERA CALIBRATION WITH SMALL DEPTH OF FIELD LENS

Barry McCullagh, Fergal Shevlin
Dept of Computer Science, Trinity College
Dublin 2, Ireland
email: Barry.Mccullagh@cs.tcd.ie

Abstract

This paper introduces a new technique for calibration of cameras which have lenses with small depth of field. Standard coplanar calibration algorithms fail if the coplanar target is parallel to the image plane. Elements of the rotation matrix which are used to calculate the focal length, F and the distance from the origin of the world coordinate system to the camera coordinate system, Tz become 0 and it is only possible to calcualte the quotient F/Tz. By combining a traditional calibration algorithm with a 'Depth from Defocus' algorithm, the distance to the world coordinate system can be calculated.

Keywords*: Coplanar camera calibration, Depth from defocus, Parallel calibration, Telecentric calibration*

1 Introduction

Camera calibration is the process of determining the internal and external parameters of the camera so that the location of the objects observed by the camera can be determined. This is necessary in many applications such as stereoscopic depth recovery. The external parameters are:

- $\mathbf{t}$, the translation vector containing the translations along the X, Y, and Z axes from the origin of the world coordinate system to the origin of the camera coordinate system
- ω, ϕ and κ, the angles of rotation of the world coordinate system axes relative to the camera coordinate system which are used to form an orthonormal 3x3 rotation matrix, $\mathbf{R}$.

The internal parameters include F, the focal length of the camera, P, the principle point of the optical system and other parameters which represent lens distortions such as barrel and radial distortion. This paper is concerned with calculating the external parameters and assumes that the internal parameters have been previously calculated.

Coplanar calibration is the process of finding the parameters using a series of world and image points, the world points lying on a two-dimensional plane in three dimensional space. For noncoplanar calibration, the collinearity condtion equations provide a set of six constraints which can be used to calculate the parameters. If the lens has a small depth of field, rotating the target relative to the image plane will result in a blurred image. Extracting accurate geometry from a blurred image is not possible so the target and image plane must be parallel which requires a setup like that shown in figure (1).

Parallel calibration is a problem which is encountered in many inspection tasks such as automated optical inspection (AOI). AOI covers many inspection tasks such as the inspection of PCBs. In PCB inspection, the height of the chips on the board is negligible relative to the size of the chips. Telecentric lenses are used in many machine vision inspection systems. Telecentric lenses virtually remove perspective error because all of the rays intersect the image plane at 90^0. This does not increase the depth of

field, it just ensures constant magnification across the depth of field. This simplifies the inspection task but makes the calibration harder.

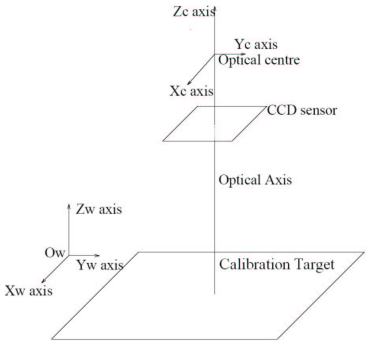


Figure 1: Camera and target allignment

The parallel constraint of the target and image plane complicates the calibration process. For coplanar calibration it is recommended that the target be tilted by at least 30^0 relative to the image plane, [9]. If the target is rotated by a small amount, then the bottom row of the rotation matrix will become 0 (or very close to 0). The rotation matrix is created by multiplying the three matrices corresponding to rotation about the X, Y and Z axes ($\mathbf{Rx}$, $\mathbf{Ry}$ and $\mathbf{Rz}$ respectively). If the rotation about the X and Y axes are almost 0, $\mathbf{Rx}$ and $\mathbf{Ry}$ will be very close to the identity matrix. Irrespective of the rotation about the Z axis, κ, the values on the bottom of the rotation matrix, $\mathbf{R}$, will consist of 0, 0 and 1 if there is no rotation relative to the X and Y axes.

$$R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} cos(\kappa) & sin(\kappa) & 0 \\ -sin(\kappa) & cos(\kappa) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

Because these elements in the rotation matrix are 0 the number of constraints which can be used to calculate the parameters is reduced to 3 [1]. Therefore it is not possible to calculate a unique value of *F* and *Tz*, only the quotient *F/Tz* [1]. In their comparison of coplanar calibrarion algorithms, Chatterjee and Roychowdhury, [1], ignore all cases when the rotation relative to the $\mathbf{x}$ and $\mathbf{y}$ axes is $\mathbf{0}$ because *F* and *Tz* cannot be solved individually.

There are many algorithms which can accuratly calibrate cameras using noncoplanar techniques. Several of these such as Ganapathy [3] and Grosky et al [4] can be extended to the solve the coplanar case. These algorithms are extended by transforming the collinearity equations so the parameters can be estimated using linear equations [1]. Neither pure coplanar calibration algorithms or these algorithms when extended to the coplanar case can calibrate the camera when the target is parallel to the image plane. The methods above along with Tsai, [7], and others use elements of the rotation matrix as a denominator to calculate F and Tz. If the target is coplanar with no tilt relative to the image plane, the denominator of the equations used to solve for F and Tz reduces to Tz, eq(3)

Not only does the calculation of *F* and Tz cause problems, but sometimes these algorithms do not calculate enough parameters. Many lenses have a variable distance from the image plane to the optical centre of the lens. If this value is not calculated the accuracy of the calibration technique cannot be calculated. To check the accuracy of the algorithm, points are transformed through the rotation and translation matrices. These image points should map to the scene points. If the image plane distance is assumed to be the same length as the focal length then this will not be the case.

In this paper we introduce a new algorithm to calculate Tz which will also calculate the distance from the centre of projection to the image plane. In [6], Pentland describes a technique which uses the amount of blur in an image to calculate the depth. He suggests that the distance to an object can be calcualted if the other parameters of the camera such as focal length and aperture are known. Using this equation in an optimisation search, both the distance to the object and the distance to the image plane can be calculated. By combining this method of calculating *Tz* and image distance with an established method for calculating translations parallel to the image plane and the rotation matrix we present an accurate method for calibrating cameras with small depths of field.

2 Tsai's calibration algorithm

Tsai's algorithm, which is one of the most widely used camera calibration algorithms, is a very fast and accurate algorithm. Unfortunatly it cannot calibrate a camera if the image plane and image sensor are

parallel. The algorithm begins by calculating initial estimates of as many internal and external parameters as possible using linear least squares fitting methods. Using nonlinear optimisation methods any unsolved parameters are then solved, and the initial parameters improved. The initial parameters calculated include both internal and external parameters of the camera and this is accomplished in two steps. The first is to calculate the rotation matrix, R and two of the translation parameters, Tx and Ty. When these have been calculated, F and Tz are then solved. However, when calibrating a system where the target is parallel to the image plane the algorithm fails during this step.

Horn, [5], shows that the correspondence between the individual points can be expressed as:

$$\frac{x_i}{F} = \frac{r_{11}x_S + r_{12}y_S + r_{13}z_S + T_x}{r_{31}x_S + r_{32}y_S + r_{33}z_S + T_z} \quad \frac{y_i}{F} = \frac{r_{21}x_S + r_{22}y_S + r_{23}z_S + T_y}{r_{31}x_S + r_{32}y_S + r_{33}z_S + T_z} \tag{2}$$

Because there is no rotation of the target away from the image plane, the values of r_{31}, r_{32}, r_{13} and r_{23} calculated in the first step are zero. The value of z_S, the height of the target relative to the Zw plane is also 0 because the target is coplanar. Due to the large number of zero valued terms in equations (2) the right hand side of both equations become 0, and so the equations become linearly dependent and *F* and *Tz* cannot be solved uniquely. Rearranging equations (2)

$$\frac{F}{T_z} = \frac{x_i}{r_{11}x_S + r_{12}y_S + T_x} \quad \frac{F}{T_z} = \frac{y_i}{r_{21}x_S + r_{22}y_S + T_y} \tag{3}$$

Because F and Tz are not expressed independently of each other they cannot be individually solved so another method is required to calculate these parameters.

3 Depth from defocus

In 1987, Pentland [6] described a new method for calculating the depth in an image. He suggests that the distance to an object, D, (which [7] calls Tz), is related to the focal length, *F*, image distance v_0, the f-number of the system, *f*, and the spatial constant of the point spread function (radius of the blur circle), σ.

$$D = \frac{F * v_0}{v_0 - F - \sigma * f} \tag{4}$$

When a point object is observed by a camera focused at a different depth, the image is a blurred circle. The point spread function (PSF) is the function which transforms the point object to the blurred circle observed by the camera, and the blur circle is the radius of the circle observed by the camera. The PSF for cameras is modelled by Pentland as a two-dimensional Gaussian function, G(r, σ), where r is the radial distance of the function. An unfocused image is therefore the convolution of a 2D Gaussian function (with appropriate r and σ) with the focused image.

In his equation, Pentland assumes that all the variables except D and σ are known. Calculation of D is trivial after calculating σ. Unfortunatly, the measurement of σ is not trivial as it depends on both the characteristics of the scene and the characteristics of the camera. In order to separate the two, Pentland suggests using areas in the scene where characteristics are known. Such areas are places where edges and sharp discontinuities can be observed.

3.1 Calculating σ

The relationship between a focused image, unfocused image and the PSF (defocus operator) is expressed by Ens and Lawrence in equation (19) of [2]:

$$I2 = I1[\otimes]D \tag{5}$$

$[\otimes]$ represents bounded convolution, where the convolution kernel does not pass outside of the boundary of the image.

Using this relationship we can calculate the defocus operator which relates the focused and unfocused images. By convolving the focused image with Gaussian functions of increasing values of σ, a value of σ will be reached which results in a very close approximation of the unfocused image being produced. In order to reduce the amount of time taken to find the value of σ, convolution can be done in one dimensional space rather than two dimensional space. By extracting rows with sharp discontinuities from the focused image, convolving them with one dimensional Gaussian functions of varying σ and comparing them to the unfocused image, it is possible to reach the correct value of σ much quicker.

The unfocused image is at a known distance, δ, from the position of best focus. In order to increase the accuracy of the the calculation, values of σ for several known distances from focus are calculated, so D and σ are replaced with $D + \delta_i$ and σ_i.

$$D + \delta_i = \frac{F * v_0}{v_0 - F - \sigma_i * f} \tag{6}$$

3.2 Calculating D and v_0

The values of σ calculated in the previous section are at known distances from focus, not at the point of focus. To calculate the values of D and v_0 it is necessary to change the equation slightly. If all the parameters in equation (4) are exact, then rearranging (4) should give a result of 0:

$$0 = \frac{F * v_0}{v_0 - F - \sigma * f} - D \tag{7}$$

Although values of σ have been calculated, none of these values are the value at focus, D. To calculate the value of D and v_0, the equation can be used in an optimisation search to iterate through values for these variables. When these values are correct, the error from equation (8) will be at its lowest.

$$\text{Error} = \sum_{1 \le i \le n} \frac{F * v_0}{v_0 - F - \sigma_i * f} - (D + \delta_i) \tag{8}$$

3.3 Combining with Tsai

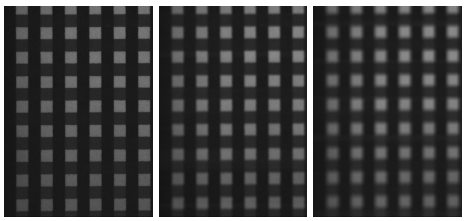
Thus far we have calculated the values for D and v_0. However, to accuratly calibrate a camera several other external parameters are required, these are Tx, Ty and the rotation matrix, $\mathbf{R}$. To calculate these parameters it will be necessary to combine our algorithm with an existing algorithm. Tsai's algorithm, [7], was the algorithm that was initially chosen to calibrate the camera. As shown earlier, it was not possible to use this algorithm to completly calibrate the camera. However the algorithm calculates the focal length and object distance after calculating the rotation matrix and the translations parallel to the X and Y axis. Because of the order in which the parameters are calcualted, the values of $\mathbf{R}$, Tx and Ty calculated are the values which relate the camera and world coordinate system axes.

4 Evaluation

The above algorithm is tested with two sets of real data obtained from our vision system. The accuracy is checked in two ways. If the camera is calibrated correctly, a ray projected from the image plane through the center of the optical system into the world coordinate system should intersect with the corresponding point on the target. The first check was to calculate the closest distance from each point on the target to the corresponding ray projected from the image plane. The second was to calculate the Euclidian

distance between the point on the target and the (x, y) value of the ray where it intersects the plane of the calibration target.

The target was a series of white squares on a black background. The squares, each of side 50 dots, were created using a PostScript file and printed on $100g/m^2$ paper using a PostScript printer of resolution 600dpi. Figure (2) shows the target along with sample images taken at various distances from defocus.



(a) Target at focus (b) 4mm from focus (c) 7mm from focus

Figure 2: Calibration target, and target at various distances from focus

The optical system is a PULNiX TM1001 CCD camera with a Computar 55mm Telecentric lens attached.

The first step is to calculate the sigmas for the images. Rows from the focused image were convolved with Gaussian functions of increasing sigma and the result compared to the corresponding row of the unfocused image. The values of σ were increased by .0001 each iteration. The value of the radial distance was chosen so there were no non 0 values truncated from the function, even at the largest used σ. Figure (3) show the graphs of σ against distance from focus. As these graphs show, the values of σ increase almost linearly as the distance from focus increases. Figure (4) shows the derivative of σ with respect to the distance from focus.

In each case, the values of σ and the corresponding distances from focus are used in equation (8).

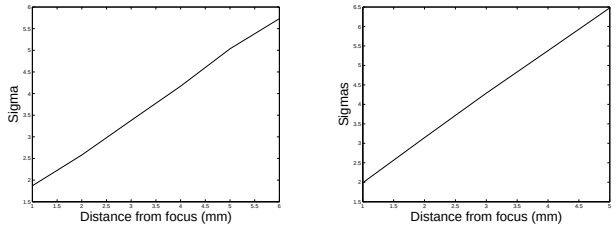


Figure 3: Sigmas vs Distance from focus

Finding the exact centre of the optical system is difficult when a compound lens is used. The values of D and v_0 returned from the minimisation search were measured against the distance from the focused position to the location of sensor and found to be within 1mm.

The values of $\mathbf{R}$, Tx and Ty were obtained in each case from Reg Wilson's implementation of the Tsai algorithm [8]. These values were as accurate as could be measured. The number of squares visible on the target was known so the translation from camera to world coordinate system could be measured very accurately.

The table below shows the errors which were calculated from projecting the rays from the image plane into the real world. Method 1 refers to the distance from each point to the ray projected from the target, while method 2 refers to the distance from the point to the intersection of the corresponding ray and the target plane.

Set (method)	1 (1)	1 (2)	2 (1)	2 (2)
Min error (mm)	0.0203	0.1899	0.0077	0.0109
Max error (mm)	0.4965	0.5038	0.4513	0.4642
Std	0.1195	0.1272	0.1162	0.1136

There are several reasons why the accuracy of this method is not as good as it could be. The centre of the optical system may not be the centre of the lens unit as was presumed when performing the experiments because it consists of many internal lenses. The position chosen as the position of focus may not be the position of best focus. When selecting the position of best focus, a continuous feed from the camera is observed. The position selected as the position of best focus is

subjective and could be incorrect. The internal parameters of the lens dealing with distortion and the center of the CCD were not calculated, so this will also reduce the accuracy of the experiment.

5 Conclusions

This task was undertaken because none of the existing camera calibration algorithms could calibrate a camera with a small depth of field lens attached. The problems that existing calibration techniques encountered was due to a lack of perspective. All that is known is a ratio of the image and object sizes. Because the section of this algorithm which calcualtes the distance to the object relies on the blur in the object it does not encounter the same problems and therefore it can be used to calibrate a camera with small depth of field lens attached.

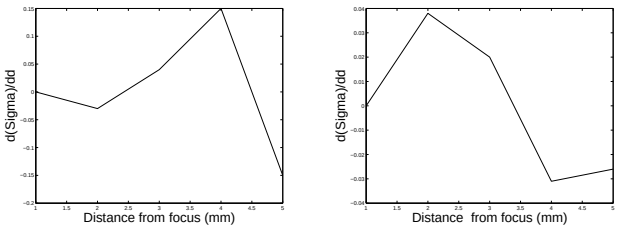


Figure 4: Derivative of sigma vs distance graphs

References

[1] C Chatterjee and V.P. Roychowdhury. Algorithms for coplanar camera calibration. *Machine Vision and Applications*, 12(2):84–97, 2000.

[2] J Ens and P Lawrence. An investigation of methods for determining depth from focus. *IEEE Transactions on Pattern Analysis amd Machine Intelligence*, 15(2):97–108, Feb 1993.

[3] S Ganapathy. Decomposition of transformation matrices for robot vision. *International Conf. on Robotics and Automation*, 1(1):130–139, March 1984.

[4] W.I. Grosky and L.A. Tamburino. A unified approach to the linear camera calibration problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):633–671, July 1990.

[5] B.K.P Horn. Tsai's camera calibration method revisited. *www.ai.mit.edu/people/bkph/papers/tsaiexplain.pdf*.

[6] A.P. Pentland. A new sense for depth of field. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9(4):523–531, July 1987.

[7] R.Y. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, RA-3(5):323–344, August 1987.

[8] R Wilson. Tsai source code. *http://www-2.cs.cmu.edu/ rgw/TsaiCode.html*.

[9] R Wilson. Tsai faq. *http://www.ien.it/is/vislib/README.Tsai_FAQ*, 1(1):1, May 1985.

INTEGRATION OF FEATURE DISTRIBUTIONS FOR COLOUR TEXTURE SEGMENTATION AND ITS APPLICATIONS

Padmapriya Nammalwar *, Ovidiu Ghita, Paul F. Whelan
Vision Systems Group
School of Electronic Engineering
Dublin City University, Dublin, Ireland
email: nammalp2@mail.dcu.ie, {ghitao,paul.whelan}@eeng.dcu.ie

Abstract

This paper proposes a framework for colour texture segmentation. The framework uses the colour and the texture distributions for discriminating the colour textured regions. The proposed colour texture segmentation method was tested in three different applications. The applications includes Irish Script on Screen (ISOS) images, skin cancer images and Sediment Profile Imagery (SPI). Image textures are used in combination with colour features for the segmentation of the colour textured regions in the document, to identify the diseased area in the skin cancer images and to segment underwater images. The inclusion of colour and texture as distributions of regions provide a good discrimination of the colour and the texture. The experimental results proved that the framework is effective and efficient.
Keywords*: Colour, Texture, ISOS Images, Skin Cancer Images, SPI Images*

1 Introduction

Colour and texture are important features in image segmentation. This paper developed a novel framework which considers the distributions of colour and the distributions of texture to discriminate the colour textured regions. Researchers have developed different frameworks for colour texture segmentation and most of them are designed for a specific application. Jolly *et al.*[5] proposed an algorithm for colour texture segmentation that was applied to update old cartographic aerial maps. Song *et al.* [9] proposed a method that uses colour and texture to detect defects in random colour textured images and in particular, granite images. Kyllonen *et al.* [6] described a wood surface inspection method that combines colour percentile features with texture features based on simple spatial operators.
An application independent colour texture method is proposed in this paper. The framework covers applications from different fields. Three different applications from three different areas were selected for testing the developed framework. The applications discussed in this paper are ISOS images for the segmentation of the color textured regions in the document, segmentation of skin cancer images to identify the diseased area and SPI images to segment the underwater images. The images evaluated in this paper are taken from different environments and the segmentation is found to be efficient.

*Corresponding author

2 Framework for Colour Texture Segmentation

2.1 Steps in Colour Texture Segmentation

A novel framework was developed for combining texture and colour information for colour texture segmentation that makes the segmentation robust and efficient for different types of images. The steps are as follows,

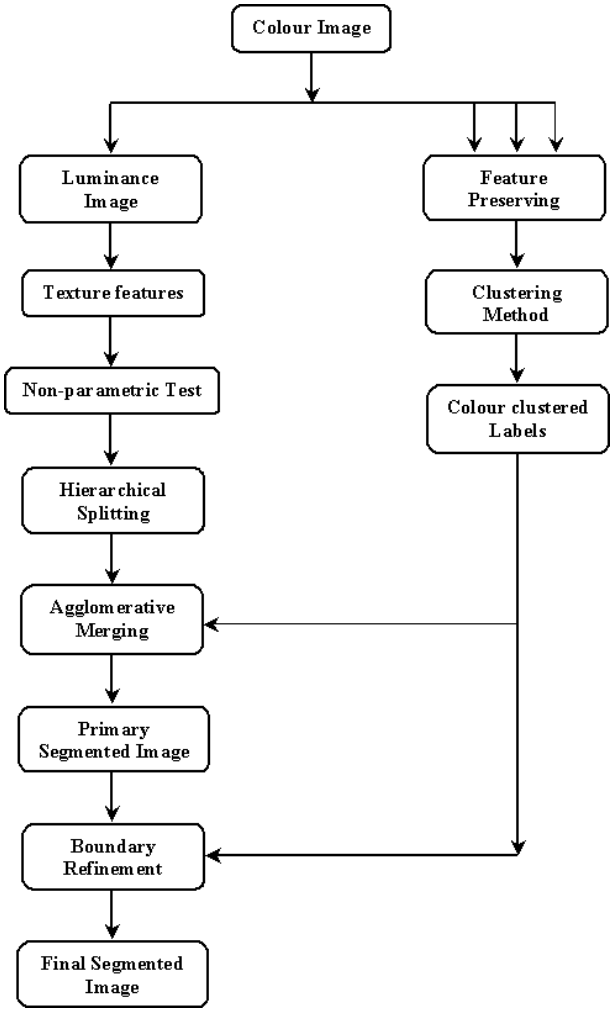


Figure 1: Framework for colour texture segmentation

- Initially the features were identified by means of feature extraction techniques, in which image information is reduced to a small set of descriptive features. The Local Binary Pattern (LBP) and the contrast features are extracted from the luminance plane.
- The distribution of the texture features are used for texture discrimination.
- A Modified-Kolmogorov Smirnov (M-KS) non-parametric statistical test is used as a similarity measure to discriminate the texture distributions.
- A hierarchical splitting method is used to split the image based on the texture descriptors using the similarity measure.
- An adaptive smoothing is performed to preserve the features and to obtain a good segmentation along the boundaries. This technique removes noise and prevents over segmentation.

- An unsupervised *k*-means clustering algorithm is performed on the image to classify the patterns into their respective classes and to obtain the distribution of the colour clustered labels.
- Distribution of the texture features and the distribution of the colour clustered labels are used to describe the texture and the colour respectively. The distributions of colour and the textures was used to derive the merger importance (MI) value between two adjacent regions. The MI value was calculated using the M-KS statistic. Two weights are computed to set the statistical relevance for texture and colour distributions.
- An agglomerative merging procedure based on the merging criteria determines the similarity between two different regions using M-KS statistic, producing the segmented image.
- The final step is to refine the boundaries of the image. A boundary refinement algorithm enhances the segmented result to obtain the final segmented image.

2.2 Details of Colour Texture Segmentation

2.2.1 Feature Distributions

Local Binary Pattern (LBP) approach provides robust texture related information and knowledge about the spatial structure of the local image texture. LBP is combined with the contrast of the texture which is a measure of local variations present in an image for the texture description. The distributions of LBP and contrast $(256 * 8)$ were used for texture description.
The proposed method uses the unsupervised clustering technique based on the *k*-means algorithm to cluster the colour features. The *k*-means algorithm [4] is the simplest and most popular technique among the iterative clustering algorithms. The k-means algorithm organises the objects into an efficient representation that characterises the population being sampled. The number of clusters is generally image dependent so the initial value is set to 10 clusters, this number is sufficient to capture all the relevant clusters. The distribution of the colour clusters is used for colour description.

2.2.2 Modified Kolmogorov Smirnov (M-KS)

A non-parametric test M-KS statistic was used for comparing LBP/C with colour clustered labels. This tests the hypothesis that two empirical feature distributions have been generated from the same population. M-KS has the desirable property that it is invariant to arbitrary monotonic feature transformations [8]. The M-KS statistic is defined as the sum of the absolute value of the discrepancies between the normalised cumulative distributions,

$$D(s,m) = \sum_i \left| \frac{F_s(i)}{n_s} - \frac{F_m(i)}{n_m} \right| \tag{1}$$

where $F_s(i)$ and $F_m(i)$ represent the sample cumulative distribution functions; n_s and n_m represent the number of pixels in the sample and model regions respectively. Since M-KS is normalised, it is advantageous over other statistical measures.

2.2.3 Segmentation Method

The unsupervised colour texture segmentation method involves three steps: hierarchical splitting, agglomerative merging and the boundary refinement.

Hierarchical Splitting The hierarchical splitting procedure recursively splits the input image into four subblocks. The six pairwise M-KS values between the LBP/C of the 4 subblocks are calculated. The uniformity of the region is tested by a decision factor

$$R = \frac{MKS_{max}}{MKS_{min}} > X \tag{2}$$

where X is a threshold value, MKS_{max} and MKS_{min} represents the highest and the lowest M-KS values.

Agglomerative Merging An agglomerative merging procedure was applied on the image which has been split into blocks of roughly uniform textures. This procedure merges similar adjacent regions until a stopping rule is satisfied. The pair of adjacent segments which has the smallest Merger Importance (MI) value were merged. The MI value between two regions is calculated as followed,

$$MI = w_1 * MKS_1 + w_2 * MKS_2 \tag{3}$$

where w_1 and w_2 represent the weights for the LBP histogram and the colour clustered histogram respectively. MKS_1 and MKS_2 represents the M-KS statistic for texture and colour histograms respectively. The weights are automatically detected using an uniformity factor defined as the maximum of the ratio between colour clustered histogram and number of pixels in the two regions under consideration, the sample and the model regions.

$$k_j = \max\left\{\frac{CL_j[i]}{N_p}\right\} \tag{4}$$

where k_j represents the uniformity factor for the two sample regions. If the difference between k_1 and k_2 is less than 0.1, i.e., both the sample and the model weights are more or less the same, then $w_2 = (k_1 + k_2)/2$ and $w_1 = 1 - w_2$. This indicates that colour influences more than texture, hence colour statistic is given more importance. On the other hand, if the difference between k_1 and k_2 is high, both the texture and the colour are given equal weights. Details about the colour and texture weights can be found in [7]. The developed method follow a simple stopping rule, $MinMI > Y$, where MinMI represents the minimum merger importance value. If this is greater than a threshold value then the merging procedure is halted.

Boundary Refinement The agglomerative merging procedure resulted in blocky segmented image. A new boundary refinement algorithm was developed and used for the improvement at the boundaries between various regions. A pixel is regarded as a boundary point if its region label is different from at least one of its four neighbours. For an examined point *P*, a discrete square with a dimension *d* around the pixel was placed and the colour histogram for this region was computed. The corresponding colour histograms for the different neighbouring points were calculated. The homogeneity of the square region and the *i*th neighbouring region, *i*=1,2,...*l*...*n* region was computed. The pixel is reclassified if the MI value between adjacent regions and the region around the pixel under consideration is lower than the merge threshold. This procedure is iterative and proceeds until no pixels are relabelled. Reassigning pixels in this way significantly improves the accuracy of the segmentation process.

3 Applications and Evaluation

The developed framework can be applied to a number of different applications. Three out of a number of possible applications were selected and presented in this paper. The applications for colour texture segmentation include the segmentation of colour textured regions in Irish manuscript document, disease detection, and the segmentation of underwater images. Application database consists of the Irish Script On Screen (ISOS) images, skin cancer images and the Sediment Profile Imagery (SPI). The following sections presents the segmented results and the salient features of the segmentation.

3.1 Irish Script on Screen Images

The ISOS images are digital images of Irish manuscripts. The sample images were taken from old Irish manuscripts online - Irish Script On Screen (See: http://www.isos.dcu.ie/). About 5,000 early Irish language manuscripts survive and those appearing on this website form an important and distinctive part of Irish heritage. This includes the Book of Leinster, which is compiled in the second half of the 12th century. The storage of letters, papers or other documents in ordinary stationery-grade folders or plastic sleeves invites certain deterioration even when kept in sealed containers. Slowly but inexorably, paper degrades and discolors, and ink fades. High heat and humidity are also detrimental. Most papers contain acid which over time will cause the paper to weaken and become brittle. A historic document which would otherwise appreciate greatly as a prime investment is debased in value. In addition, the corrosive effects of modern environment and time is a threat to document preservation. The objective of ISOS is

(1) (2) (3) (4)

Figure 2: (1 - 4) represents segmented results of ISOS images after the boundary refinement stage

to create digital images of Irish manuscripts, and to make these images available together with relevant commentary, accessible on a website. The ISOS images are available in joint photographic experts group (jpeg) format. The ability to segment a document into functionally different parts has been an ongoing goal of segmentation of the document analysis research. Segmentation of the old document images helps to determine the amount of damage in the document, caused by the afore-mentioned factors. This encompasses decomposing a document into its various corrupted components. This provides information on the amount of care to be taken to preserve the document from further damage. Four ISOS images were considered as the application database.

Figure 2-(1) illustrates the correct identification of the stained regions on the script images. A small portion of the green region is identified. Figure 2-(2) shows the tarnished region. Figure 2-(3) demonstrates the segmentation of the soiled region and the unsoiled region. In addition, one large coloured script was identified precisely. Figure 2-(4) categorises the discoloured and the blemished regions. Colour plays a vital role in the developed framework which is evident from the presented results. Though the small scripts were not segmented separately, the damaged region and the different colours in the script were identified properly. The quantification of the segmentation of the ISOS images was based on the ground truth images. A boundary was drawn around the stained regions in the image and these regions were considered as the ground truth for quantification. The average segmentation error for four script images was found to be 2.6 percentage.

3.2 Skin Cancer Images

Skin cancer is the most prevalent form of human cancer that is generally caused by over exposure to sun. There are different types of skin cancer and some are likely to be fatal. Skin cancers can be classified into melanoma and non-melanoma. Melanoma is the most dangerous form of skin cancer. It can spread through the whole body and is usually fatal if it does. If detected early, the cure rate for melanoma is almost 100 percent. Late detection, when the melanoma is more than three millimeters deep, results in only a 59 percent survival rate. Melanoma's are much less common than non-melanoma's, but they

account for most of the mortality from skin cancers. Detection of malignant melanoma in its early stages considerably reduces morbidity and mortality [10]. People are considered more at risk if they have many moles, are fair skinned with blue eyes, tend to sunburn easily or have freckles [3]. The rate of melanoma cases worldwide is increasing faster than any other cancer, with an annualised rate of increase of six percent. Since 1973, the mortality for melanoma has increased by 50 percent.
Clinical features of pigmented lesions suggestive of skin cancer are known as the ABCD's of the skin cancer: asymmetry, border irregularity, colour variation, diameter greater than 6mm. There are various image analysis techniques developed to measure these features. Measurement of image features for diagnosis of the skin cancer images requires the detection of the lesions and their localisation in an image. It is essential to determine the lesion boundaries accurately so that the measurements such as maximum diameter, irregularity of the boundary, and colour characteristics can be accurately computed. As a first

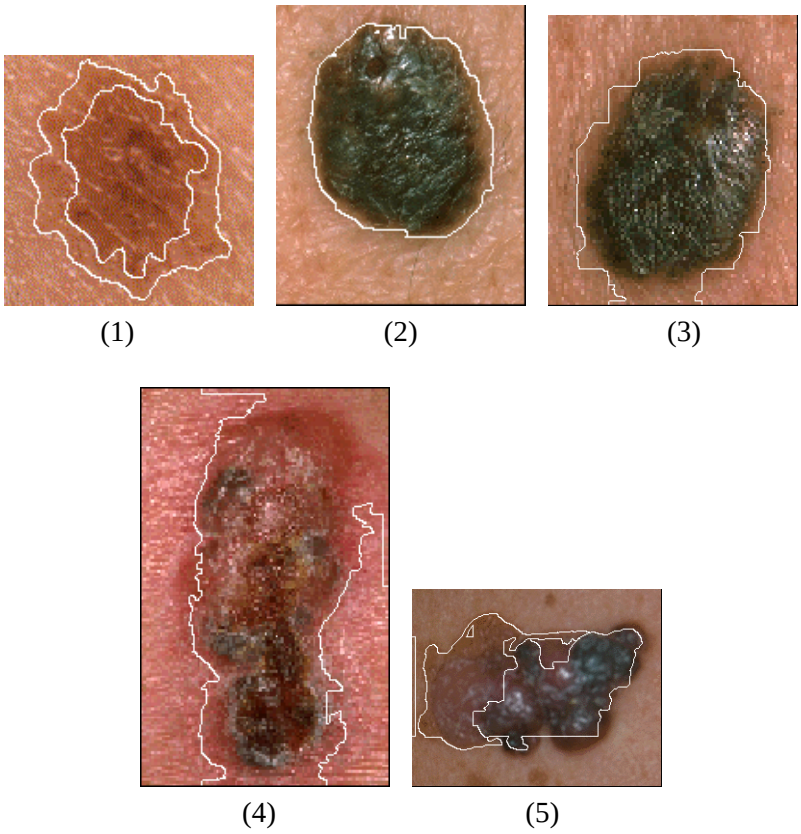


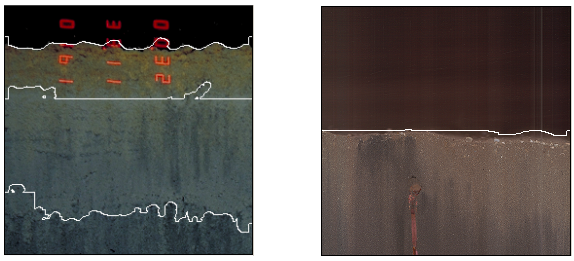
Figure 3: (1 - 5) represents segmented results of skin cancer images after the boundary refinement stage

step in skin cancer identification, the lesion boundaries are delineated by various image segmentation techniques. In this research work, colour and texture information from an image is used for the segmentation of the lesion boundaries. The segmentation helps to diagnose the skin lesions in the early stages. The skin cancer images obtained were in graphics interchange format (gif). Five skin cancer images considered for the application database were taken from the [2].
The skin lesions have complex structure, colour as well as large variations in size. Generally, the lesions have a high contrast with respect to healthy skin areas. The borders of lesions are not always well defined which makes the segmentation more complex. To analyse skin lesions, it is necessary to accurately locate and isolate the lesions. The efficient performance of the proposed colour texture segmentation method exactly recognised the boundaries in the skin lesions as shown in Figure 3.
Colour is one of the significant feature in the examination of a skin lesion. Typical examples of lesions show reddish, bluish, grey and black areas and spots. Figure 3-(1), shows the segmentation of the skin lesion. The fine variation in the colour is identified and segmented accurately. Figure 3-(2), Figure 3-(3),

Figure 3-(4) and Figure 3-(5), illustrates the segmented results of the skin lesion. This segmentation clearly identifies the difference in colours in the skin lesion. The distribution of texture and colour features presents significant information, hence the segmentation based on the two features seems to be appropriate. This allows for the isolation of the lesion from healthy skin and extracts homogeneous coloured regions separately. The quantification of the skin lesion segmentation was based on visual results. The experimental results obtained proved to be encouraging and indicate that this method of colour texture segmentation is appropriate to be applied for detection of skin cancer images. Further evaluations on the segmentation can only be performed by an experienced dermatologist.

3.3 Sediment Profile Imagery

Sediment Profile Imagery (SPI) is a remote sensing technique that is used to determine whether the marine sediments provide suitable habitat for bottom dwelling fauna. This is an innovative and cost efficient method of surveying and monitoring lake or marine aquatic environments. The traditional method of sample collection and subsequent laboratory analysis is time consuming and expensive and data return time is slow. SPI is based on single lens reflex (SLR) camera photography and computer-based image analysis which greatly accelerates the time required to write reports and provide relevant data. The physical, chemical and biological features associated with organic enrichment of the underwater sediment are imaged and measured with the SPI system. The segmentation of the SPI images is the preliminary step in most pictorial pattern recognition and scene analysis problems. These images are hard to process due to



(1) (2)

Figure 4: (1) and (2) represents segmented results of SPI images after the boundary refinement stage

the light absorption, changing image radiance and lack of well defined features. The underwater images shows fluctuating oxygenation levels under different organic loading and hydrographic conditions.
Figure 4-(1) shows an example of a sediment image under high organic loading stress in hypoxic conditions. This is an example of a heavily impacted sediment. There is a clear difference in colour between the sediment surface layer and that lying under it. The colour texture segmentation clearly identifies different layers. Figure 4-(2) represents the sediment image with burrowing marine worms. The opportunistic worms thrive in high organic loading conditions and their burrowing action can often reintroduce oxygen into depleted sediments. Due to the thin feature difference in the organic sediment and the worm, the colour texture distribution could not identify the worm separately. But the sediments were segmented accurately. The results were compared with the results obtained by Ghita *et al.* [1] and found to be similar. The segmented result indicates that the developed framework for colour texture segmentation is able to identify the different sediment layers in the image.

4 Conclusions

The goal of this paper is to find the performance of the developed colour texture segmentation method in the script images, skin cancer images and underwater images. The proposed algorithm, was tested in three different applications, ISOS script images, skin cancer images and the underwater images. The

algorithm was applied on the images and was found to produce proper segmentation results. All these applications use different images taken from different cameras and varying environment. In spite of these differences, the proposed colour texture segmentation method is able to identify different colour textured regions in the image and the results of the segmentation process are appropriate and visually acceptable. In all the three applications, colour is a determinant factor in the segmentation process.

References

[1] O. Ghita, P.F. Whelan, and R. Kennedy. A practical approach for analysing spi images. In *Proceedings of Systemics, Cybernetics and Informatics*, 2003.

[2] Skin Cancer Images. http://dermatlas.med.jhmi.edu/derm/indexdisplay.cfm?imageid=1061935956, http://matrix.ucdavis.edu/tumors/tradition/gallery-melanoma.html, http://tray.dermatology.uiowa.edu/dermdb.htm.

[3] Skin Cancer Images. http://www.cmis.csiro.au/iap/recentprojects/melanoma.htm.

[4] A.K. Jain and R.C. Dubes. *Algorithms for clustering data*. Prentice Hall, Advanced Reference Series, New Jersey, 1998.

[5] M.D. Jolly and A. Gupta. Colour and texture fusion: Application to aerial image segmentation and gis updating. *IEEE workshop on applications of computer vision*, pages 2–7, 1996.

[6] J. Kyllonen and M. Pietikainen. Visual inspection of parquet slabs by combining colour and texture. In *IAPR workshop on machine vision applications*, 2000.

[7] P. Nammalwar, O. Ghita, and P.F. Whelan. Integration of feature distributions for colour texture segmentation. In *Proceedings of International Conference for Pattern Recognition*, 2004.

[8] J. Puzicha, Y. Rubner, C. Tomasi, and J.M. Buhmann. Empirical evaluation of dissimilarity measures for color and texture. In *ICCV (2)*, pages 1165–1172, 1999.

[9] K.Y. Song, J. Kittler, and M. Petrou. Defect detection in random colour textures. *Image and Vision Computing*, (14):667–683, 1996.

[10] L. Xu, M. Jackowski, A. Ghoshtasby, D. Roseman, S. Bines, C. Yu, A. Dhawan, and A. Huntley. Segmentation of skin cancer images. *Image and Vision Computing*, (17):65–74, 1999.

SEGMENTATION TECHNIQUES OF THE IMAGES OF SINGLE CELL ELECTROPHORESIS

Bogdan Smolka*
Silesian University of Technology
Department of Automatic Control,
Akademicka 16 Str, 44-100 Gliwice, Poland,
email: bsmolka@ia.polsl.gliwice.pl

Abstract

The single cell gel electrophoresis, called *comet assay*, is a microelectrophoretic technique of direct visualization of the DNA damage at the cell level. In the comet assay, the cells suspended in an agarose gel on a microscope slide are subjected to lysis, unwinding of DNA and electrophoresis. After staining with fluorescent DNA binding dye, cells with DNA damage display increased migration of genetic material from the cell nucleus. Under the influence of weak, static electric field, charged DNA migrates away from the nucleus, forming the so called comet. The damage is quantified by measuring the amount of the genetic material, that migrates from the nucleus to form the comet tail. The foremost advantage of the comet assay is that it analyzes individual cells, thus allowing the measurement of the heterogeneity of response within a cell population. In this paper we present three novel methods of the comet tail and head extraction, that allow to quantify the cell's damage.
Keywords*: image enhancement, comet assay, biomedical image processing*

1 Introduction

The comet assay, (single cell gel electrophoresis, SCGE or microgel electrophoresis, MGE) is a useful method for quantifying the cellular DNA damage caused by different genotoxic agents. The idea of single cell electrophoresis as a method of measurement of the DNA damage was introduced by Rydberg and Johanson, [1] and the comet assay was introduced by Östling and Johanson, [2].

The assay was named for the characteristic shape of the DNA, flowing from the nucleus and migrating under the influence of applied static electric field, (see Figs. 1, 2). The measurement of the DNA in the comet's tail enables to quantify the intensity of the DNA damage caused by various genotoxic agents. The information about the comets' tail and head boundaries, enables different calculations of the distribution of the DNA, that escaped from the cell nucleus.

In the recent years the use of the comet assay has grown considerably, as this method detects damages with high sensitivity and it is relatively fast and reliable. As a result, this method of detection of the DNA strand breaks on the individual cell level is now in wide use in genetic toxicology and oncology.

One of the application of the assay is the analysis of the effects of the ionizing radiation, [3–5] on the DNA structure. The formation of a comet may be a result of the DNA single strand breaks (SSB), double strand breaks (DSB) and alkali labile sites. Using different assay pH conditions, allows the study of either SSB or DSB. This ability of analyzing these two kinds of DNA damage is an important advantage of the comet assay.

For the experiments, the peripheral blood lymphocytes were taken from patients before the beginning of radiotherapy. Comet assay was performed according to Singh, [3] with some modifications described

*Supported by the KBN GRANT 4T11F01824

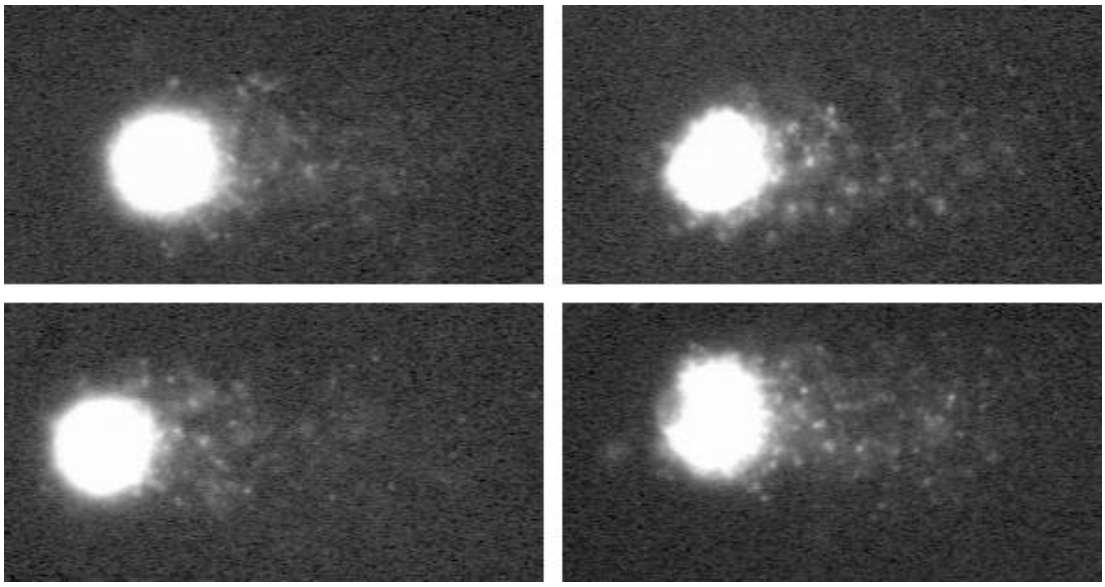


Figure 1: Typical comet assay images.

in [6]. The comets were observed using a fluorescence microscope at a 400-fold magnification and the images were acquired using a 512x512, 256 gray-levels frame grabber and stored on a computer disk. From the laboratory database, a collection of 30 digital comet images was prepared and all experiments were performed on these picture set.

2 New Methods of Comet's Tail Extraction

The currently applied techniques of the evaluation of the cell's damage are mostly based on simple thresholding, which has some severe disadvantages, as the global binarization is sensitive to noise and changes of the background illumination. In this paper three novel methods, which enable better analysis of the comet assay images are presented.

2.1 Probabilistic, Iterative Approach

The algorithm introduced here is based on a model of a virtual particle, which performs a random walk on the image lattice. It is assumed, that the probability of a transition of the jumping particle from a lattice point to a point belonging to its neighborhood is determined by a Gibbs distribution, defined on the image lattice with the eight-neighborhood system, [7–9].

Using this model, the image is treated as a realization of a Markov random field and it is assumed that the information on the local image properties is contained in the partition function Z of the local statistical system.

Let the image be represented by a matrix I of size N_r, N_c, and let us introduce a virtual particle, which can perform a random walk on the image lattice visiting its neighbors or staying at its temporary position. In this work it is assumed, that the particle moves on the image lattice with the probabilities of a transition from the point (i,j) to (k,l) derived from the Gibbs distribution formula

$$P\left\{(i,j),(k,l)\right\} = \frac{\exp\left\{-\beta\left(I(i,j) - I(k,l)\right)\right\}}{Z(i,j)},\ Z(i,j) = \sum_{(m,n)\Leftrightarrow(i,j)} \exp\left\{-\beta\left(I(i,j) - I(m,n)\right)\right\}, \quad (1)$$

where the symbol $\Leftrightarrow$ denotes the neighborhood relation, β plays the role of the inverse of temperature of the statistical system and $Z(i,j)$ is the partition function, (statistical sum) of the local structure, $I(i,j) \in [0,1],\ \ i = 1, \ldots N_r\,, j = 1, \ldots N_c$.

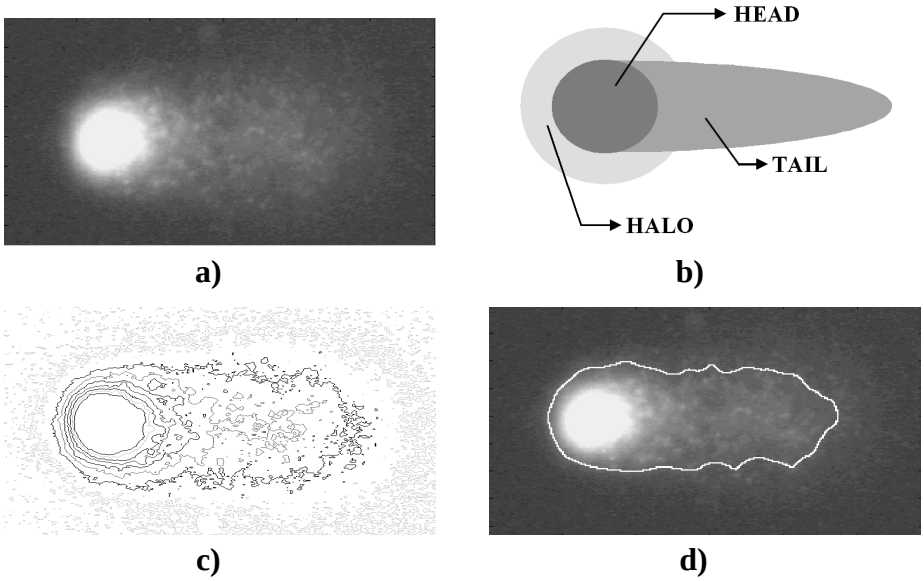


Figure 2: Comet assay image **(a)** and its model **(b)**, below the intensity-sliced image **(c)**, and manually determined boundary of the comet, **(d)**.

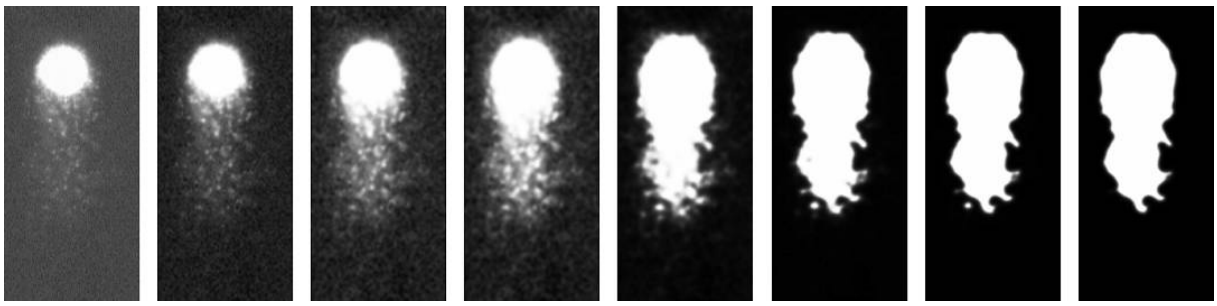


Figure 3: Image segmentation using the probabilistic, iterative approach. The pictures show the evolution of the test image, used for the comet's segmentation, in successive iterations.

As we are interested in the statistical sum Z, which can serve as a measure of the pixel's relation to its neighbors, let us assume that the value of the pixel (i,j) is set to zero. Under this assumption

$$P^*\{(i,j),(k,l)\} = \frac{\exp\{\beta\, I(k,l)\}}{Z^*(i,j)},\ Z^* = \sum_{(m,n)\,\Leftrightarrow\,(i,j)} \exp\{\beta\, I(m,n)\},\ \ I(i,j)=0. \tag{2}$$

The probability that the virtual particle will stay at its current position (i,j) with $I(i,j) = 0$ will not escape from (i,j) is then given by

$$P^*\{(i,j),(i,j)\} = \frac{1}{Z^*(i,j)} = \left[\sum\nolimits_{(m,n)\,\Leftrightarrow\,(i,j)} \exp\{\beta\cdot\, I(m,n)\}\right]^{-1}, \tag{3}$$

Assigning to each image point the probability that the randomly jumping particle will stay at its current position leads to a map of probabilities, which can be treated as a new image. The successive iterations, lead to a binary image consisting of the comet head and its tail, (see Figs. 3 and 7a).

It is worth noticing that the proposed iterative segmentation is insensitive to the noise contamination and illumination conditions as shown in Figs. 4 and 5 respectively.

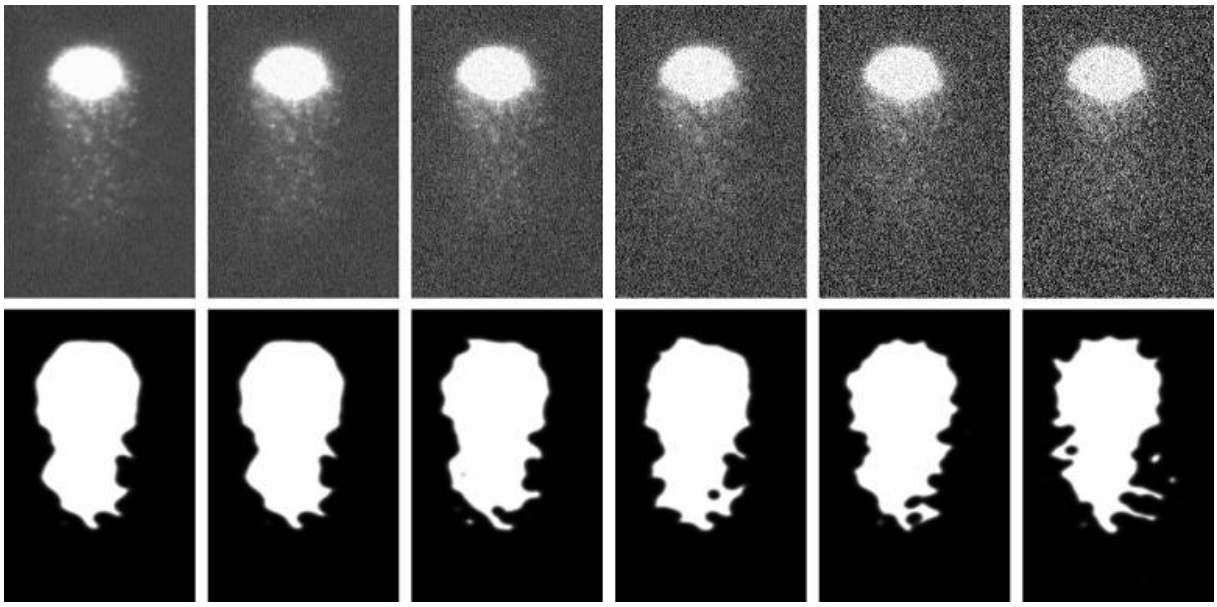


Figure 4: Image segmentation results using the probabilistic, iterative approach. The comet assay image was obtained under noisy conditions with increasing intensity.

2.2 Region Based Segmentation

In [10] an effective image segmentation method, which works without defining the seeds needed to start the segmentation process, was proposed. This method, originally developed for the vector valued color images, can be also applied for the segmentation of gray level images.

At the beginning of the algorithm, each pixel has its own label, (the image consists of one-pixel regions). In the construction of the algorithm, the 4-neighborhood system was used to increase the computational efficiency of the method. For the region growing process, the centroid linkage strategy was applied. This strategy adds a pixel to a region if it is 4-connected and has a color or gray scale value lying in a specified range around the mean value of an already constructed region.

After the inclusion of a new pixel, the region's mean color value is being updated. For this updating, recurrent scheme can be applied. In the first step of the algorithm, a simple raster scan of the image pixels is employed: from left to right and from top to bottom. Next pass, in this two-stage method, starts from the right bottom corner of the image. This pass permits additional merging of the adjacent regions, which after the first pass, possess features satisfying a predefined homogeneity criterion.

During this merging process, each region with a number of pixels below a specified threshold is merged into a region with a larger area, if the homogeneity criterion is fulfilled. After the merging, a new mean color (intensity) of a region is calculated and the labels of pixels belonging to a region are modified. The segmentation results are strongly determined by the design threshold, which defines the homogeneity criterion.

The segmented image can be further post-processed by removing small regions that are usually not significant in further stages of image processing. Their intensities are different from the intensity of the object and its background. Post-processing needs additional third pass from the top left corner to the bottom right corner, whose aim is to remove the regions, which consist of a number of pixels smaller than a certain area threshold. During this algorithm step, small regions are merged with the neighboring regions, which are closest in terms of a color or intensity distance.

The described region-based segmentation technique has been applied to the segmentation of gray level comet assay images. As already mentioned, the segmentation technique works also for single channel images. The only difference is that instead of the color distance between pixels in a specific color space, the absolute difference of their gray scale values is used.

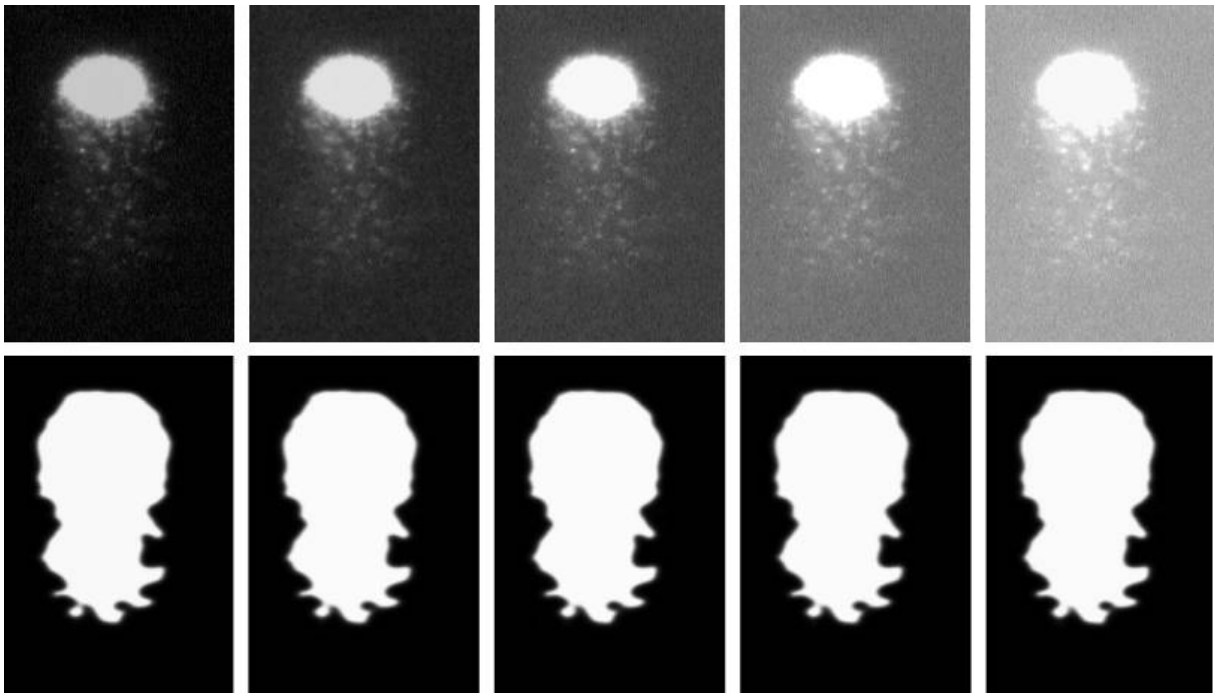


Figure 5: Image segmentation results obtained using the probabilistic, iterative approach. The comet assay image was obtained under conditions with increasing illumination intensity.

The results of the segmentation of comet assay images using the described technique are presented in Fig. 7b). As can be seen this method detects well the comet head and tail. Of course the results are slightly different from those delivered by the previous algorithm, however they correspond well with the assessment of a human observer.

2.3 Active Contour Segmentation

In the past decades image segmentation has played an increasingly important role in medical imaging. Image segmentation still remains a difficult task, due to tremendous variability of medical objects shapes and the variations of image quality affected by different sources of noise and sampling artifacts. To address these difficulties, deformable contours have been extensively studied and widely used in medical image segmentation.

Deformable contours are curves defined within an image domain that can evolve under the influence of internal and external forces. The internal forces, which are defined within the curve itself, are designed to keep it smooth during deformations. They hold the curve together through elasticity forces and keep it from too much bending through the bending forces, (see Fig. 6), [11–14].

The external forces, which are computed from the image data, are defined to move the model toward an object boundary and attract the curve toward the desired object boundaries. The evolution of an active contour can be described as a process of minimization of a functional representing the contour energy, consisting of internal and potential energy terms.

The internal energy specifies the tension or the smoothness of the contour, whereas the potential energy is defined over the image domain and has local minima at the image edges. A deformable contour is a curve $X(s) = \{X(s), Y(s)\}, \ s \in [0,1]$, which evolves on the image domain to minimize the energy functional

$$E(s) = E_{int}(X) + E_{pot}(X), \quad \text{where} \quad \mathrm{E_{int}(X)} = \frac{1}{2}\int_0^1 \alpha(\mathrm{s}) \left|\frac{\partial \mathrm{X}}{\partial \mathrm{s}}\right|^2 + \beta(\mathrm{s}) \left|\frac{\partial^2 \mathrm{X}}{\partial^2 \mathrm{s}}\right|^2 \mathrm{ds}, \quad (4)$$

is the internal energy.

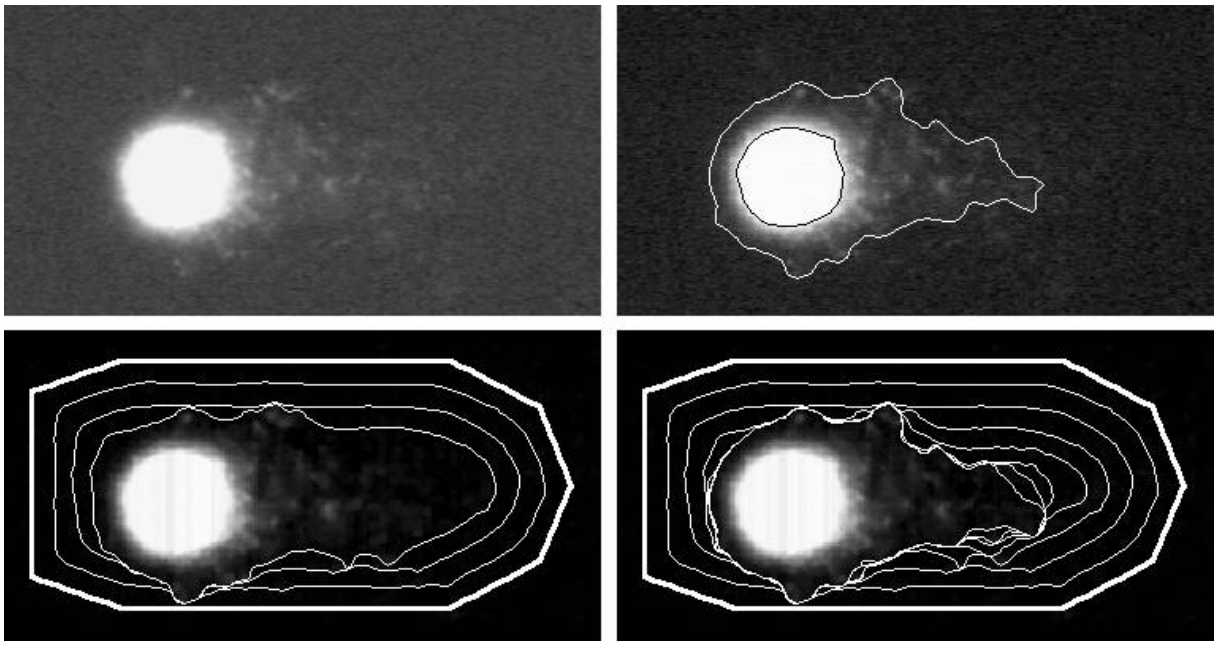


Figure 6: Test image and beside the result of the comet's segmentation using the active contour approach. Below the initial contour (thick line) and the contour evolution after 30 (left) and 70 (right) iterations.

The first-order derivative discourages stretching and makes the contour behave like an elastic string, while the second order derivative discourages bending and makes the model to behave like a rigid rod. The second term is the potential energy

$$E_{pot}\left(X\right) = \int_0^1 P(X(s))\ ds, \tag{5}$$

where the potential function $P(x,y)$ is derived from the image data and takes smaller values at object boundaries. If $I(x,y)$ denotes the gray level value at (x,y), then

$$P\left(x,y\right) = -w\left|\nabla\ \left[G_\sigma\left(x,y\right) * I\left(x,y\right)\right]\right|^2 \tag{6}$$

where $G_\sigma(x,y)$ is a two-dimensional Gaussian, $*$ denotes the convolution operation and w is a parameter. The curve that minimizes the total energy must satisfy the Euler - Lagrange equation

$$\frac{\partial}{\partial s}\left(\alpha\frac{\partial X}{\partial s}\right) - \frac{\partial^2}{\partial s^2}\left(\beta\frac{\partial^2 X}{\partial s^2}\right) - \nabla P\left(X\right) = 0. \tag{7}$$

This equation says that $E_{int}(X) + E_{pot}(X) = 0$, where the internal force is given by $F_{pot} = -\nabla P\left(X\right)$. To find a solution of the energy minimization problem, the deformable contour is made dynamic by treating $X(s)$ as a function of time $X(s,t)$. Then we have to solve

$$\gamma\frac{\partial X}{\partial t} = \frac{\partial}{\partial s}\left(\alpha\frac{\partial X}{\partial s}\right) - \frac{\partial^2}{\partial s^2}\left(\beta\frac{\partial^2 X}{\partial s^2}\right) - \nabla P\left(X\right). \tag{8}$$

When the solution $X(s,t)$ stabilizes, the left side of the above equation is 0 and we achieve a solution of the total energy minimization. In practical applications special external forces, (damping force, multi-scale potential force, pressure forces, distance potential force, dynamic distance force, interactive forces etc.) can be added to the energy minimization scheme.

The results of the segmentation of comet assay images using the described active contour technique are presented in Fig. 7c). In practical applications, the initial contour can be the rectangle placed at the image boundaries.

3 Conclusions

The single cell gel electrophoresis is a powerful tool that can indicate lesions in nucleus DNA caused by various genotoxic agents. However, the lack of standardization is a serious obstacle for evaluating and comparing results obtained in different laboratories. In this paper three novel methods of comet's tail and head extraction were proposed.

As can be seen the presented methods detect well the comet's tail, despite the strong noise present in the comet assay images. The results obtained using different algorithms are naturally not identical, however they all correspond well with the assessment of experts. In the future work we will examine, which of the proposed segmentation methods yields the best results in the practical evaluation of the comet assay results.

References

[1] Rydberg B., Johanson K.J., Estimation of DNA strand breaks in single mammalian cells, in "DNA Repair Mechanisms", P.C. Hanawalt, E.C. Friedberg, C.F. Fox - editors, Academic Press, New York, 465-468, 1978.

[2] Östling O., Johanson K.J., Microelectro-phoretic study of radiation-induced DNA damage in individual mammalian cells, Biochemical and Biophysical Research Communications, 123, 291-298, 1984.

[3] Singh N.P., McCoy M.T., Tice R.R., Schneider E.L., A simple technique for quantification of low levels of DNA damage in individual cells, Experimental Cell Research, 175, 184-191, 1988.

[4] Fairbairn D.W., Olive P.L., O'Neil K.L., The comet assay: a comprehensive review, Mutation Research, 339, 37-59, 1995.

[5] Olive P.L., DNA damage and repair in individual cells: applications of the comet assay in Radiobiology, Int. J. Radiat. Biol., 75, 4, 395-405, 1999.

[6] Wojewódzka M., Kruszewski M., Iwanienko T., Collins A.R., Szumiel I., Application of the comet assay for monitoring DNA damage in workers exposed to chronic low-dose irradiation, Mutation Research, 416, 21-35, 1998.

[7] Smolka B., Wojciechowski K., A new method of texture binarization, Lecture Notes in Computer Science, 1296, 629-636, 1997.

[8] Smolka B., Wojciechowski K., Contrast enhancement of badly illuminated images, Lecture Notes in Computer Science, 1296, 271-278, 1997.

[9] Smolka B., Wojciechowski K.W., Random walk approach to image enhancement, Signal Processing, 81, 465-482, 2001.

[10] Palus H., Bereska D., Region-based Colour Image Segmentation, Proc. of the 5th Workshop on Color Image Processing, Ilmenau, Germany, 67-74, 1999.

[11] Kass M., Witkin A., Terzopoulos D., Snakes: active contour models, Int. J. Comp. Vision, 1, 321-331, 1987.

[12] Xu C., Prince J.L., Snakes shapes and gradient vector flow, IEEE Trans. Imag. Proc. 7, 359-369, 1998.

[13] Cassales V., Kimmel R., Sapiro G., Geodesic active contours, Int. J. Comp. Vision, 22, 61-69, 1997.

[14] Singh A., Goldgof D., Terzopoulos D., IEEE Computer Society Press, Los Alamitos, CA, USA, 1998.

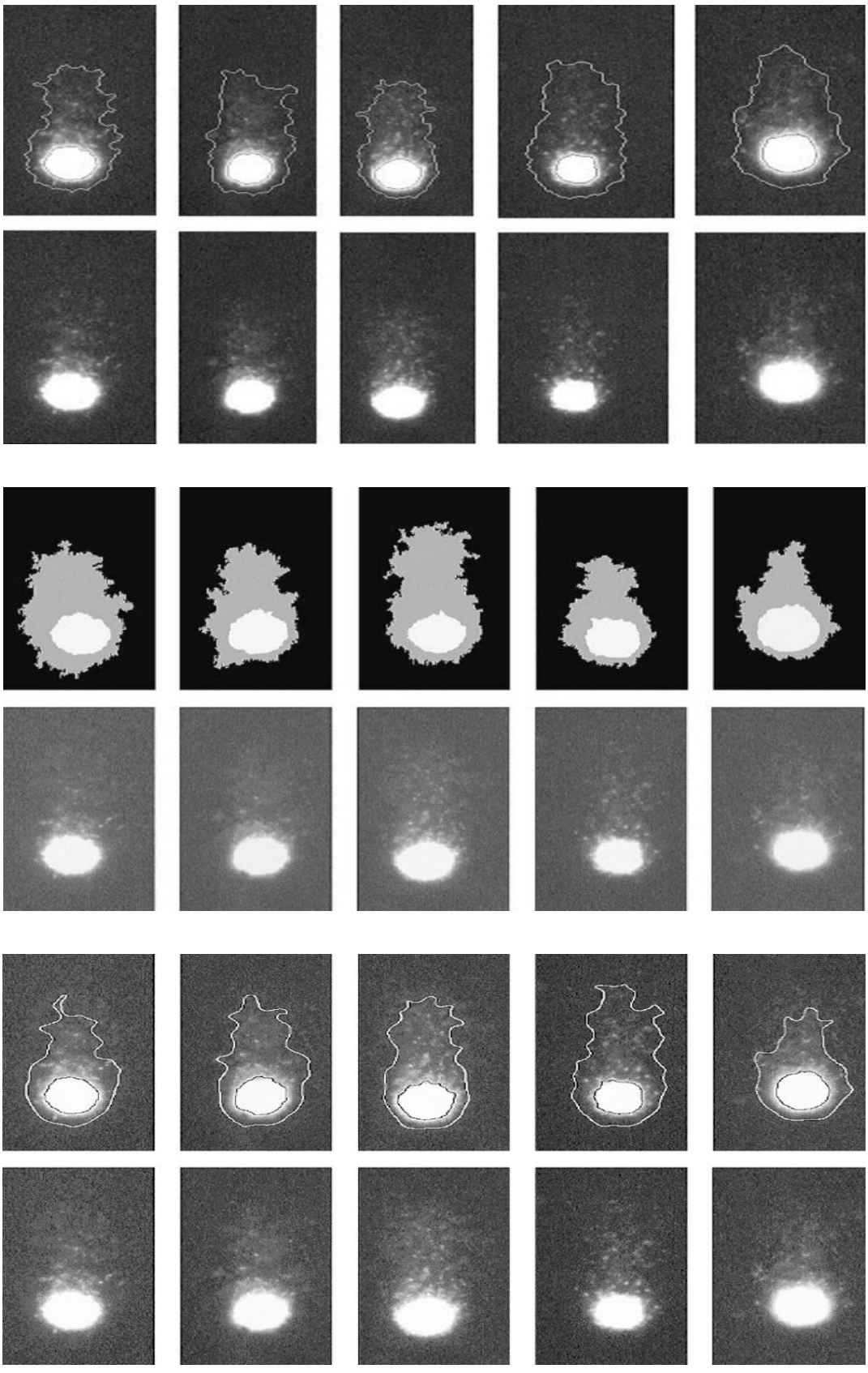


Figure 7: Results of the comet tail and head extraction, (to the left the original comet assay images, beside the extracted tail and head boundaries), obtained with the use of: (a) the probabilistic approach, (b) the region-based method, (c) the active contour method.

UNSUPERVISED IMAGE SEGMENTATION AND BOUNDARY DETECTION USING INFORMATION GAIN

H. Singh and R. Zwiggelaar
School of Computing Sciences,
University of East Anglia,
Norwich NR4 7TJ, England.
email: {harbir.singh, r.zwiggelaar}@uea.ac.uk

Abstract

This paper presents a novel approach to unsupervised segmentation and boundary detection of digital images using an algorithm that utilizes the concept of information theory. *Information Gain* is calculated locally, at a pixel level, resulting in a gain image where high gain occurs at contrasting boundaries and zero gain within homogeneous regions. Subsequently, a multi-scale thresholding approach based on the gain image is used to obtain the optimal segmentation results. The segmentation is guided by both local and global parametric constraints. Comparative evaluation on real and artificial images shows promising results.
Keywords*: Segmentation, Computer Vision, Pattern Recognition, Information Theory*

1 Introduction

Automated image segmentation is an important processing step with widespread applications in performing computer vision tasks such as pattern recognition and image retrieval. Image segmentation algorithms classify the picture elements of an image into different classes so that pixels corresponding to an object of interest belong to the same class [7]. Approaches to carrying out automated segmentation can be divided into two groups, namely supervised and unsupervised methods. Interpretation of objects of interest is often application dependent and in case of supervised segmentation priori information is used for image segmentation by incorporating properties of pixels in relation to its neighboring pixels. Unsupervised approaches are undertaken when prior information of objects of interest is not available. Given the importance of unsupervised image segmentation, various methods are reported in the literature. Some of the methods for carrying out unsupervised image segmentation include the Bayesian approach [1], Markov trees and complex wavelets [11], histogram clustering [9], neuro-fuzzy systems [8], higher-order hidden Markov chains [3] and the use of information theory and entropy [12, 10].

The rest of the paper is organized as follows. In Sec. 2 three existing and novel segmentation approaches are described. A comparative study on real and artificial data is presented in Sec. 3. Conclusions are provided in Sec. 4.

2 Segmentation methods

Image segmentation carried out by Deng and Manjunath [2], results in a J-image while the method undertaken by Jing et. al. [5], gives an H-image. Our method introduces the concept of G-image, which is based on information gain and forms the basis of the segmentation process. In addition, we include

an edge-detection approach as described by Lindeberg [6]. It should be made clear that this method was developed as an edge detector and was as such slightly different from the previous two and proposed method which form the initial step in a segmentation process. However, as will be shown, there is also similarity in the resulting images which is the main reason for including this edge detector.

2.1 J-Image [2]

A region growing method based on image quantization called JSEG is proposed in [2]. The image pixels are first replaced by quantized values forming a class-map of the image where a criterion for good segmentation is defined as in Eq. 3 . Let Z be the set of all image data points in the class-map. Let $z = (x, y), z \epsilon Z$, and m be the mean of all data points. Suppose color has been quantized into C levels, thus Z is classified into C classes, $Z_i, i = 1, ...C$. Let m_i be the mean of the image points in class Z_i. Let

$$S_T = \sum_{z \epsilon Z} \|z - m\|^2 \tag{1}$$

and

$$S_W = \sum_{i=1}^{C} \sum_{z \epsilon Z_i} \|z - m_i\|^2 \tag{2}$$

A criterion for good segmentation is defined as

$$J = \frac{(S_T - S_W)}{S_W} \tag{3}$$

J is a measure of the distances between different classes over the distances between the members within each class S_W. This is similar to Fisher's multi-class linear discriminant. Applying the criterion to local windows in the class-map results in the J-image, in which high and low values correspond to possible region boundaries and region centers. Finally, a region growing method is used to segment the image based on the J-image.

2.2 H-Image [5]

Jing et. al. proposed a similar method to JSEG but with a simpler segmentation criteria, which could be calculated directly from the original image instead of the class-map, and therefore no initial quantization was required. To quantize the homogeneity of a pattern an H-image was derived with each pixel value being replaced by the calculated H value. The pixels of an image were viewed as a set of spatial data points located in a 2D plane with the top left corner being the origin. A pixel was denoted as (x, y) with intensity $I(x, y)$. P was a pattern to compute homogeneity and considered to be a square window of width $2N + 1$. If $c = (x_c, y_c)$ be the center of the pattern with the intensity being $I(x_c, y_c)$ then each pixel $p_i = (x, y)$, $1 \leq i \leq (2N + 1)^2$ in P corresponded to a vector $cp_i = (x_i - x_c, y_i - y_c)$. Based on cp_i a new vector f_i was constructed where

$$f_i = (I(x_i, y_i) - I(x_c, y_c)) \frac{cp_i}{\|cp_i\|} \tag{4}$$

A sum of all the vectors defined in P was taken to be f, i.e.

$$f = \sum_{i=1}^{(2N+1)^2} f_i \tag{5}$$

Finally the measure H was defined as the norm of f, i.e. , $H = \|f\|$.

Based on the H value a H-image was derived. This is a grayscale image whose pixel values were the H values calculated over local windows centered on those pixels. The dark and bright areas in the H-image which represented the region centers and region boundaries were used in carrying out region growing based on local homogeneity analysis.

2.3 L-Image [6]

Lindeberg describes an approach to edge detection, based on first order derivatives [6]. He proposes several different measures of edge strength. We have used the $G_{\gamma-norm}L$ index. The edge-strength is given by

$$G_{\gamma-norm}L = t^{\gamma}(L_x - L_y)^2 \tag{6}$$

where t is the scale, γ is a normalisation constant, and L_x and L_y are the first order derivatives with respect to the subscripts. We used $\gamma = 0.75$ in our experiments, as suggested by the author.

2.4 G-Image : Gain-based Segmentation

This segmentation method is based on regions growing using N_8 pixel connectivity and incorporates information gain heuristic at the pixel processing stage. We consider a grayscale image, $I(x, y)$, of size $M \times N$ where $x\epsilon[0, M)$ and $y\epsilon[0, N)$. Each pixel is characterized by a grayscale value, which is restricted to one of L possible values $0, 1...L-1$, where maximum $L = 256$ gives a 8 bit quantization scheme. An image may be assumed to consist of J regions, each of these is represented by a class $J = 1, 2...J-1$ with $J = 0$, representing background. Gain is calculated at each pixel using N_8 connectivity to obtain a G-image.

Let set *S* consist of *ns* data points in the N_8 neighborhood of a candidate pixel. The intensity value of each pixel is $I(x, y)$. Considering N_8 neighborhood, our sample *S*, will consist of 9 points. A pixel can be included in the region growing process or it can be excluded. Hence there are two classes which the sample points will be classified into. If the I of each pixel is less than the global threshold T, then that pixel is assigned a class of inclusion and if the I of a pixel is more that the threshold T, then that pixel is assigned a class of exclusion. Given that our class label *C* has two values, (*C=include* or *C=exclude*), let ns_i be the number of pixels belonging to class C_i in our sample.

The expected information for the whole sample *S*, is obtained as

$$EI(S) = -\sum_{i=1}^{m} p_i log_2(p_i) \tag{7}$$

where p_i is the probability of occurrence of pixel s_i with class C_i in the arbitrary sample and is given by ns_i/ns.

For estimating local constraint, let I_i be the intensity at each pixel i, μ_i be the mean of all pixels in the N_8 neighbourhood of pixel i and σ_i be the standard deviation of all pixels in the neighbourhood. The pixels at which the following condition

$$\sqrt{(I_i - \mu_i)^2} < \sigma_i \tag{8}$$

is satisfied are chosen for inclusion in the region growing process. In other words if at a pixel the above condition is satisfied, then that pixel is assigned a class of inclusion else it is assigned a class of exclusion. The pixels which are assigned a class of inclusion, are taken to represent the greatest entropy reduction at the candidate pixel and reflect the least randomness or impurity at the candidate pixel [4].

The expected information *E*, at a given sample pixel s_i is given as

$$E(s_i) = p(s_i) * EI(s_i) \tag{9}$$

where p(s_i) is the probability of sample point s_i.

Total gain is defined as

$$G = EI(S) - E(s_i) \tag{10}$$

Based on the G values, a G-Image is derived. This is a grayscale image whose pixel values are the G values calculated over local windows centered on those pixels. The areas with a high G value in the G-image represent region boundaries while areas with low or zero values represent homogeneous patterns.

2.4.1 Algorithm outline

The steps of the algorithm are outlined below.

1. Initialize the algorithm by providing a grayscale image X(*i,j*) and a start global threshold value *T* = x_s and an end global threshold value *T* = x_e
2. For each global threshold level ranging from *T* = x_s to *T* = x_e perform the following steps
   - 2.1 Using N_8 pixel connectivity, for all pixels with intensity below global threshold assign class of inclusion
   - 2.2 Using N_8 pixel connectivity, for all pixels with intensity above global threshold assign class of exclusion
   - 2.3 Calculate expected information of pixels in N_8 neighbourhood using Eq. 7
   - 2.4 At the current pixel, using N_8 pixel connectivity, assign class to current pixel according to criteria in Eq. 8
   - 2.5 Calculate expected information at the current pixel using Eq. 9
   - 2.6 Calculate gain at the current pixel using Eq. 10
3. Display and store G-image
4. Repeat from step 2

3 Results and Discussion

We tested our method on a variety of images as shown in Fig. 1. Column (a) shows the original images, column (b) is the resultant G-image based on our segmentation method, column (c,d) are the resulting H-image and J-image, respectively. We also compared our method with an edge detection method developed by Lindeberg [6]. The implementation of this on our test images is shown in Fig. 2.

Our method performs equally well as compared to the other methods. In the first row of Fig. 1, the gain image shows sufficient detail and is successful in identifying features such as the eyes, hat and face in the image. In the second row, the gain image shows the intricate pattern of spots detected. Minute details such as the whiskers were also picked up. The performance closely matched the result of the H-image. The third row shows results based on lung CT data. Our method is successful in identifying the two lung region along with the trachea in the center and a few structures of interest within the two lung regions. This is again comparable to the brightest lines in the H-image. Lastly, in the last row we test the method on an artificial image of different object shapes with the same contrast. Once again the performance of the method is comparable to the other methods in identifying the object regions. These results indicate a clear boundary detection between classes for both the H and G images and as such provide an appropriate starting point for image segmentation. It should be noted that the H images seem to represent a noisy version of the G images. When comparing the result with the J images, these seem less well defined and as such might provide a poor starting point for segmentation.

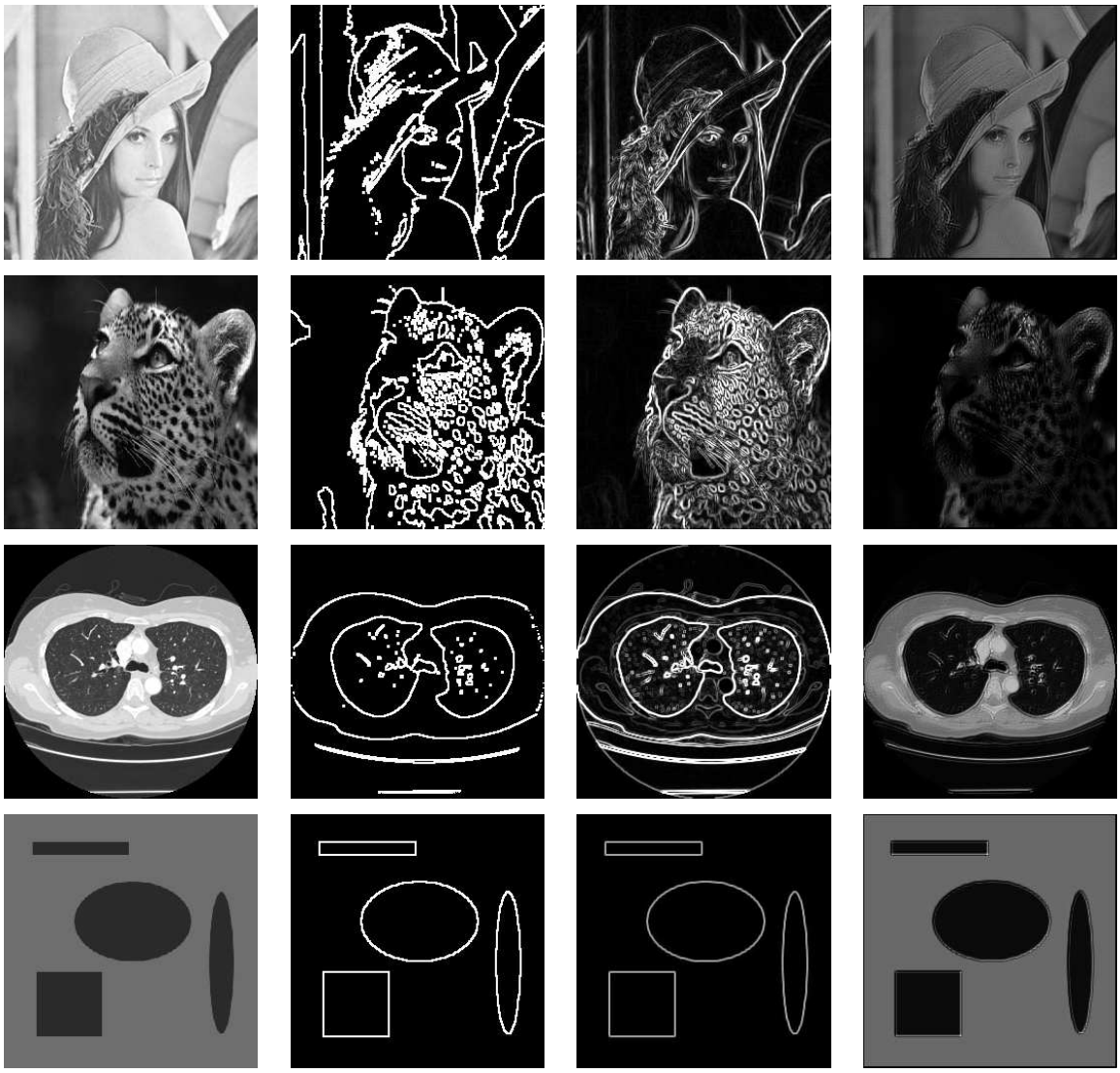


Figure 1: Segmentation results: (a) original image, (b) G-image at optimum threshold, (c) H-image, (d) J-image.

In our implementation of J-image we used quantization of greyscale images into 64 bins. The original algorithm deals with color images and uses a more complex quantization method based on peer-group filtering [2] where high and low J values correspond to possible boundaries and interiors of color texture regions. Deng and Manjunath mention that even though JSEG can be applied on grayscale images, the result are reasonable to an extent but not as good as color image ones as intensity alone is not as discriminative as color.

In addition when comparing with the L images (see Fig. 2), it is clear that the obtained class boundaries in the H and G images are a subset of the detected edges. On the other hand, the L images provide more detail as weak edges which do not represent class boundaries are also highlighted.

Accurate medical image segmentation to extract relevant parts of the anatomy is a crucial precursor for diagnosis and quantitative analysis. Some CT lung image results are shown in Fig. 3. A CT lung slice for the mid-thoracic region was segmented at different global threshold values, T. This shows that depending on the value of T various anatomical structures are extracted, e.g. at high T values the rib bones are found whilst at lower values soft tissue class boundaries are enhanced. To obtain the results

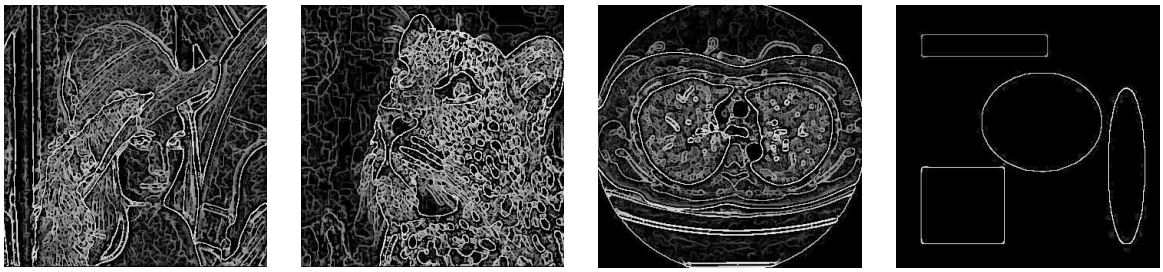


Figure 2: Edge detection as described in Sec. 2.3, see Fig. 1(a) for the original images.

provided in Fig. 1 these individual threshold results are combined to provide the overall most likely class boundaries.

To further ascertain the utility of our method we show in Fig. 4 attempts at object selection dependent on contrast. Image (a) shows contrasting objects. Based on different global threshold values, we were able to select the objects as shown in the subsequent images. This would be difficult using methods which merely detect edges of objects and added steps of region labelling and selection based on region labels would be required.

Although not covered here, the extension of the developed G images can easily be extended to G volumes and as such can be used for anatomical segmentation of volumetric medical data, such as CT or MRI.

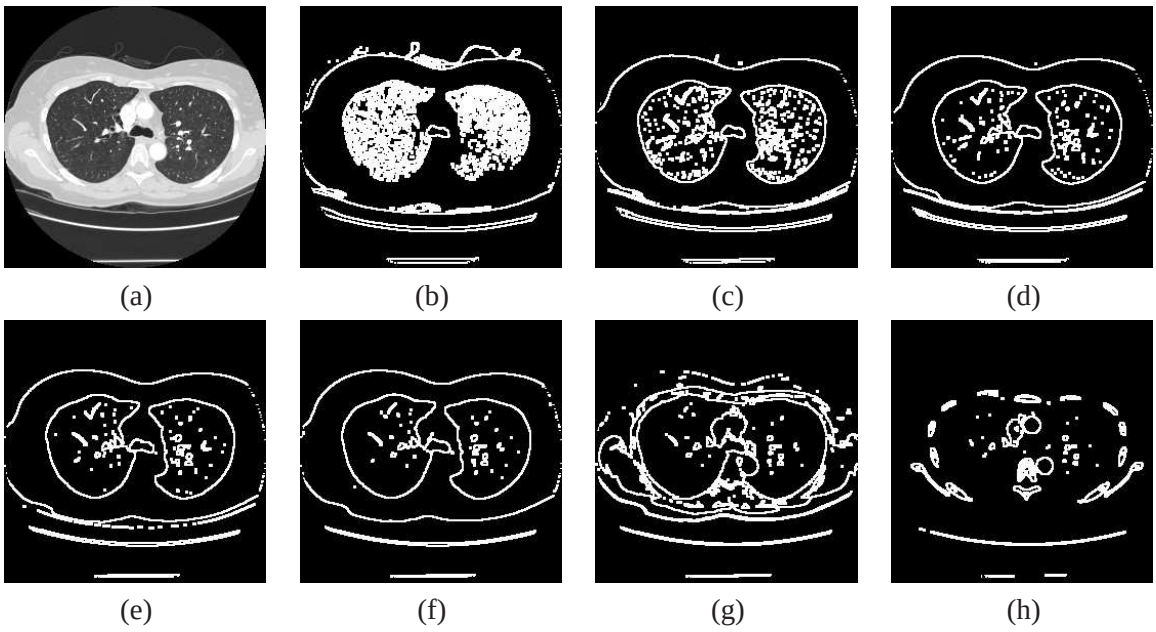
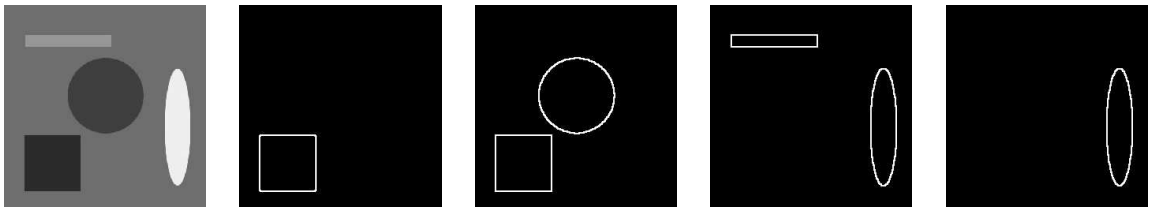


Figure 3: Sequence of CT lung images at different global threshold levels: (a) original image, (b) T=60, (c) T=90 (d) T=120, (e) T=150, (f) T=180, (g) T=210, (h) T=243.

4 Conclusion

We have presented a novel approach to carrying out segmentation when little prior knowledge is known about the scene. In addition we have also compared our method with existing techniques highlighting the uniqueness of our method. In future we intend to extend our analysis for carrying out unsupervised segmentation to 3D volumes and do further analysis in the region growing and merging area. Extension



(a) (b) (c) (d) (e)

Figure 4: Effect of contrast: (a) original image, (b) T=50, (c) T=70, (d) T=120, (e) T=160.

to color images is also planned where an approach similar to [5] could be applied to the three RGB color values and the results combined by taking the norm of the RGB component results.

References

[1] L. Cheng and T. Caelli. Unsupervised image segmentation: a bayesian approach. *The 16th International Conference on Vision Interface, Halifax, Canada*, 7(2), June 2003.

[2] Y. Deng and B.S. Manjunath. Unsupervised segementation of color-texture regions in images and video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(8):800–810, 2001.

[3] C. Derrode, C. Carincotte, and S. Bourennane. Unsupervised image segmentation based on high-order hidden markov chains. *IEEE International Conference on Acoustics, Speech, and Signal Processing, Montreal, Canada*, May 2004.

[4] J. Han and M. Kamber. *Data Mining Concepts and Techniques*. Morgan Kaufmann, San Francisco CA, 2001.

[5] F. Jing, M. Li, H. Zang, and B. Zang. Unsupervised image segmentation using homogeneity analysis. *Proc. IEEE International Symposium on Circuits and Systems*, 2:456–459, May 2003.

[6] T. Lindeberg. Edge detection and ridge detection with automatic scale selection. *International Journal of Computer Vision*, 30(2):117–154, 1998.

[7] D. Marr. *Vision*. Freeman, San Francisco CA, 1982.

[8] H.H. Muhammed. Unsupervised hyperspectral image segmentation using a new class of neuro-fuzzy systems based on weighted incremental neural networks. *31st Applied Image Pattern Recognition Workshop, Washington, D.C., USA*, pages 171–177, 2002.

[9] J. Puzicha, T. Hofmann, and J.M. Buhmann. Histogram clustering for unsupervised image segmentation. *Pattern Recognition Letters*, 20(9):899–909, 1999.

[10] P.K. Saha and J.K. Udupa. Optimum image thresholding via class uncertainity and region homogeniety. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(7):689–706, 2001.

[11] C.W. Shaffrey, N.G. Kingsbury, and I.H. Jermyn. Unsupervised image segmentation via markov trees and complex wavelets. *IEEE International Conference on Image Processing*, 3:801–804, Sept. 2002.

[12] C.F. Sin and C.K. Leung. Image segmentation by edge pixel classification with maximum entropy. *International Symposium on Intelligent Multimedia, Video and Speech Processing*, pages 283–286, May 2001.

ORIENTED PARTICLE SPRAY: PROBABILISTIC CONTOUR TRACING WITH DIRECTIONAL INFORMATION

François Pitié*, Anil C. Kokaram and Rozenn Dahyot
Electronic & Electrical Engineering
Trinity College Dublin
Ireland
email: fpitie@mee.tcd.ie, anil.kokaram@tcd.ie, dahyot@mee.tcd.ie

Abstract

Contour following is a standard activity in rotoscoping in the digital post production domain. An artist might need to *cut out* or edit an object separately from its background and it is left to the artist to manually create the cut out. Techniques for automatically tracing the edges of the object exist, but these operate with heavy manual intervention. The most recent technique called JetStream is a considerable advance on manual or semi-automatic tracing, but suffers from a lack of direction information in the image. This paper considers the incorporation of this information and so reworks the principle of density propagation for contour following. The approach is more robust than previous methods although inevitably needs user intervention to incorporate image semantics.
Keywords*: Particle Filter, Contour tracking, Rotoscoping, Bayesian Inference, Sequential Importance Resampling, Directional Filters, Steerable Filtering*

1 Introduction

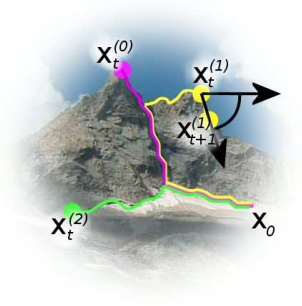
Manual or semi-automatic contour following is an important task in image editing. The tracing of object contours in general is also seen as an important task in early vision [3]. Cut-out tools that assist the user in following a contour, can be seen in Adobe Photoshop for instance. Automated or semi-automated contour following is complicated by the ambiguity of any contour in an image. Not only is it difficult to track exactly the position of a contour because of poor image contrast and noise, but also it is impossible to forsee the contour chosen by the user on the basis of semantics.

Recently, Perez, Blake and Gagnet [4] have proposed a robust technique—called *JetStream*—for contour following that handles this ambiguity by sampling from the posterior distribution for the contour location. It is based on the use of a Particle Filter and its operation can be understood as explained in the following section.

Probabilistic Tracing Approach using Particle Filters

The approach proposed in JetStream [4] to extract a contour can be understood by using an analogy with manual tracing. Starting from a point $\mathbf{x}_0$, the pencil draws a contour by following the edge of the picture. The current position of the pencil at time t is denoted $\mathbf{x}_t$. Tracing the contour can then be understood as *tracking* the pencil. The growing contour is represented by an ordered sequence $\mathbf{x}_{0:t} \equiv (\mathbf{x}_0 \dots \mathbf{x}_t)$.

*This work has been funded by HEA PRTLI TRIP and Enterprise Ireland Grant CASMS



Let θ_{t+1} be the angle formed by the segment $[\mathbf{x}_i; \mathbf{x}_{i+1}]$ with the horizontal axis and let assume that the points are equally spaced by a step d. To simplify the problem, we assume that pencil speed is constant and therefore d is set to $d = 1$.

$$\mathbf{x}_{t+1} = \mathbf{x}_t + d \begin{bmatrix} \cos(\theta_{t+1}) \\ \sin(\theta_{t+1}) \end{bmatrix} \tag{1}$$

The idea of using Particle Filters for tracing is understood more easily with the help of the adjacent figure. While following a contour in the mountain picture, the pencil encounters bifurcations and edge junctions. To select the most likely path, the idea is to try all possible paths and to decide afterwards which one is the best. In our mountain picture example, growing contours $\mathbf{x}_{0:t}^{(0)}$ (in pink), $\mathbf{x}_{0:t}^{(1)}$ (in yellow) and $\mathbf{x}_{0:t}^{(2)}$ (in green) correspond to 3 different possible tracings all originating from the same starting point $\mathbf{x}_0$. The Particle Filter framework—described properly in the next section—proposes to grow simultaneously a number of possible contours—also called *particles*. The particles can take separate decisions when they reach an edge junction. The framework decides whether a particle should grow further, duplicate itself, or stop, depending on its performance.

JetStream, though an elegant solution to a combinatorially difficult problem, suffers from an inability to handle sudden changes in direction without the use of a switching process. In effect, upon encountering a corner, the idea is to propose unconstrained direction possibilities in the expectation that one of the proposed direction will regain a contour 'lock'. This paper resolves the problem by designing a directional probability density function (pdf) that is better able to control the evolution of the contour. Because of the reliability of this pdf it is then possible to relieve the need for heavy control on contour smoothness. The particle filter framework is presented next and the new design explained as problems are highlighted.

2 Probabilistic Contour Tracking Framework

2.1 Standard Approach using Particle Filters

Recall that the ordered sequence $\mathbf{x}_{0:t} \equiv (\mathbf{x}_0 \dots \mathbf{x}_t)$ represents the 2D points of the curve being tracked. This chain is assumed to be a Markov Chain of order 2, ie. $p(\mathbf{x}|\mathbf{x}_{0:t}) = p(\mathbf{x}|\mathbf{x}_t, \mathbf{x}_{t-1})$. Given the observed image represented by a vector $\mathbf{y}$, a probabilistic approach to tracking proceeds by manipulating the posterior, $p(\mathbf{x}_{0:t+1}|\mathbf{y})$ to estimate the most probable next position $\mathbf{x}_{t+1}$. This distribution can be written in a recursive form:

$$p(\mathbf{x}_{0:t+1}|\mathbf{y}) = p(\mathbf{x}_{t+1}|\mathbf{y}, \mathbf{x}_{0:t})\, p(\mathbf{x}_{0:t}|\mathbf{y}) \tag{2}$$

This form admits a solution which manifests as the propagation of densities from point to point on each contour. Bayes rule combined with the Markovian hypothesis on the contour leads to the following expression for the posterior:

$$p(\mathbf{x}_{0:t+1}|\mathbf{y}) \propto \prod_{i=2}^{t+1} p(\mathbf{x}_i|\mathbf{x}_{i-1}, \mathbf{x}_{i-2})\, p(\mathbf{y}|\mathbf{x}_i, \mathbf{x}_{i-1}) \tag{3}$$

It is then possible to show that the following recursion arises:

$$p(\mathbf{x}_{0:t+1}|\mathbf{y}) = p(\mathbf{x}_{t+1}|\mathbf{x}_t, \mathbf{x}_{t-1})\, p(\mathbf{y}|\mathbf{x}_{t+1}, \mathbf{x}_t)\, p(\mathbf{x}_{0:t}|\mathbf{y}) \tag{4}$$

The term $p(\mathbf{x}_{t+1}|\mathbf{x}_t, \mathbf{x}_{t-1})$ corresponds to the *prior* on the contour and $p(\mathbf{y}|\mathbf{x}_{t+1}, \mathbf{x}_t)$ to the *data model*.

Although we might have an analytical expression for the prior and the data model, this expression presents usually no simple closed form. Sequential Monte Carlo methods (also called *particle filters*) provide however a flexible and easy way of propagating an approximation of this posterior distribution. In this framework the posteriors are approximated in a grid-based fashion by a finite set $(\mathbf{x}_{0:t}^{(m)})_{m=1...M}$ of M samples or *particles*:

$$p(\mathbf{x}_{0:t}|\mathbf{y}) \approx \sum_{m=1}^{M} w_t^{(m)} \delta(\mathbf{x}_{0:t} - \mathbf{x}_{0:t}^{(m)}) \tag{5}$$

where $\delta(.)$ denotes the Dirac delta measure which is 1 in 0 and zero otherwise; $w_t^{(m)}$ the importance weight attached to particle $\mathbf{x}_{0:t}^{(m)}$. Note that our particles correspond to contours ($\mathbf{x}_{0:t}^{(m)}$) and not to single 2D points. The posterior approximation can be propagated in time by the generic boostrap filter (or Sequencial Importance Resampling (SIR) Particle Filter) [1, 2] as proposed for instance in JetStream. At each time iteration, the weights are chosen using the principle of *importance sampling* [1, 2]. As we know, it can be difficult to draw directly samples from the posterior $p(\mathbf{x}_{0:t}|\mathbf{y})$. However, it is usually possible to find as a first step a proposal—called *importance density*—from which we can easily draw samples. In the bootstrap filter the proposal is simply the prior density $p(\mathbf{x}_{t+1}|\mathbf{x}_t^{(m)}, \mathbf{x}_{t-1}^{(m)})$ and the weights are therefore given by the likelihood [1, 2]:

$$w_{t+1}^{(m)} \propto \frac{p(\mathbf{x}_{t+1}^{(m)}, \mathbf{x}_t^{(m)}|\mathbf{y})}{p(\mathbf{x}_{t+1}^{(m)}|\mathbf{x}_t^{(m)}, \mathbf{x}_{t-1}^{(m)})} = p(\mathbf{y}|\mathbf{x}_{t+1}^{(m)}, \mathbf{x}_t^{(m)}) \tag{6}$$

To avoid that the weight distribution becomes more and more skewed which leads to the degeneracy of the particles, the bootstrap filter adds a *selection* step. In this crucial step the M growing contours are drawn from the normalised weight distribution. The idea is that 'good' contours will be statistically replicated whereas the 'bad' one will be deleted.

From these approximations of the posterior distribution $p(\mathbf{x}_{0:t}|\mathbf{y})$, an approximation of the Maximum A Posteriori can be derived by taking the 'best' contour.

2.2 Exact Importance Sampling

A good choice for the proposal is key to the success of the particle filter algorithm. In JetStream—as in many tracking algorithms—the importance distribution is however constrained by the smoothness of the particle's trajectory. For instance the trajectory of the contour cannot deviate by more than a few degrees. A special case is made when particles reach a corner: particles are allowed to take any direction. With such hypotheses the position of the next particle is strongly restricted and in our experience, at the price of missing frequently sharp turns in the contour as shown in figure 7. This problem arises due to the difficulty in designing a prior that will both play the role of a good proposal—able to restrict the search area—and that will give enough flexibility to model the dynamics of the contour.

As a key deviation from this classical approach, we propose to reconsider equation 3 and choose directly as the proposal

$$q(\mathbf{x}_{t+1}|\mathbf{y}, \mathbf{x}_{0:t}^{(m)}) = \frac{p(\mathbf{x}_{t+1}|\mathbf{x}_t^{(m)}, \mathbf{x}_{t-1}^{(m)})\; p(\mathbf{y}|\mathbf{x}_t^{(m)}, \mathbf{x}_{t+1})}{\int_{\mathbf{x}_{t+1}} p(\mathbf{x}_{t+1}|\mathbf{x}_t^{(m)}, \mathbf{x}_{t-1}^{(m)})\; p(\mathbf{y}|\mathbf{x}_t^{(m)}, \mathbf{x}_{t+1})\; d\mathbf{x}_{t+1}} \tag{7}$$

By doing so, we take the optimal proposal and we ensure a perfect sampling of the posterior, without any additional constraint on the prior function. The difficulty lies now in drawing the sample $\mathbf{x}_{t+1}^{(m)}$ directly from the proposal. Both prior $p(\mathbf{x}_{t+1}|\mathbf{x}_t^{(m)}, \mathbf{x}_{t-1}^{(m)})$ and likelihood $p(\mathbf{y}|\mathbf{x}_t^{(m)}, \mathbf{x}_{t+1})$ functions will be explicited in section 3 and section 4.

Figure 1: Outlines of the Oriented Particle Spray

1. **Initialisation.** $t = 0$, manually set $\mathbf{x}_0^{(m)} = \mathbf{x}_0$
2. **Importance Sampling Step**
   For each particle m, do:
   - **Prediction:**

$$\mathbf{x}_{t+1}^{(m)} \sim p(\mathbf{x}_{t+1}|\mathbf{x}_t^{(m)}, \mathbf{x}_{t-1}^{(m)}) p(\mathbf{y}|\mathbf{x}_t^{(m)}, \mathbf{x}_{t+1}) \tag{10}$$

   - **Weighting:**

$$w_t^{(m)} = \int_{\mathbf{x}_{t+1}} p(\mathbf{x}_{t+1}|\mathbf{x}_t^{(m)}, \mathbf{x}_{t-1}^{(m)}) p(\mathbf{y}|\mathbf{x}_t^{(m)}, \mathbf{x}_{t+1}) d\mathbf{x}_{t+1} \tag{11}$$

3. **Selection Step.** Resample with replacement M contours from the set $\left(\mathbf{x}_{0:t+1}^{(m)}; m = 1, \ldots, M\right)$ according to the normalised importance weights $w_t^{(m)} / \sum_m w_t^{(m)}$.

The weights are defined by

$$w_{t+1}^{(m)} \propto \frac{p(\mathbf{x}_{t+1}|\mathbf{y}, \mathbf{x}_{0:t}^{(m)})}{q(\mathbf{x}_{t+1}|\mathbf{y}, \mathbf{x}_{0:t}^{(m)})} \tag{8}$$

$$= \int_{\mathbf{x}_{t+1}} p(\mathbf{x}_{t+1}|\mathbf{x}_t^{(m)}, \mathbf{x}_{t-1}^{(m)})\ p(\mathbf{y}|\mathbf{x}_t^{(m)}, \mathbf{x}_{t+1})\ d\mathbf{x}_{t+1} \tag{9}$$

The final outline of our contour tracking algorithm is summarised in figure 1.

3 The Prior on the Contours

As the prior does not serve as a proposal, we can adopt a weak constraint on the dynamic of the contour. We only assume that a particle cannot return to a previous position. This problem—trivial in appearance—has to be handled carefully to avoid that the particles try to rediscover their exact reverse trajectory.

Using the trajectory angle θ, the prior can then be rewritten as

$$p(\mathbf{x}_{t+1}|\mathbf{x}_t^{(m)}, \mathbf{x}_{t-1}^{(m)}) = p(\theta_{t+1}|\theta_t^{(m)}) \tag{12}$$

We propose here a naive solution that disallows angles diametrically opposed to the previous direction angle taken by the particle.

$$p(\mathbf{x}_{t+1}|\mathbf{x}_t^{(m)}, \mathbf{x}_{t-1}^{(m)}) = \phi_b(\mathrm{dist}(\theta_{t+1}, \theta_t^{(m)})) \tag{13}$$

where ϕ_b is a kernel function based on the distance between angles as represented in figure 2.

4 Likelihood

Introducing the angle notation as previously, the likelihood can be reexpressed as

$$p(\mathbf{y}|\mathbf{x}_{t+1}, \mathbf{x}_t^{(m)}) = p(\mathbf{y}|\theta_{t+1}, \mathbf{x}_t^{(m)}) \tag{14}$$

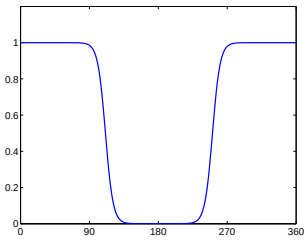


Figure 2: Values of the kernel for $\theta \rightarrow \phi_b(\text{dist}(\theta, 0))$.

which stands for the probability that at pixel $\mathbf{x}_t^{(m)}$ an edge goes along the direction θ_{t+1}. The likelihood presented in JetStream relies mainly on the simple definition of the edge: the angle of the edge is defined by $\theta = \text{atan2}(I_y, I_x)$[1] and its norm by $N = \sqrt{I_x^2 + I_y^2}$, where I_x and I_y are the derivatives of the picture I. This definition presents a strong drawback: it assumes that only one edge passes by the pixel of consideration. In consequence, this approach cannot cope with corners, or junctions. Even if JetStream attempts to handle this problem by using a Harris corner detector beforehand, Figure 7 shows that JetStream still tends to fail quite easily in its tracking. We propose therefore to fully integrate the orientation of the contours in our likelihood function. To do so, we make $p(\mathbf{y}|\theta_{t+1}, \mathbf{x}_t^{(m)})$ explicit by an approach similar to Steerable Filters [6, 5] and more specifically in [7].

Let us assume that the probability that at pixel $\mathbf{x}_t^{(m)}$, the direction θ_{t+1} corresponds to an edge is proportional to the absolute variation of the angular intensity, i.e.:

$$p(\mathbf{y}|\theta_{t+1}, \mathbf{x}_t^{(m)}) \propto \left| \frac{dI_\theta}{d\theta} \right| \tag{15}$$

where the intensity in direction $\theta \in [0; 2\pi]$ I_θ is equal to:

$$I_\theta = \int_{\rho>0} I(\rho, \theta) g(\rho) \, \mathrm{d}\rho \tag{16}$$

(ρ, θ) is a pixel coordinate location in polar coordinates, with origin at the current contour point. The integral is just the sum of pixels along the direction θ. $g(\rho)$ is a smoothing kernel (a gaussian for instance), which ensures that pixels closer to the origin are more important than those further away. Note that $\rho > 0$ since we wish to design a meaningful direction metric.

To interpolate I_θ to all values of θ we can take advantage of the periodicity of I_θ (since the function would repeat every 360^{deg}) and so consider its Fourier series:

$$I_\theta = \sum_{n=0}^{n=N} \mathrm{H}_n e^{jn\theta} \tag{17}$$

and respectively for its derivative:

$$p(\mathbf{y}|\theta_{t+1}, \mathbf{x}_t^{(m)}) \propto \left| \frac{dI_\theta}{d\theta} \right| = \left| \sum_{n=0}^{n=N} n \; j \; \mathrm{H}_n e^{jn\theta} \right| \tag{18}$$

The Fourier coefficients can be computed with:

$$\mathrm{H}_n = \int_{\phi,\rho} I(\rho, \phi) w_n(\rho, \phi) \; \rho \mathrm{d}\phi \mathrm{d}\rho \tag{19}$$

[1] atan2 is $\tan^{-1}$ with unwrapped angles.

where

$$w_n(\rho,\phi) = \frac{1}{\rho} g(\rho) e^{jn\phi} \tag{20}$$

The continuous values of $I(\rho,\theta)$ are obtained by interpolation from the image grid. This can be classically obtained by convolving the sampled picture $\mathrm{I}(x,y)$ with an interpolation kernel k. To simplify notations we will consider cartesian coordinates:

$$\begin{cases} (u,v) & \equiv & (\rho,\phi) = (\sqrt{u^2+v^2}, \mathrm{atan2}(v,u)) \\ (x,y) & \equiv & (r,\psi) = (\sqrt{x^2+y^2}, \mathrm{atan2}(y,x)) \end{cases} \tag{21}$$

$$\begin{aligned} \mathrm{H}_n &= \int_{u,v} (\mathrm{I} * k)(u,v) w_n(u,v)\, \mathrm{d}u\mathrm{d}v & (22)\\ &= \int_{u,v} \left(\sum_{x,y} \mathrm{I}(x,y) k(u-x, v-y) \right) w_n(u,v)\, \mathrm{d}u\mathrm{d}v & (23)\\ &= \sum_{x,y} \mathrm{I}(x,y) \int_{u,v} k(u-x, v-y) w_n(u,v)\, \mathrm{d}u\mathrm{d}v & (24) \end{aligned}$$

By making explicit the interpolation kernel in this way, we are able to derive a complete framework for calculation of the direction information. Finally we have:

$$\begin{aligned} \mathrm{H}_n &= \sum_{x,y} \mathrm{I}(x,y) h_n(x,y) \\ h_n(x,y) &= \int_{u,v} k(u-x, v-y) w_n(u,v)\, \mathrm{d}u\mathrm{d}v \end{aligned}$$

So $\mathrm{H_n}$ can be computed by the use of a filter bank whose mask $h_n(x,y)$ can be computed offline. We still need to make explicit the kernels k and g. Here is a possible implementation:

$$\begin{aligned} g(\rho) &= \frac{1}{\sqrt{2\pi\sigma_g^2}} \exp\left(-\frac{\rho^2}{2\sigma_g^2} \right) \\ k(u-x, v-y) &= \frac{1}{2\pi\sigma_k^2} \exp\left(-\frac{\rho^2 + r^2 - 2r\rho\cos(\psi-\phi)}{2\sigma_k^2} \right) \end{aligned}$$

Figure 4, shows examples of 11-tap filters h_n.

Examples. Figure 5 shows an example of such a pdf. On the right the values of $\left|\frac{dI_\theta}{d\theta}\right|$ correspond to the pdf of the contour directions at the center of the picture on the left. This was obtained for $\sigma_g = 2.25$, $\sigma_k = 0.7$ at order $N = 10$. On the left side, the red lines correspond to the lobes of $\left|\frac{dI_\theta}{d\theta}\right|$.

5 Conclusion

Figure 7 shows some simulations of JetStream (on the left) and the Oriented Particle Spray (on the right). It is visible that JetStream tends to overshoot sharp angles of the contours whereas our method can follow them correctly, for a computational time equivalent to JetStream (the simulations where performed under matlab). This comparison has been carried out without user interaction that is an essential tool in a contour tracing application. The proposed improvements, in dealing better with sharp angles, should henceforth simplify and limit the user efforts.

A further development of this algorithm could be also to automatically extract *all* relevant contours of a picture by letting branches to grow separately after edge junctions.

Figure 3: Summary of the algorithm for computing the likelihood $p(\mathbf{y}|\theta_{t+1}, \mathbf{x}_t^{(m)})$

Offline computations:

$$h_n(x,y) \propto \int_{\rho,\phi} \exp\left(-\frac{\rho^2 + r^2 - 2r\rho\cos(\psi-\phi)}{2\sigma_k^2}\right) \exp\left(-\frac{\rho^2}{2\sigma_g^2}\right) \exp\left(jn\phi\right)\ \mathrm{d}\phi\mathrm{d}\rho \tag{25}$$

with the normalizing constant:

$$C = \frac{1}{2\pi\sigma_k^2\sqrt{2\pi\sigma_g^2}} \tag{26}$$

Online computations:

$$\mathrm{H}_n = \sum_{x,y} \mathrm{I}(x,y) h_n(x,y) \tag{27}$$

$$p(\mathbf{y}|\theta_{t+1}, \mathbf{x}_t^{(m)}) \propto \left|\frac{dI_\theta}{d\theta}\right| = \left|\sum_{n=0}^{n=N} jn\mathrm{H}_n e^{jn\theta}\right| \tag{28}$$

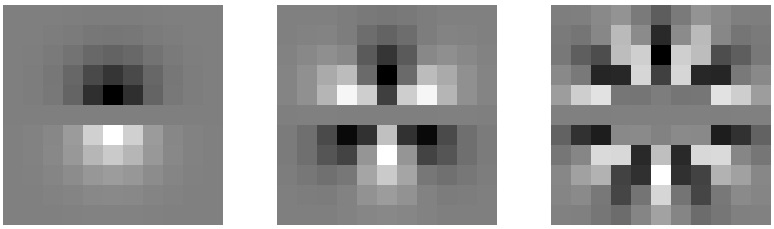


Figure 4: Examples of 11-tap filters $h_n(x,y)$ for $n=1$, $n=3$ and $n=7$.

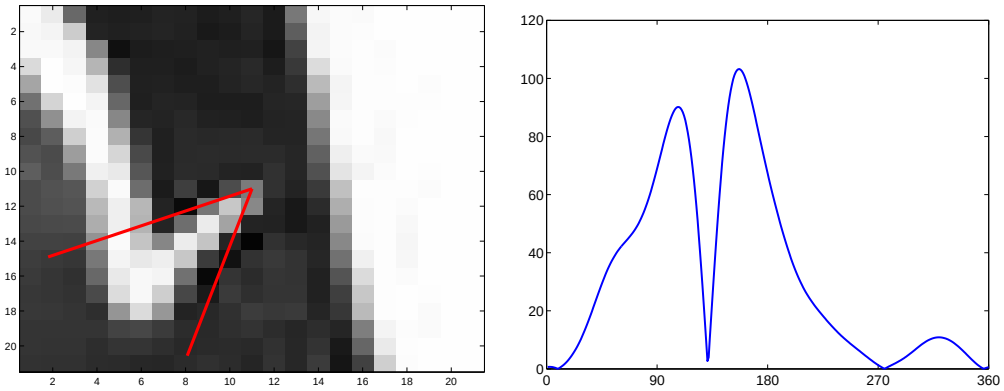


Figure 5: Example of an image (on the left) and the corresponding values of $\left|\frac{dI_\theta}{d\theta}\right|$ for θ in $[0°; 360°]$. On the left the red lines correspond to the directions of maximum variations (lobes of $\left|\frac{dI_\theta}{d\theta}\right|$).

References

[1] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking. 2002.

[2] Arnaud Doucet, Nando de Freitas, and Neil Gordon. *Sequential Monte Carlo Methods in Practice*. Springer, 2000.

Figure 6: Example of the Oriented Particle Spray in action, with on the right a zoom on the multiple hypotheses tracking.

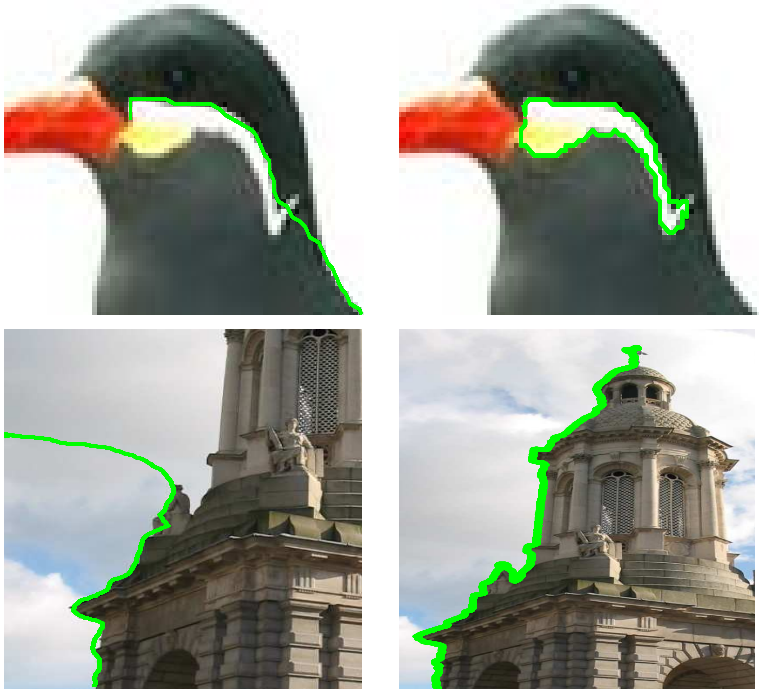


Figure 7: Contour tracings for JetStream on the left column and the Oriented Particle Spray on the right.

[3] David Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. W.H. Freeman, 1982.

[4] P. Pérez, A. Blake, and M. Gangnet. Jetstream: Probabilistic contour extraction with particles. *Proc. Int. Conf. on Computer Vision (ICCV)*, II(5):524–531, 2001.

[5] Pietro Perona. Deformable kernels for early vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(5):488–499, 1995.

[6] Eero P. Simoncelli and Hany Farid. Steerable wedge filters. In *ICCV*, pages 189–194, 1995.

[7] W. Yu, K. Daniilidis, and G. Sommer. Approximate orientation steerability based on angular gaussians, 1999.

NONPARAMETRIC TECHNIQUE OF IMPULSIVE NOISE REMOVAL FOR COLOR IMAGES

Bogdan Smolka
Silesian University of Technology
Department of Automatic Control,
Akademicka 16 Str, 44-100 Gliwice, Poland,
email: bsmolka@ia.polsl.gliwice.pl

Abstract

In this paper the problem of nonparametric impulsive noise removal in multichannel images is addressed. The proposed filter class is based on the nonparametric estimation of the density probability function in a sliding filter window. The obtained results show good noise removal capabilities and excellent structure preserving properties of the new impulsive noise reduction technique.
Keywords*: color image enhancement, impulsive noise removal, image restoration*

1 Introduction

The majority of the nonlinear, multichannel filters are based on the ordering of vectors in a sliding filter window. The output of these filters is defined as the lowest ranked vector according to a specific vector ordering technique.

Let the color images be represented in the commonly used RGB color space and let $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$ be N samples from the sliding filter window W. Each of the $\mathbf{x}_i$ is an m-dimensional multichannel vector, (in our case $m = 3$). The goal of the vector ordering is to arrange the set of N vectors $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ belonging to W using some sorting criterion.

In [1, 2] the ordering based on the cumulative distance function $R(\mathbf{x}_i)$ has been proposed: $R(\mathbf{x}_i) = \sum_{j=1}^{N} \rho(\mathbf{x}_i, \mathbf{x}_j)$, where $\rho(\mathbf{x}_i, \mathbf{x}_j)$ is a function of the distance among $\mathbf{x}_i$ and $\mathbf{x}_j$. The ordering of the scalar quantities according to $R(\mathbf{x}_i)$ generates the ordered set of vectors. The most commonly used measure to quantify distance between two multichannel signals is the Minkowski norm $\rho_\gamma(\mathbf{x}_i, \mathbf{x}_j) = [\sum_{k=1}^{m} |x_{ik} - x_{jk}|^\gamma]^{1/\gamma}$. The Minkowski metric includes the city-block distance ($\gamma = 1$), Euclidean distance ($\gamma = 2$) and chess-board distance ($\gamma = \infty$) as the special cases.

One of the most important noise reduction filter is the vector median. In the case of gray scale images, given a set W containing N samples, the median of the set is defined as $x_{(1)} \in W$ such that

$$\sum_j |x_{(1)} - x_j| < \sum_j |x_i - x_j|, \quad \forall \; x_i, x_j \in W. \tag{1}$$

Median filters exhibit good noise reduction capabilities, (especially when long tailed distribution noise is involved) and outperform simple nonadaptive linear filters in preserving signal discontinuities. As in many applications the signal is multidimensional, in [3] the *Vector Median Filter* (VMF) was introduced, by generalizing the definition (1) using a suitable vector norm. Given a set W of N vectors, the vector median of the set is defined as $\mathbf{x}_{(1)} \in W$ satisfying

$$\sum_j \|\mathbf{x}_{(1)} - x_j\| < \sum_j \|\mathbf{x}_i - \mathbf{x}_j\|, \quad \forall \; \mathbf{x}_i, \mathbf{x}_j \in W. \tag{2}$$

The orientation difference between two vectors can also be used as their distance measure. This so-called vector angle criterion is used by the *Vector Directional Filters* (VDF), to remove vectors with atypical directions, [4]. The *Basic Vector Directional Filter* (BVDF) is a ranked-order, nonlinear filter which parallelizes the VMF operation. However, a distance criterion, different from the distance norms used in VMF is utilized to rank the input vectors. The output of the BVDF is that vector from the input set, which minimizes the sum of the angles with the other vectors. To improve the efficiency of the directional filters, another method called *Directional-Distance Filter* (DDF) was proposed. This filter retains the structure of the BVDF, but utilizes the combined distance criterions to order the vectors inside the processing window, [4, 5].

2 Nonparametric Estimation

Applying statistical pattern recognition techniques requires the estimation of the probability density function of the data samples. Nonparametric techniques do not assume a particular form of the density function since the underlying density of the real data rarely fits common density models.

Nonparametric Density Estimation is based on placing a kernel function on every sample and on the summation of the values of all kernel function values at each point in the sample space, [6, 7]. The nonparametric approach to estimating multichannel densities can be introduced by assuming that the color space occupied by the multichannel image pixels is divided into m-dimensional hypercubes. If h_N is the length of an edge of a hypercube, then its volume is given by $V_N = h_N^m$. If we are interested in estimating the number of pixels falling in the hypercube of volume V_N, then we can define the window function $\phi(\mathbf{x_i}) = 1$, if $|x_{ij}| \leq 1/2$, $j = 1, \ldots, m$ and 0 otherwise, which defines a unit hypercube centered in the origin.

The function $\phi\left(\|\mathbf{x} - \mathbf{x}_i\| / h_N\right)$ is equal to unity if the pixel $\mathbf{x}_i$ falls within the hypercube V_N centered at $\mathbf{x}$ and is zero otherwise. The number of pixels in the hypercube with the length of edges equal to h_N is then $k_N = \sum_{i=1}^{N} \phi\left(\|\mathbf{x} - \mathbf{x}_i\|/h_N\right)$ and the estimate of the probability that a sample $\mathbf{x}$ is within the hypercube is $p_N = k_N/NV_N$, which gives

$$p_N(\mathbf{x}) = (NV_N)^{-1} \sum_{i=1}^{N} \phi\left(\|\mathbf{x} - \mathbf{x}_i\|/h_N\right). \tag{3}$$

This estimate can be generalized by using a smooth kernel function K in place of $\phi(\cdot)$ and the width parameter h_N satisfying: $K(\mathbf{x}) = K(-\mathbf{x})$, $K(\mathbf{x}) \geq 0$, $\int K(\mathbf{x})\, d\mathbf{x} = 1$ and $\lim_{N\to\infty} h_N = 0$, $\lim_{N\to\infty} h_N^m = \infty$.

The multivariate estimator in the m-dimensional case is defined as

$$p_N^*(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{h_1 \ldots h_q} \mathcal{K}\left(\frac{|x_1 - x_{i1}|}{h_1}, \cdots, \frac{|x_q - x_{iq}|}{h_m}\right), \tag{4}$$

with $\mathcal{K}$ denoting a multidimensional kernel function $\mathcal{K}\colon \mathbb{R}^m \to \mathbb{R}$, $h_1, \ldots, h_m$ denoting bandwidths for each dimension and N being the number of samples in W. A common approach to build multidimensional kernel functions is to use a *product kernel* $\mathcal{K}(u_1, \ldots, u_m) = \prod_{i=1}^{m} K(u_i)$, where K is a one-dimensional kernel function

$$p_N^*(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^{N} \prod_{j=1}^{m} \left(\frac{|x_{ij} - x_j|}{h_i}\right). \tag{5}$$

The shape of the approximated density function depends heavily on the bandwidth chosen for the density estimation. Small values of h lead to spiky density estimates showing spurious features. On the other hand, too big values of h produce over-smoothed estimates that hide structural features.

If we chose the Gaussian kernel, then the density estimate of the unknown probability density function at $\mathbf{x}$ is obtained as a sum of kernel functions placed at each sample $\mathbf{x}_i$

$$p_N(\mathbf{x},h)=\frac{1}{N\left(h\sqrt{2\pi}\right)^m}\sum_{i=1}^{N}\exp\left(-\frac{\|\mathbf{x}-\mathbf{x}_i\|^2}{2h^2}\right). \tag{6}$$

The smoothing parameter h depends on the local density estimate of the sample data. The form of the data dependent smoothing parameter is of great importance for the non-parametric estimator.

Choosing the Gaussian kernel function for $\mathcal{K}$, the optimal bandwidth is

$$h^*=(4/(m+2))^{-\frac{1}{m+4}}\,\hat{\sigma}\,N^{-\frac{1}{m+4}}\,, \tag{7}$$

where σ denotes the approximation of the standard deviation of the samples. In one dimensional case (7) reduces to the well known, 'rule of thumb', $h^*=1.06N^{-\frac{1}{5}}\hat{\sigma}$, [6, 7]. A version which is more robust against outliers in the sample set can be constructed if the interquartile range is used as a measure of spread instead of the variance, [6]. This modified estimator is $h^*=0.79\varrho N^{-\frac{1}{5}}\hat{\sigma}$, where ϱ is the inter-quartile range. Another robust estimate of the optimal bandwidth is $h^*=0.9AN^{-\frac{1}{5}}\hat{\sigma}$ with $A=\min\left(\hat{\sigma},\varrho/1.34\right)$. Generally the simplified rule of choosing the optimal bandwidth h can be written as

$$h_1^*=C\,\hat{\sigma}\,N^{-\frac{1}{m+4}}\,, \tag{8}$$

where C is an appropriate weighting coefficient.

From the maximum likelihood principle and assuming independence of the samples, one can write the likelihood of drawing the complete dataset as the product of the densities of one sample

$$\mathcal{L}(h)=\prod_{j=1}^{N}p_N(\mathbf{x}_j,h)=\prod_{j=1}^{N}\frac{1}{N}\sum_{i=1}^{N}\frac{1}{\left(h\sqrt{2\pi}\right)^m}\exp\left(-\frac{\|\mathbf{x}_j-\mathbf{x}_i\|^2}{2h^2}\right). \tag{9}$$

As this likelihood function has a global maximum for h=0, in [8] a modified approach has been proposed

$$\mathcal{L}^*(h)=\left[\prod_{j=1}^{N}\frac{1}{N}\sum_{i=1,\,i\neq j}^{N}\frac{1}{\left(h\sqrt{2\pi}\right)^m}\exp\left(-\frac{\|\mathbf{x}_j-\mathbf{x}_i\|^2}{2h^2}\right)\right]^{\frac{1}{m}}. \tag{10}$$

This function has one maximum for h, which can be found by setting to 0 the derivative of the logarithm of $\mathcal{L}^*(h)$ with respect to h, which gives

$$\frac{1}{N}\sum_{j=1}^{N}\frac{\sum\limits_{i\neq j}^{N}\frac{\|\mathbf{x}_j-\mathbf{x}_i\|^2}{h^3}\exp\left(-\frac{\|\mathbf{x}_j-\mathbf{x}_i\|^2}{2h^2}\right)}{\sum\limits_{i\neq j}^{N}\exp\left(-\frac{\|\mathbf{x}_j-\mathbf{x}_i\|^2}{2h^2}\right)}=\frac{m}{h}\,. \tag{11}$$

A crude but rather fast way to obtain an approximate solution of (11) is by assuming that the density estimate of Eq. (5) on a certain location $\mathbf{x}$ in the feature space is determined by the nearest kernel only, [8]. In this case

$$\frac{\partial\log(\mathcal{L}^*(h))}{\partial h}=\frac{1}{N}\sum\nolimits_{j=1}^{n}\frac{\|\tilde{\mathbf{x}}_j-\mathbf{x}_i\|^2}{h^3}=\frac{m}{h}\,. \tag{12}$$

In this paper we use the optimal h derived from (12) defined as

$$h_2^*=C\left((mN)^{-1}\sum\nolimits_{j=1}^{N}\|\tilde{\mathbf{x}}_j-\mathbf{x}_j\|^2\right)^{\frac{1}{2}}\,, \tag{13}$$

where $\tilde{\mathbf{x}}_i$ represents the nearest neighbor of the sample $\mathbf{x}_i$, and C is a tuning parameter.

3 Proposed Algorithm

Let us assume a filtering window W containing N image pixels, $\{x_1, \ldots, x_N\}$ and let us define the similarity function $\mu : [0; \infty) \to \mathbb{R}$ which is non-ascending and convex in $[0; \infty)$ and satisfies $\mu(0) = 1$, $\mu(\infty) = 0$. The similarity between two pixels of the same intensity should be 1, and the similarity between pixels with minimal and maximal gray scale values should be very close to 0. The function $\mu(x_i, x_j)$ defined as $\mu(x_i, x_j) = \exp\{-[(x_i - x_j)/h]^2\}$, where h is the bandwidth of the Gaussian kernel, defined by (8) or (13), satisfies the required conditions.

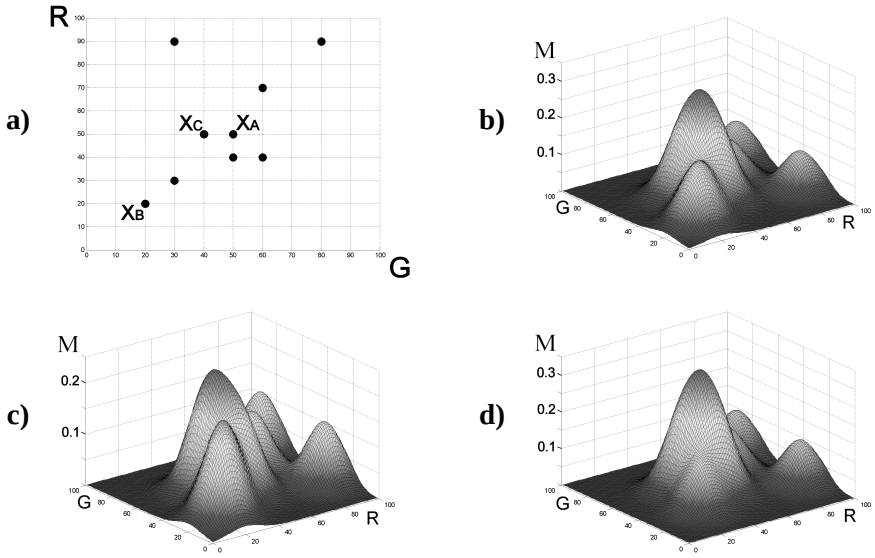


Figure 1: Impulsive noise removal technique in the 2D case. Fig. **a**) depicts the arrangement of pixels in W and Fig. **b**) their nonparametric probability density estimation. Figs. **c**) and **d**) present the density plots for the cases when the central pixels x_A and x_B are removed from W. It can be seen that in the first case **c**) the pixel $x_1 = x_A$ will be retained and in the second case **d**) the pixel $x_1 = x_B$ will be replaced by x_A. The pixel x_A will be preserved, as in Fig. **c**) the plot attains its maximum at x_C, but this maximum is less than the maximum for x_A in Fig. **b**). Regarding sample x_B, its rejection causes that the maximum is attained at x_A and this pixel will replace the central pixel x_B.

Let us additionally define the cumulated sum M of similarities between a given pixel and all other pixels belonging to window W. For the central pixel x_1 we introduce M_1 and for the neighbors of x_1 we define M_k as

$$M_1 = \sum_{j=2}^{N} \mu(x_1, x_j), \quad M_k = \sum_{j=2,\ j\neq k}^{N} \mu(x_k, x_j),\ k > 1, \tag{14}$$

which means that for x_k, which are neighbors of x_1, we do not take into account the similarity between x_k and x_1, which is the main idea of this algorithm. The omission of the similarity $\mu(x_k, x_1)$ when calculating M_k, privileges the central pixel, as in the calculation of M_1 we have $N - 1$ similarities $\mu(x_1, x_k)$, $k > 2$ and for M_k, $k > 1$ we have only $N - 2$ similarity values, as the central pixel x_1 is excluded from the calculation of M_k, [9, 10], (see Figs. 1, 3).

In the construction of the new filter, the reference pixel x_1 in the window W is replaced by one of its neighbors if $M_1 < M_k,\ k = 2, \ldots, N$. If this is the case, then x_1 is replaced by that x_{k^*} for which $k^* = \arg\max M_k,\ k = 2, \ldots, N$. In other words x_1 is detected as being corrupted if $M_1 < M_k,\ k = 2, \ldots, N$ and is replaced by its neighbors x_k which maximizes the sum of similarities M between all the pixels from W excluding the central pixel.

The basic assumption is that a new pixel must be taken from the window W, (introducing pixels, that do not occur in the image is prohibited like in the VMF). For this purpose μ must be convex, which

means that in order to find a maximum of the sum of similarity functions M it is sufficient to calculate the values of M only in points $x_1, x_2, \ldots, x_N$.

The working scheme of the new filter is presented in Fig. 3 for the gray scale case and in Fig. 1 for the two-dimensional data. In the example provided by Fig. 3, the supporting window W contains 9 pixels of intensities $\{15, 24, 33, 41, 45, 55, 72, 90, 95\}$, (their special arrangement in W is not relevant). Each of the graphs from **a)** to **i)** shows the dependence of M_1 and $Mi_{/1}$ on the gray scale value, ($M_{/1} < M_1$), where $Mi_{/1}$ denotes the cumulative similarity value with rejected central pixel x_1, on the sample's intensity. Graph **a)** shows the plot of M_1 and $M_{/1}$ for $x_1 = 15$, plot **b)** for $x_1 = 24$ and so on till plot plot **i)**, which shows the graphs of M_1 and $M_{/1}$ for $x_1 = 95$. The central pixel will be replaced in cases: **(a)**, **(b)**, **(f)** - **(i)**, as in those cases there exists a pixel x_k for which $M_1 < M_k$. The continuous plots show that the extremum of the similarity function $M_{/1}$ is always obtained at points $x_k \in W$, which is an important feature of this algorithm. Because the function $M_{/1}$ is convex, the maximum can be found by calculating the similarity values in N points only, which makes the algorithm relatively fast.

The presented approach can be applied in a straightforward way to multichannel images using the similarity function defined as $\mu(\mathbf{x}_i, \mathbf{x}_j) = \exp\{-[\|\mathbf{x}_i - \mathbf{x}_j\|/h]^2\}$, where $\|\cdot\|$ denotes the specific vector norm and h denotes the bandwidth. Now in exactly the same way we can maximize the total similarity function M for the vector case.

4 Results

The performance of the proposed impulsive noise reduction filters was evaluated using the widely used PSNR quality measure. Figure 2a) shows the dependence of the noise attenuation capability of the proposed filter class on the bandwidth type h_1^* and h_2^* defined by (8) and (13). Clearly the filter based on the h_2^* outperforms the technique based on the h_1 bandwidth for the whole range of used contamination probabilities p, ($p = 0.01$ - 0.1).

Figure 2b) presents the dependence of the PSNR restoration quality measure on the kind of the Minkowski norm. Surprisingly, the L_∞ norm yields significantly better results than the L_1 or L_2 norms. This is the result of the construction of the h_2^* bandwidth, which depends on the nearest neighbor in the sliding filter window. This behavior is advantageous, as the calculation of the L_∞ norm is much faster than the evaluation of distances determined by L_1, L_2 norms.

The efficiency of the filters based on adaptive h_1^* and h_2^* bandwidths are dependent, (especially for very small noise contamination) on the coefficient C in (8) and (13). Figure 2c) shows the dependence of PSNR for the filter based on h_2^* as a function of C in (13). For low noise intensity the parameter C should be significantly larger than for the case of images corrupted by heavy noise process. However, setting C to 4 is an acceptable trade-off, as can be seen in Fig. 2 d), which depicts the efficiency of the proposed filter in comparison with VMF, AMF and BVDF. It can be observed that although the $C = 4$ is not an optimal setting for the whole range of tested noise intensities, nevertheless the described filter yields much better results than the traditional techniques.

This is also testified by Fig. 4, which compares the filtering results obtained by the filter based on adaptive h_2^* bandwidth with the performance of the *reference* VMF, BVDF, DDF filter. As can be observed the new filtering has much better detail preserving properties than VMF, BVDF and DDF.

5 Conclusions

In this paper a new nonparametric technique of impulsive noise removal in multichannel images has been proposed. The described filter class is based on the estimation of the kernel bandwidth using the technique proposed in [8]. The experiments revealed, that the proposed algorithm yields the best results when applying the L_∞ norm, which makes the filter computationally very attractive. The obtained results show that the proposed technique excels significantly over the standard techniques like VMF, BVDF and DDF. The future work will focus on the automatic adjustment of the tuning parameter C in (8) and (13).

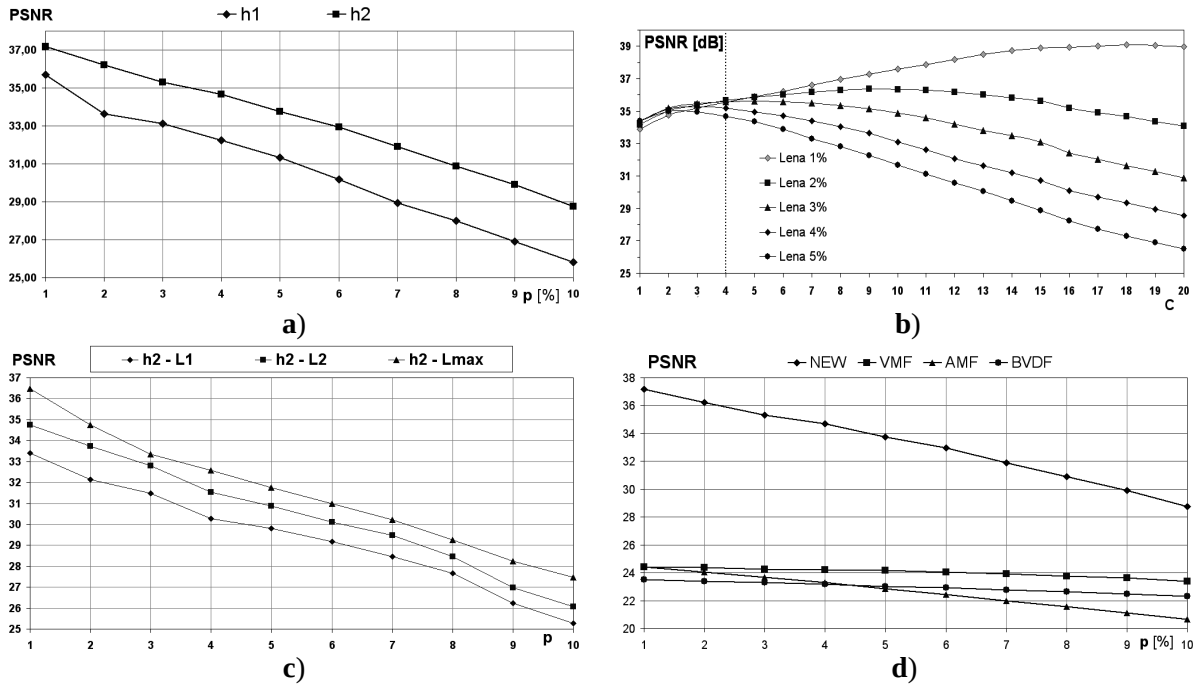


Figure 2: Dependence of the efficiency of the proposed filtering scheme on h_1^* (8) and h_2^* (13) - (**a**), besides the dependence of the PSNR on the tuning parameter C in (13) - (**b**) and the dependence on the kind of Minkowski norm for the bandwidth h_2* - (**c**). Figure (**d**) shows the comparison of results obtained using the h_2^* bandwidth, L_∞ norm and $C = 4$ with the standard multichannel filters VMF and BVDF, (test were performed on the color image *LENA*); p denotes the probability of a pixel corruption - to RGB channels random, uniformly distributed values from the interval [0,255] were assigned.

References

[1] I. Pitas, P. Tsakalides, Multivariate ordering in color image processing, IEEE Trans. on Circuits and Systems for Video Technology, 1, 3, 247-256, 1991.

[2] K. Tang, J. Astola, Y. Neuovo, Nonlinear multivariate image filtering techniques, IEEE Trans. on Image Processing, 4, 6, 788-797, 1995.

[3] J. Astola, P. Haavisto, Y. Neuvo, Vector median filters, Proceedings of the IEEE, 78, 678-689, 1990.

[4] P.E. Trahanias, A.N. Venetsanopoulos, Vector directional filters: a new class of multichannel image processing filters, IEEE Trans. on Image Processing, 2, 4, 528-534, 1993.

[5] K.N. Plataniotis, A.N. Venetsanopoulos, "Color Image Processing and Applications", Springer Verlag, August 2000.

[6] B.W. Silverman, "Density Estimation for Statistics and Data Analysis", London, Chapman and Hall, 1986.

[7] D.W. Scott, "Multivariate Density Estimation", New York, John Wiley, 1992.

[8] M.A. Kraaijveld, A Parzen classifier with an improved robustness against deviations between training and test data, Pattern Recognition Letters, 17, 679-689, 1996.

[9] B. Smolka, K.N. Plataniotis, A. Chydzinski, M. Szczepanski, A.N. Venetsanopulos, K. Wojciechowski, Self-adaptive algorithm of impulsive noise reduction in color images, Pattern Recognition, 35, 1771-1784, 2002.

[10] B. Smolka, R. Lukac, A. Chydzinski, K.N. Plataniotis, K. Wojciechowski, Fast adaptive similarity based impulsive noise reduction filter, Real Time Imaging, 9, 261-276, 2003.

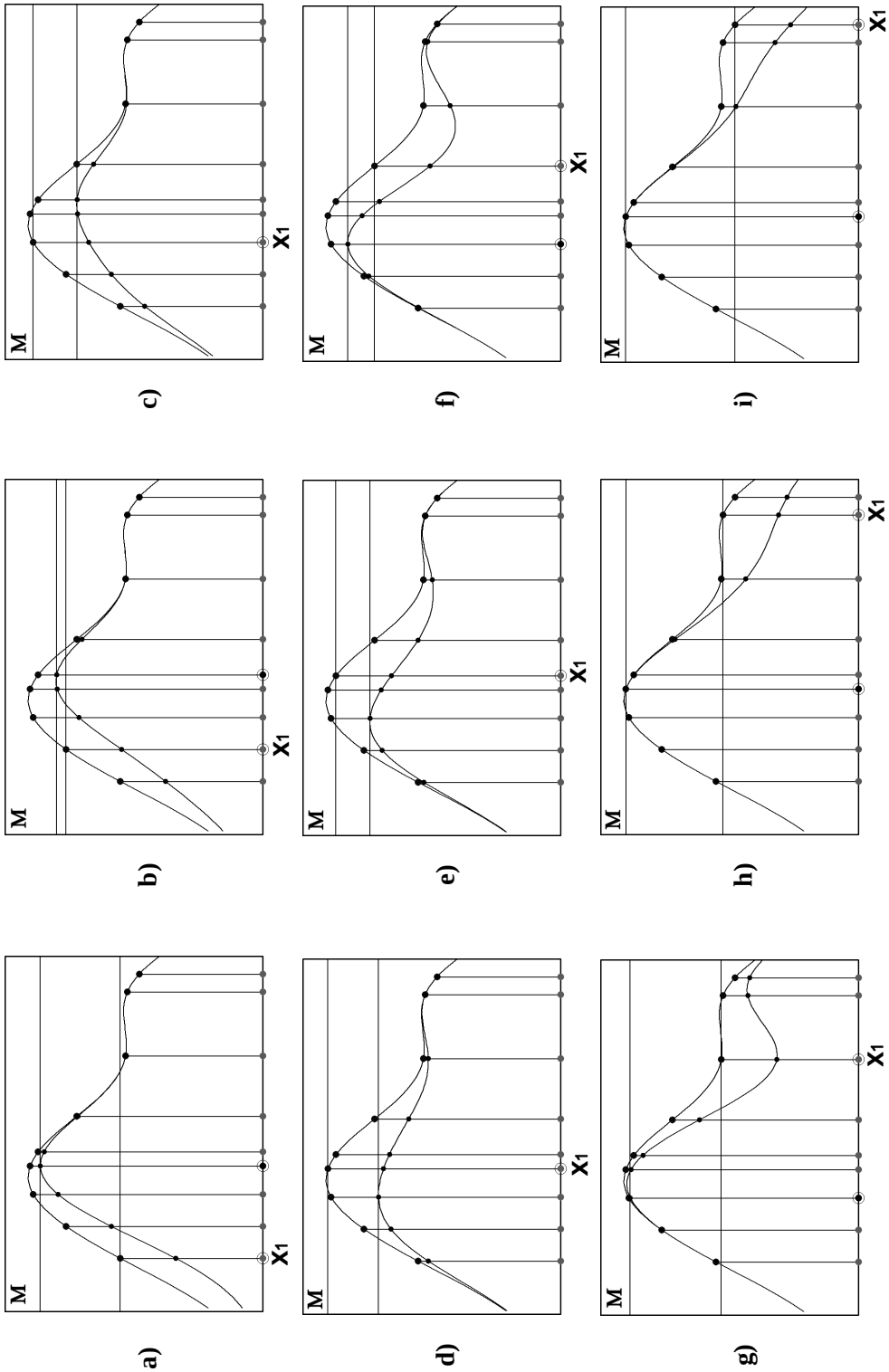


Figure 3: Illustration of the new filter construction. The supporting window W of size 3×3 contains 9 pixels of intensities $\{15, 24, 33, 41, 45, 55, 72, 90, 95\}$. Each of the graphs from **a)** to **i)** shows the dependence of M_1 and $M_{/1}$, ($M_{/1} < M_1$), where $M_{/1}$ denotes the cumulative similarity value with rejected central pixel x_1 on the gray scale value. Graph **a)** shows the plot of M_1 and $M_{/1}$ for $x_1 = 15$, plot **b)** for $x_1 = 24$ and so on till plot plot **i)** shows the graphs of M_1 and $M_{/1}$ for $x_1 = 95$. The arrangement of pixels surrounding the central pixel x_1 is not relevant. The central pixel will be replaced in cases: **(a)**, **(b)**, **(f - i)**, as in those cases there exists a pixel x_k for which $M_1 < M_k$ or $R_1 > R_k$ is satisfied

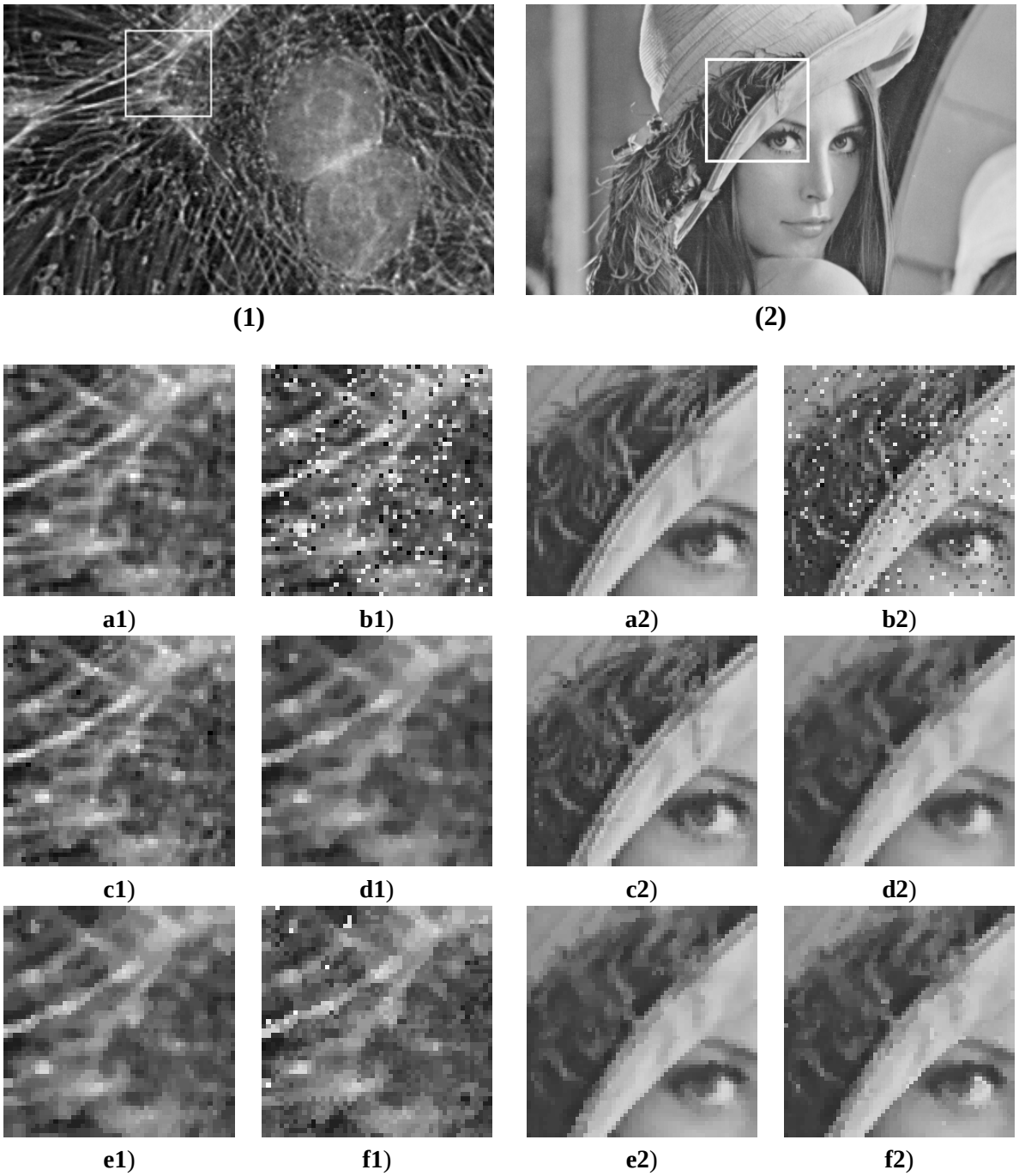


Figure 4: Illustrative example of the efficiency of the proposed algorithm: **a**) zoomed parts of the test color images, **b**) image corrupted by 3% of impulsive noise, **c**) image after filtering with the proposed filter, **d**) VMF output, **e**) DDF output, **f**) BVDF output.

OBJECT TRACKING IN LOW FRAME-RATE VIDEO

Alfred Levy
University of Central Florida
School of Computer Science
4000 Central Florida Blvd
Orlando, FL 32816
USA
email: alfredk@cs.ucf.edu

Dr. Niels Da Vitoria Lobo
University of Central Florida
School of Computer Science
4000 Central Florida Blvd
Orlando, FL 32816
USA
email: niels@cs.ucf.edu

Dr. Mubarak Shah
University of Central Florida
School of Computer Science
4000 Central Florida Blvd
Orlando, FL 32816
USA
email: shah@cs.ucf.edu

Abstract

Tracking moving objects is a basic tool which allows the development of solutions to complex problems such as target acquisition, automatic surveillance, action recognition, etc. Tracking algorithms generally deal with video that is from 15 to 30 frames per second, and the objects in motion do not exhibit huge jumps. However, if the video frame rate is low, or more precisely, if the objects in motion move large distances from frame to frame, current tracking methods will perform very poorly.

We propose a method of tracking that allows for large spatial discontinuities in object motion and is still able to track successfully. The feasibility of tracking in these sequences is demonstrated, and results are given from application of the proposed method to video sequences taken at 2 frames per second.
Keywords*: Uncooperative Video, Tracking*

1 Introduction

The problem of tracking moving objects in a video sequence is a well known and well researched problem in Computer Vision. There are many tracking algorithms such as Mean Shift [3], Multiple Hypothesis Trackers [5], Bayesian methods [7] [6], even Monte Carlo methods [4]. However, most tracking problems and solutions deal with video data that has relatively good frame rates, i.e. from 15 to 30 fps, and the objects in motion do not exhibit huge jumps. However, if the video frame rate is low or, more precisely, the objects in motion move so much from frame to frame that their new positions do not overlap their previous positions, current tracking methods will perform very poorly.

Part of the problem of tracking in these conditions is that we do not immediately know what we are tracking, the number of objects, or their boundaries. Simple application of connected components to foreground blobs will not be correct since a group of objects will be mistaken for one object, and when the group splits up the tracker will become confused and lose them. This confusion arises because there is no longer any area of the video frame which looks enough like the group before it split. However, subsections of that group (the individual people) still exist. Since the group still exists as separate components, it is reasonable to expect that if you subdivide the original group correctly, and then look for the subsections, your search will be successful.

The proposed tracking method is to break all foreground area into pieces small enough so that they will most likely only belong to one object. If the motion of these small pieces can be accurately determined, we can then reconstruct objects from the pieces.

2 Background Subtraction

Background subtraction is used to limit processing to areas that are likely to have moving objects. The background subtraction used in our method determines foreground areas with only knowledge of the previous, current, and next frames. We have tried more complicated background subtraction, such as the multiple gaussian method in [1], but they typically require too much "warm up" time to be practical in low frame rate sequences. We decided on a simple substitute method which works acceptably in sequences with significant frame to frame motion.

The algorithm calculates a difference picture between the previous and current frames, then again between the current and next frames. To lessen the effects of lighting and shadow, the differences are calculated as the Euclidean distance between red and blue crominance "points" taken from the YC_rC_b colors of the pixels in question. It thresholds these differences to produce binary images, then uses connected components to remove small noise, then dilates and erodes to close up small holes in foreground areas. Finally, silhouettes of moving objects are obtained by calculating an intersection of the two inter-frame differences.

To understand why this works, consider that the difference between two frames will produce an image of regions that changed between those two frames. Thresholding this difference so that the result is a binary difference image produces, in effect, a union of moving areas. If we have the motion areas of 3 consecutive frames A, B, and C, we can use frame differencing to find their unions: $A \cup B$ and $B \cup C$. Then we can recover B quite easily by the equation: $B = (A \cup B) \cap (B \cup C),$ if$A \cap C = \emptyset$. Proof not shown for brevity. The assumption that $A \cap C = \emptyset$ simplifies calculations, and works best when the moving objects are moving quickly enough that they do not not overlap much of the same area from one frame to the next. In low frame rate video sequences this is generally the case.

3 Patches

Tracking of objects is similar to object recognition in that we are trying to recognize an object from frame N in frame $N + 1$. This is especially true in low frame-rate video sequences since it is not assumed that objects will be found close to their original positions. [2] gives a method for object recognition which uses small squares of pixels sampled from interest points on an object. Our method for tracking uses a similar concept for "recognizing" objects from frame to frame, although, instead of selection by interest operators, all foreground areas are covered by small regions which are then searched for in the next frame.

These small regions, called patches, are the basic unit of tracking in this method. The creation of patches is illustrated in the left half of figure 1 As you can see, the foreground region is diced into small square patches. Notice that not all squares are fully occupied by foreground pixels. We will only be interested in correlating areas that are foreground so only the pixels in the square that are part of the foreground will take part in the correlation step.

If a patch has a few pixels that will not be used for correlation, it will still contribute valuable information. However, we don't want to have a patch that consists of only 3 valid pixels. This would not add any meaningful information, and would only waste memory. In order to strike a balance, patches are required to have at least half of their pixels be valid foreground pixels. This constraint also carries over to the correlation process.

After patches are created, they will be correlated with every location in the next frame of the video. This is, obviously, a time consuming procedure. The cost of performing all these calculations is ameliorated by placing two constraints on patch motion. First, a patch cannot move more than half of the image diagonal length. Second, a patch can only move into foreground areas. Every location in the video frame that is within the maximum allowable motion distance of a patch's present location is considered to see if it is in the foreground. Like the patch creation step, it is not required that the entire location square be composed of foreground pixels. If the overlap between the mask of the patch and the foreground region

Figure 1: Creation and correlation of patches: Left, generating patches from foreground area. Right, patch from one frame correlated with next. Foreground overlap highlighted

of the location is more than half the pixels in the patch, the location can be considered as a valid match. See the right half of figure 1.

The correlation function used to calculate matches is a modified sum of squared differences function. The squared difference we are talking about will be the square of the magnitude, or Euclidean norm, of the vector difference between two RGB pixels. That is: $\|\vec{x}-\vec{y}\|^2 = (x_r-y_r)^2+(x_g-y_g)^2+(x_b-y_b)^2$. Correlation will calculate the mean pixel squared difference, and then normalize it to be between 0 and 1. Thus: $\rho = 1 - \frac{\sum_{(i,j)\in Q} \frac{\|P(i,j)-I(x+i,y+j)\|^2}{|Q|}}{3\times 255^2}$ Range $= [0,1]$. P and I are the color pixels of the patch and image, respectively. (x,y) is the top left corner of the rectangular location with which the patch is being compared. Q is all (i,j) such that the patch's pixel (i,j) is a foreground pixel, and the image pixel $(x+i, y+j)$ is also a foreground pixel.

In a normal correlation based search, the patch would be moved to the location in the next frame with which it had the highest correlation. However, for low frame-rate sequences, it is beneficial to take other factors into consideration. Instead of choosing the location with the best correlation and moving the patch there, all correlation results are saved for later when the final decision will be made.

4 Objects

Once patches have had their correlation coefficients calculated with all valid foreground regions of the next frame of the input video, they can be grouped into objects. Each "Object" structure in the tracking program should correspond to real world moving objects like people, cars, clouds, etc. This being the case, it makes sense that the grouping criteria would match the characteristics of a real world object. Two main assumptions are used to group patches: patches belonging to one object should move in roughly the same direction, patches belonging to one object should be relatively close to each other.

As was stated previously, patches are not automatically moved to the position with which they have the highest correlation. Thus, we cannot emphatically determine the position, or motion vector of any patch. How then shall these patches be grouped into objects? If, hypothetically, each patch was placed in all locations with which that patch had an acceptable correlation, we could then determine all groups of which that patch could be a part. Then the best of these hypothetical groups could be chosen. To put it more directly, make all possible groups, choose the best ones, throw out the rest.

Each hypothetical group will be made so that all patches meet the two criteria of spatial closeness and motion similarity. The worth of a group is then decided based on how many patches that group has since more patches mean more total pixels and a more accurate correlation. This is refined to consider the fitness of match determined by the correlation process for each patch. If a patch could be placed in several groups, it makes sense to place the patch where it has the highest correlation. Or, from the viewpoint of the group, Several patches with high correlations is better than more patches with mediocre correlations. The worth of a group will still be measured by how many patches it has, but weighted by the patches' correlation with their locations.

The first step in object creation is to make groups of patches that move with about the same motion vector. We will call these "Iso-Kinetic Groups." Making iso-kinetic groups would be easy if we knew where each patch was going, but since we only have the possible locations of each patch finding the optimal solution would take exponential time. Fortunately a simple greedy algorithm provides an alternative. First, let U be the set of all ungrouped patches. Second, find all $I_j = \{(p_i, \vec{v_i}) : p_i \in U$ and $\|\vec{v_i} - \vec{v_j}\| \leq \varepsilon\}$. Where v_j is the main vector of I_j and ε is the maximum acceptable deviance from that vector. Third, determine I_k such that $|I_k| \geq |I_j| \ \forall I_j$. Fourth, make I_k into an iso-kinetic group and repeat the entire process until $|U| = 0$.

The bulk of the work in this algorithm lies in step 2. Finding all possible iso-kinetic groups is a very time consuming process, and also unnecessary. The goal is to find groups of *patches*, so the algorithm should only check vectors a patch can accept. Thus, $\forall p_i \in U$ and $\forall \vec{v_i}$ acceptable to p_i we generate a group I_j of all patches p_i for which $\exists \vec{v_i}$ such that $\|\vec{v_i} - \vec{v_j}\| \leq \varepsilon$. Furthermore, if there is more than one acceptable $\vec{v_i}$, the algorithm tries to find the vector $\vec{v_i}$ which is closest to $\vec{v_j}$. The threshold ε is chosen to specify how homogenous the motion of patches in a group should be. Small ε results in groups with little deviance in the motion vectors, but there may not be a vector upon which patches can agree.

Patch overlap is not allowed. If patches are allowed to overlap, they tend to increase in overlap as tracking continues. Inevitably the patches overlap completely and become exact duplicates of each other. This is obviously bad, so the second step in object creation is to rearrange the patches of each iso-kinetic group to not overlap, but still move in roughly the same direction. That is, $\vec{v_i}$ from $(p_i, \vec{v_i})$ is changed slightly to $\vec{v_i'}$ such that $\|\vec{v_i'} - \vec{v_j}\| \leq \varepsilon$ and $\vec{v_i'}$ is acceptable to p_i, but, unlike $\vec{v_i}$, When the patch moves with vector $\vec{v_i'}$ it will not overlap another patch that has a better correlation coefficient. If no acceptable $\vec{v_i'}$ exists, the patch p_i is deleted. The process of moving weak patches out of the way of strong patches is repeated until all overlaps are resolved.

The final step of the grouping process to apply a connected components process where connectedness is defined as the distance between two patches being less than a proximity threshold. Thus, we call this "Proximate Components." Each iso-kinetic group I_j is passed through the proximate components algorithm. A threshold is set that requires a component have a minimum number of patches to become an object. Otherwise, its patches are deleted. The resulting objects satisfy the two criteria of spatial closeness and approximate homogeneity of motion.

One happy consequence of using motion as a grouping criterion is that the same grouping process responsible for object creation can be used just as well for object tracking with a few additional constraints. Firstly, most, if not all, of the patches in the object will be in the same iso-kinetic group. The object tracker will apply the same process used in object creation to the patches that belong to the object. It will take the largest of these new "sub-objects" and discard the rest. The object is moved in the direction these patches took. The patches of the discarded objects are returned to the list of free patches so they can possibly be made into new objects. This handles the case that a group of real world objects are traveling close together and then split up.

The second additional constraint is more of a suggestion than a hard rule. In low frame-rate video, objects do not necessarily have smooth velocity, or acceleration. However, observation of many sequences shows that it is not uncommon for an object to move in a relatively straight path. Consequently, the object tracker will suggest that an object should continue following the same motion vector as it took in the previous frame. In the iso-kinetic grouping, it will choose the group I_j whose main vector v_j is the same as the object's previous motion. If $|I_j|$ has half or more of the patches of the object, it will determine the object's motion. Otherwise, the largest iso-kinetic group is chosen as in the previous paragraph.

After the existing patches of an object have been tracked, newly created patches are added to the object. The object needs to be able to gain patches in case previously occluded parts of the object come into view. Patches that meet the two grouping criteria (spatial closeness and motion similarity) will be added to each object on a first come first served basis. If a patch is close enough to more than one object, and it can accept motion vectors that are within ε of the main vectors of those objects, the first object to claim that patch gets it. The patches in each object are rearranged to fix patch overlaps just like in the

reorganization algorithm given above. Any patches that are discarded in the rearrangement are returned to the free patches list for later use. This is important since the object tracking phase actually comes before object creation so that existing objects are able to claim new patches that are rightfully theirs before new objects are made from them.

5 Occlusion Handling

Any useful tracking algorithm must be able to handle occlusions to some degree. In our method, we start to handle occlusion when the bounding rectangles of two objects overlap by more than 80%. Meaning, the overlapping area is 80% of the smaller rectangle. It is not uncommon for bounding rectangles to overlap by a large amount, even when the actual real world objects do not overlap much, because patches don't always fill up the entire rectangular bounds of the object. 80% was found to be a good cut-off point. If 80% of the smaller of the bounding rectangles is occupied by the other, it is quite likely that the real objects are occluding each other. We do not check for overlap by finding if the objects' patches overlap because if the actual pieces of the objects represented by the patches were to overlap, the result would be an area that was not similar to any one patch and unlikely to be chosen during the correlation search, so they wouldn't overlap anyway.

After determining which objects are overlapped, we decide which object is in front. Obviously we cannot update the patches of objects that are underneath other objects. Those patches would no longer represent their object. However, the front-most object can, and should, be updated. The front object is determined by applying a simple observation. If a real world object is in front of another, most of the image pixels in the occlusion area will be a part of the front object. If the front object looks sufficiently different from the rear object(s), then it is reasonable to assume that patches of the front object will have the best correlation coefficients with their positions in the image. Thus, the front object is the object with the highest sum of patch correlation coefficients.

Motion prediction is very basic since the objects only exist for a short time and it is hard to build up a statistically accurate motion model. The front-most object can be tracked, so it is updated with accurate positions. The velocity of a rear object is assumed to remain constant while it is occluded. Its position is updated every frame, based on the constant velocity assumption, until it is freed from occlusion. So the complete occlusion handling mechanism is: First, find objects whose bounding rectangles overlap more than 80%. Second, determine which object is the front-most object by finding the object with the best total correlation. Then, objects behind the front object are "put on hold" until they no longer occlude, and they are tracked using simple constant velocity prediction.

6 Results and Conclusion

These sequences can be downloaded from http://www.cs.ucf.edu/ vision/lowframeratetracking/downloads.html along with other sequences and results not shown here.

Figure 2: Good results tracking 3 people from overhead view

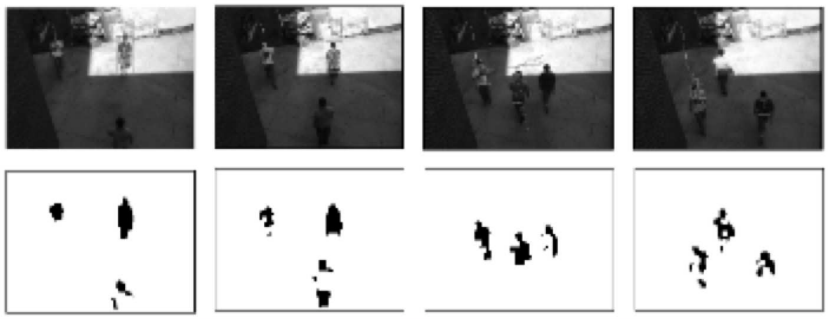


Figure 3: Objects lost. Failure in background subtraction due to lighting

Figure 4: A simple occlusion is handled well

Tracking objects in low frame-rate video is, unfortunately, not studied or used in the field of Computer Vision. Unfortunate because, as we have shown, it can be done even with primitive background subtraction and practically non-existent motion prediction models. Objects can be found and tracked by partitioning the set of patches into iso-kinetic groups, then determining proximate components. This method could be much more effective if a robust background subtraction method, tuned for low frame-rate sequences, were employed to give more accurate foreground/background segmentation. As it stands, the concept of low frame-rate tracking has been demonstrated as a feasible concept, and implemented with good results.

References

[1] C. Stauffer and W.E.L. Grimson, "Adaptive background mixture models for real-time tracking," *IEEE CS Conf. Computer Vision and Pattern Recognition,* (CVPR 99), vol. 2, nos. 23-25, June 1999, pp. 252

[2] D. Jugessur and G. Dudek, "Local appearance for robust object recognition" *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* (CVPR 2000), vol. 1, nos. 13-15, June 2000, pp. 834-839.

[3] D. Comaniciu and V. Ramesh, "Mean shift and optimal prediction for efficient object tracking," *Proc. Int'l Conf. Image Processing,* (ICIP 2000), IEEE CS Press, vol. 3, nos. 10-13, Sept. 2000, pp. 70-73.

[4] E. Poon and D.J. Fleet "Hybrid Monte Carlo filtering: edge-based people tracking," *Proc. Workshop Motion and Video Computing,* nos. 5-6, Dec. 2002, pp. 151-158

[5] I.J. Cox and S.L. Hingorani "An efficient implementation of Reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 18, no. 2, Feb. 1996, pp. 138-150

[6] M.G.S. Bruno, and J.M.F. Moura "Multiframe Bayesian tracking of cluttered targets with random motion," *Proc. Int'l Conf. Image Processing,* vol. 3, nos. 10-13, Sept. 2000, pp. 90-93

[7] M.J. Black and H. Sidenbladh "Learning image statistics for Bayesian tracking," *Proc. 8th IEEE Int'l Conf. Computer Vision,* (ICCV 2001), vol. 2, nos. 7-14, July 2001, pp. 709-716

SEMI-AUTOMATIC IDENTIFICATION OF HUMPBACK WHALES

Elena Ranguelova Mark Huiskes Eric Pauwels
Center for Mathematics and
Computer Science (CWI)
Amsterdam, The Netherlands
Elena.Ranguelova@cwi.nl Mark.Huiskes@cwi.nl Eric.Pauwels@cwi.nl

Abstract

This paper describes current work on a photo-id system for humpback whales. Individuals of this species can be uniquely identified by the light and dark pigmentation patches on their tails. We propose semi-automatic algorithm based on marker-controlled watershed transformation for segmenting the animal's tail from the surrounding sea. We propose fitting an affine invariant coordinate grid to the resulting segmentation. The grid can be adjusted according to the level of occlusion by the sea. A numerical feature vector capturing the patch-distribution with respect to the grid is then automatically extracted and used to match the individual against the database of similarly processed images.
Keywords*: photo-identification, biodiversity, watershed segmentation, affine-invariant coordinate grid, similarity matching*

1 Introduction

Individual identification of cetaceans (marine mammals, i.e. whales, dolphins and porpoises) is of great interest to marine biologists. Identification plays an important role in their long-term studies of the population and behavioral patterns of the mammals [1, 4, 6]. The method of photo-identification hinges on the uniqueness of the natural markings which can be captured by photographing the dorsal fins or flukes (i.e. tail). Marine biologists discovered more than 30 years ago that humpback whales exhibit sufficient variation in their natural markings to allow the identification of individuals based on images of their flukes. As the photographic collections grew, so did the need for more efficient retrieval methods that would allow a researcher to quickly match new photographs against the image database.

There are several approaches to photo-identification available in the literature. In [6] manually generated code is used, based on a set of 38 generic fluke patterns which takes into account the shape of the central notch and the location of blotches/scars. Similarly, WhaleNet [8] is a graphical user interface (GUI) which allows the user to narrow down the search for matches by visually selecting one of 18 fluke types. Araabi extends a curve-matching technique, originally developed for the identification of bottlenose dolphins [2], for the encoding of the fluke's trailing edge of humpback whales [1].

The approach proposed in this paper comprises two main steps, namely the extraction of the fluke region and patches, and the actual matching. While Kehtarnavaz et al. [4] introduced an interactive live-wire algorithm for the fluke extraction, we favour the use of morphological segmentation tools. And while [4] introduces affine moment invariants as features, we avoid the use of high order integrals by constructing an affine invariant grid that is automatically fitted to the tail. To decrease the effect of different levels of occlusion (submersion) by the sea the grid can be dynamically adjusted according to the level of occlusion of another tail- candidate for matching. Next, each region is characterized by the relative contribution of dark and light patches. This maps the visual information into a numerical feature vector which can then be compared to the feature vectors obtained from other images.

2 Fluke Patches Extraction

Because the photographic material is typically quite challenging (small colour differences between animal and background, confounding factors such as water splash, highlights on wet surfaces, etc.) automatic segmentation of the tail is unable to deliver the accuracy required for photo-identification. For that reason, we have opted for semi-automatic segmentation based on a *marker - controlled watershed algorithm* [3].

The watershed transformation is a powerful and well - established mathematical morphology tool for image segmentation which has been used in many applications [3]. Any gray-level image can be considered as a topographical surface. Flooding this surface from its minima while preventing the merging of water coming from difference sources, will result in a partitioning of the image into *catchment basins* associated with each minimum. If we apply this transformation to the gradient of an image, we should obtain catchment basins corresponding to homogeneous gray-level regions. The transform, however, tends to produce an over-segmentation due to the local variations in the gradient. A *marker-controlled* transformation is a solution to this problem. The location and support of the minima is given *a priori* in the form of markers, after which the gradient image is modified via morphological reconstruction [3]. In this way only the most significant gradient edges in the areas of interest between the markers appear in the final segmentation.

The tail extraction process is initialized by the user, who specifies a rough initial contour (marker) within the tail. This is illustrated in Fig. 1.

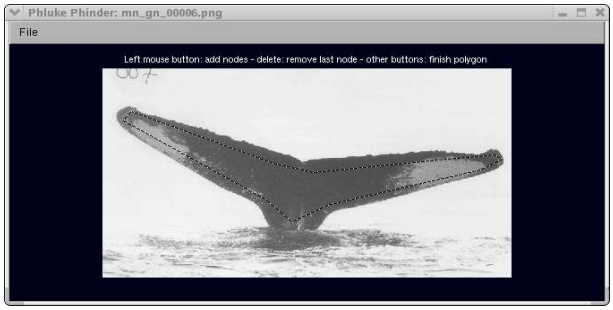


Figure 1: Original image and initial rough marker for the tail.

The watershed transformation is then applied to the modified gradient and automatically produces an estimated boundary contour for the fluke. Whenever needed, the program interface allows the user to fine-tune the result by interactively introducing set of additional *positive* and *negative* markers. The noise- and error-prone region at the basis of the tail (due to wave occlusion, water splash, etc.) is removed by clipping the contour (Fig. 2).

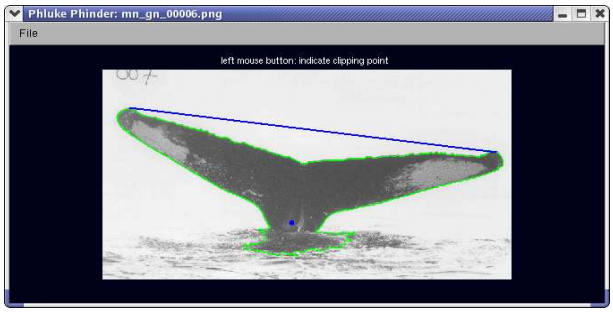


Figure 2: Watershed-based segmentation of fluke (green contour). The fluke is clipped at its base at a user-supplied point (blue) by fitting a line parallel to the blue line connecting the fluke tips.

The user is prompted to specify three tail *landmarks*, viz. the left and right flukes tips and the central fluke notch. These landmarks have also been used for photo-identification of flukes in [5]. Next, we use Otsu's gray-level thresholding [7] on the extracted fluke to obtain an initial segmentation into dark and light patches.

Finally, the interface supports local thresholding in order to allow the user to fine-tune this patch segmentation in regions of special interest where the global thresholding failed to catch subtle, but significant details (Fig.3).

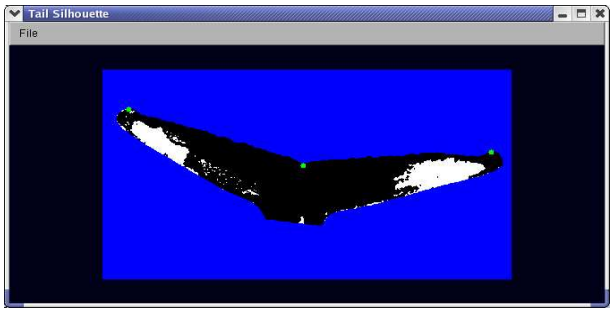


Figure 3: Final segmentation result divided in black and white patches used for the identification. The three landmark points are indicated as green dots.

3 Matching

3.1 Fitting a Coordinate Grid

Images typically exhibit a large variation in viewing angles, distances and fluke inclination. In [4] it is argued that since fluke surfaces are nearly planar with dimensions significantly smaller than the distance to the camera, these variations can be modelled using affine transformations such as rotation, translation and scaling. To be robust with respect to the above-mentioned variability, we therefore propose a coordinate grid that is superimposed on the tail and will divide it into N_R regions. The idea is very simple and straightforward. A triangle LOR (Fig. 4) defined via the three preselected landmarks is constructed.

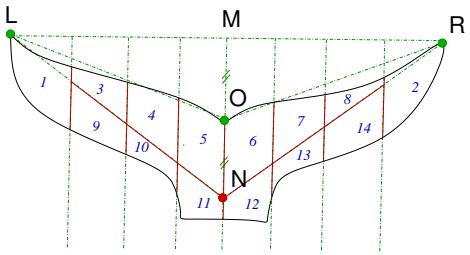


Figure 4: Affine grid construction.

The base of the triangle (the line connecting the fluke's tips L and R) is divided in two equal parts by the point M. The symmetrical point of M in respect to O, i.e. N is found. Each fluke is then divided into n parts with lines parallel to the median NM. Thus, the grid delineates $N_R = 4n - 2$ (the tips are considered single regions) grid regions. These regions are labelled 1 through N_R by scanning left to right, top to bottom. For the grid on Fig. 4 $n = 4, N_R = 14$.

Notice that since the construction is solely based on affine invariant concepts (i.e. middle point, symmetry, equal distances, parallel lines), the resulting grid is invariant under affine transformations.

3.2 Feature Extraction and Comparison

After the grid has been fitted to the segmented fluke an N_R-dimensional feature vector $\mathbf{f} = (f_1, \ldots, f_{N_R})$ is computed. Each element f_i equals the ratio of the number of white pixels to the total number of pixels in the i-th grid region. The feature vector for each fluke image is computed and stored in a database of features $\mathbf{F} = \{\mathbf{f}_1, \ldots, \mathbf{f}_N\}$ for all N images of the image database.

The matching process involves a comparison of the feature vector $\mathbf{q}$ calculated from a query image against each entry $\mathbf{f}$ in $\mathbf{F}$. This is done by computing the average Euclidean distance per fluke segment

$$d(\mathbf{q}, \mathbf{f}) = \frac{\sqrt{\sum_{i=1}^{N_R} I_i (q_i - f_i)^2}}{\sum_{i=1}^{N_R} I_i}, \tag{1}$$

where the indicator variable I_i determines if the corresponding region of any of the pair of flukes to compare should be considered, i.e.: $I_i = I_i^{\mathbf{q}} I_i^{\mathbf{f}}$. The indicator equals 1 for all regions above the clipping line and 0 for the ones which are occluded. Because different regions will be occluded with the different flukes we need to normalize the distance over the number of regions used for computation of the similarity of any pair of flukes. The images in the database are then ranked based on their similarity to the query image.

3.3 Adaptive Adjustment of the Grid

The flukes are submerged into the sea up to a different level, therefore the totally or partially occluded grid regions are not directly comparable. There are two ways of dealing with this problem, namely ignoring all affected grid regions (i.e. to set $I_i = 0$) or to adapt their relative size. If the first approach is adopted one can lose important information from characteristic patches/ markings located in the partially occluded regions. We propose an adaptive grid adjustment scheme. If the location of the point N is below the clipping line $c \parallel LR$, the level of occlusion can be defined as the ratio of the heights of the similar triangles as depicted in Fig.5: $l = h_c/h$.

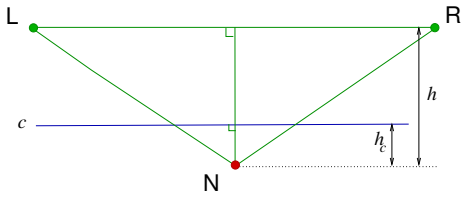


Figure 5: Adjustment of the coordinate grid to accommodate the different level of occlusion.

When comparing a query image to a potential match from the database the levels of occlusion may be different, i.e. $l^{\mathbf{q}} \neq l^{\mathbf{f}}$. If the database image has been occluded more than the query, i.e. if $l^{\mathbf{f}} > l^{\mathbf{q}}$, to preserve the area ratio the clipping tail line of the query has to be adjusted to a new height:

$$\tilde{h_c^{\mathbf{q}}} = \frac{h_c^{\mathbf{f}} h^{\mathbf{q}}}{h^{\mathbf{f}}} = l^{\mathbf{f}} h^{\mathbf{q}}. \tag{2}$$

Analogously if $l^{\mathbf{q}} > l^{\mathbf{f}}$, the clipping line of the potential match has to be adjusted. In this manner, it is possible to use the correct part of the partially occluded regions within $\triangle LNR$.

Therefore the final retrieval scheme is modified as follows. Initial matching is performed as described in Section 3.2. For the partially occluded regions within $\triangle LNR$ the indicator variable is set to 1. The query is compared against the whole database. Then the grid adjustment if performed between each pair of images: the query and the candidate within the top 20 from the initial ranking. During this process the features are recomputed for the new grid regions and a new final ranking is performed.

4 Results

A database of 69 gray-scale images of humpback whale flukes of different resolution and different quality was available for testing the proposed methodology. The database has been manually processed by an expert and 32 individuals were identified. For 5 of these individuals there were 3 different images available (triple) and for the rest (27) there were 2 images each (pairs) in the database.

4.1 Fluke and Patches Extraction

The watershed segmentation provided an excellent contour of the tail for most of the data at one iteration (immediately after the user specifies the tail marker). For the remaining images (mainly of poor quality), the user could achieve very good extraction after few iterations of fine-tuning additional markers using the GUI. The subsequent thresholding produced a very good binary representation of the flukes and the natural markings. Figure 6 illustrates the performance of the flukes and patches extraction for 2 pairs of images of the database. It can be seen that the segmentation captured the important markings well.

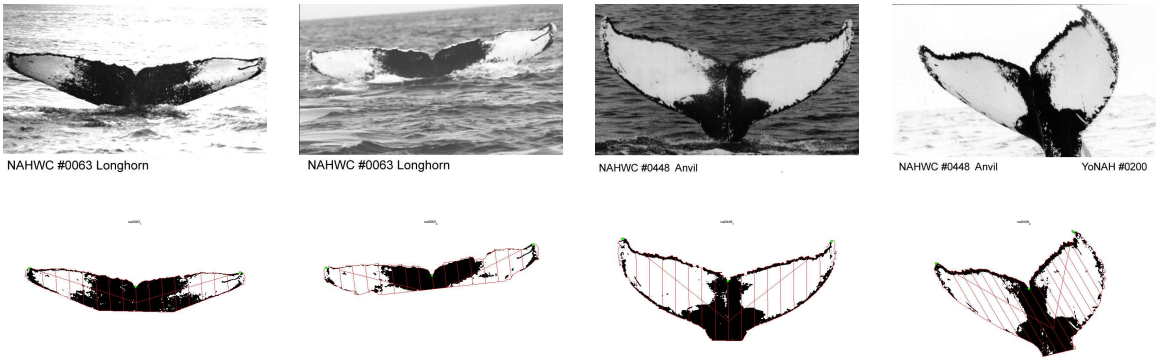


Figure 6: Segmentation and grid fitting to two pairs of images subject to affine transformation.

4.2 Grid Fitting and Matching

Figure 6 illustrates also the grids ($n = 8, N_R = 30$) fitted to the segmented pairs of images. It should be noted that salient markings appear in the correct grid region independently of the viewing angle and tail slant, especially visible for the second pair.

Two matching strategies were tested. The first one ignores all completely or partially (i.e. $I_i = 0, \forall i : R_i \cap c \neq \emptyset$) occluded regions, and no grid adjustment was performed. The second strategy performs grid adjustment within the top 20 matches obtained after an initial ranking as described in section 3.3. Although the first strategy is faster as it uses the pre-computed feature database $\mathbf{F}$ and no re-computation is needed, the second one achieves better retrieval results as summarized in Table 1.

Table 1: Percentage of individuals whose true match is ranked among the top k.

Grid adjustment	$k = 10$	$k = 3$	$k = 1$
No	94.2	81.1	60.8
Yes	100	84	66.7

Both strategies reduced the number of images which had to be reviewed by the expert by a factor of 7. All images had their true match ranked amongst the top 10 using the grid adjustment strategy. For more than two-thirds of the database images the true match(es) were correctly identified as the first (or first and second in case of triples). A more difficult case is illustrated in Fig. 7 where the true match of

a query image belonging to a pair was ranked third. It can be noted that the "false positives" are still visually similar to the query image.

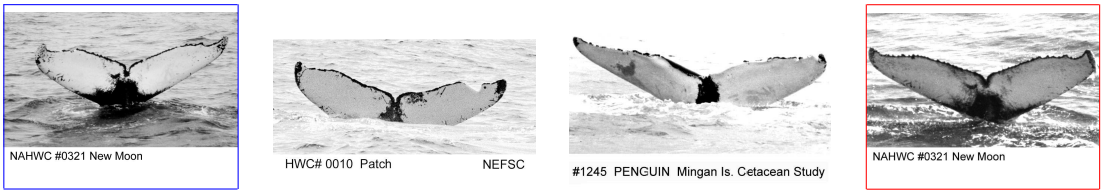


Figure 7: Query image (left framed in blue) and first three matches. The true match is framed in red.

5 Discussion

The work reported in this paper addresses two aspects of the photo-identification problem: the *segmentation* of relevant image information (fluke, patches) and the *feature extraction and matching* based on an affine invariant coordinate grid.

The *segmentation* program has been tested by marine biologists during a Europhlukes Project software evaluation test meeting, where it had a very favourable reception.

The performance of the *matching* needs to be confirmed on a much larger database. However, the methodology is quite generic and can easily be extended to other photo-identification problems (e.g. dorsal fins for dolphins). Also, our hypothesis is that on a larger database the grid adjustment will achieve more substantial improvement in the retrieval performance.

The current research efforts are focused on developing a salient patterns detector. The descriptors of the salient markings in respect to the affine coordinate system, could dramatically improve the recall specificity.

ACKNOWLEDGMENTS: This work has been partially supported by Europhlukes (www.europhlukes.net, EC project-ID EVR1-CT2001-20007). Judy Allen is gratefully acknowledged for providing the images.

References

[1] B.N.Araabi, Syntactic/Semantic Curve-Matching and Photo-Identification of Dolphins and Whales, PhD Thesis, Texas A&M University, 2001.

[2] B.N.Araabi et al., Generalization of Dorsal Fin Ratio for Dolphin Photo-Identification, Annals of Biomedical Engineering, 28(10), Oct. 2000, 1269-1279.

[3] P. Soille, Morphological Image Analysis, *Springer*, 2003.

[4] N. Kehtarnavaz et al., Photo-identification of Humpback and Gray Whales Using Affine Moment Invariants, *Proceedings of 13th Scandinavian Conference on Image Analysis*, Sweden, July 2003.

[5] G. Hillman et al., Computer-Assisted Photo-Identification of Flukes Using Blotch and Scar Patters, *Proceedings of 15th Biennial Conference on the Biology of Marine Mammals*, US, Dec. 2003.

[6] S. Mizroch et al., Computer Assisted Photo-Identification of Humpback Whales, *Individual Recognition of Cetaceans*, International Whaling Commission, Cambridge, 1990.

[7] N.Otsu, A Threshold Selection Method from Gray-Level Histogram, *IEEE Transactions on System, Man, Cybernetics*, SMC-9, 62-66, 1997.

[8] WhaleNet http://whale.wheelock.edu

VISUALISATION MODELS FOR IMAGE DATABASES: A COMPARISON OF SIX APPROACHES

Simon D. Ruszala and Gerald Schaefer
School of Computing & Technology
Nottingham Trent University, England, NG1 4BU
simonruszala@hotmail.com, Gerald.Schafer@ntu.ac.uk

Abstract
Provided is a critical evaluation of six visualisation models for image database navigation. Difficulties in visualising the results produced by content-based image retrieval systems have driven research into finding optimal ways of displaying these images so they convey as much information to the user as possible. Initial visualisation of an entire database is also a desirable asset which helps browsing through an image database as a whole. Research of these systems aims to find a system that integrates fast indexing and accurate retrieval with easily navigable and intuitive image database browsing. Accuracy of the displayed results are compared to the computational complexity taken to produce a visualisation determining if certain systems are realistically suitable for certain image databases. This paper provides details of the attractive features offered by each as well as the drawbacks, concluding in the best available system for the features desired.

Keywords: *Content-based image retrieval (CBIR), image database navigation, image database browsing, PCA, MDS, FastMap, picSOM, MARS 3D, hierarchical clustering, global visualisation.*

1. Introduction

Content-Based Image Retrieval (CBIR) is playing a major role in image retrieval systems such as those provided by stock photo companies. Initially concept based methods were adopted where each image were individually annotated and categorised before keyword searching could be performed. This method is still extensively used in many image database systems but the need for more detailed, automated and accurate indexing techniques has led to the adoption of CBIR which is based on features computed directly from images and a defined similarity between these features resulting in a computed resemblance between images which ideally corresponds to the visual similarity humans would assign. Properties of concept based methods meant they were able to catch the semantic content of an image by assigning descriptive words which (currently) could not be captured using CBIR (although the latter are able to match primitive features that are difficult do describe using words). Both are very useful for different reasons and are combined in some cases increasing the efficiency of a retrieval system.

Indexing systems such QBIC [2], visualSEEK [10] or NETRA [7] that use these methods of extracting content features generally have good retrieval results but lack when it comes to the facets available for searching. Presently their main method of retrieving similar images from a collection is to query by example where a specific image is used to initiate the search and retrieval of similar looking images. Other querying procedures such as query by sketch, metadata or text have also been investigated but are typically rarely used due to various drawbacks of each method. One requirement all 'query by X' methods have in common is that an initial input is needed to instigate. Often the precise content of the image sought after is unknown, making existing querying methods increasingly impractical: if you don't know the shape of the object in an image, you can't sketch it for querying, and if the colour content is not well known either then searching by this method will yield unsatisfactory results.

Another aspect which has not been extensively researched into is the visualisation of the images returned as the results obtained from querying an image database system. In general, all images and results are displayed in a one-dimensional linear fashion, either in rank order after a query or in the order they were read in from the database. This gives no indication to the user of where a certain image can be found unless queried. When visualising larger datasets, the number of images becomes far greater than are realistically viewable on a single screen. Drawbacks like this have lead to the research and development of how to arrange the images in such a way that they are positioned on the screen in relation to all other images. While this may cause significantly more images to be displayed at once the advantage is that all images are visualised at once and clusters of related images will appear which can then be investigated further. With the images positioned relative to their similarity with other images, the display gives structure as homing in on one area means all neighbouring images will be alike.

Displaying an entire dataset on a single screen and allowing the user to localise specific areas to home in on creates the option of browsing whilst also giving indication of the size of the collection. Navigation can be accomplished through a top-down hierarchical approach (through zooming into an area of interest) thus giving more visual information on the entire range of the database to the user. Using this approach to visualisation is a widely desired feature by both users with personnel image albums and businesses that manage larger image compilations.

This paper surveys six methods that have been used to visualise image datasets and introduce the ability to browse freely through them. Besides explaining the underlying techniques advantages and disadvantages of each method will be highlighted and a recommendation for a useful visualisation system provided.

2. Principal Component Analysis (PCA)

Principal component analysis involves a mathematical procedure that transforms a number of high dimensional correlated variables into a smaller number of uncorrelated variables called principal components by reducing the dimensionality whilst preserving the 'essence' of the data. High dimensional data is normally vast in size and ungraspable by the human mind, making some form of representation necessary. Technically this involves the computation of the top eigenvectors of the original distance covariance matrix. This is the most common method of embedding axis in a linear combination of the original axis. In terms of image database visualisation the input data typically consists of pairwise relationships between data elements, normally similarities or distances.

Various approaches exist that perform the operation of finding the bases which best maximise the variance operation; the one briefly explained here is based on the relationship between PCA and the Singular Value Decomposition (SVD). Initially the mean vector (the 1st principal component) of all samples is calculated and subtracted from each dimension (hence resulting in a distribution with the origin as its mean). SVD computes the remaining components by producing a diagonal matrix with eigenvalues in descending order. Each singular value from the SVD is proportional to the square root of the variances (proportionality constant is the 'unbiased' covariance estimator $1/(N-1)$). The corresponding eigenvectors are the principal components. Once these have been calculated all samples (i.e. images) in the database can be projected on the principal components and the projection weights be used for assigning co-ordinates for the display of each image, i.e. for the display in a two-dimensional space e.g. on screen the first two principal components are exploited.

Using this linear strategy is more limited than their non-linear counterparts but still hold some advantages. Results shown are reliable, with genuine properties of the original data if the similarity matrix were constructed using the L2 norm (Euclidean distance). If distances between images are based on any other norm (e.g. L1 norm – 'Manhattan' distance) or indeed any other distance function, the results will not be as reliable which follows from the fact that PCA maximises the captured variance in a least-squares Euclidean way. Hence, if accuracy is of interest, further configuration rearrangements should be considered as any non-linear correlation between variables is missed out and not captured. On the plus side, the mapping of images to display co-ordinates is straightforward. The way to compute these positions is very efficient as PCA calculates them using a linear approach, hence the overall computational complexity is relatively low.

3. Multidimensional Scaling (MDS)

Multidimensional scaling [5] expresses the similarities between different objects in a small number of dimensions, allowing for a complex set of inter-relationships to be summarised in a single figure. MDS can be used to analyse any kind of distance or similarity/dissimilarity matrix created from a particular dataset.

There are two types of multidimensional scaling methods, metric and non-metric. *Metric* MDS is where the distances between the data items are given and a configuration of points that would give rise to the distances is sought, for example distances between cities in a particular country would use metric MDS. This perfect reproduction of distances is not always possible, in which case *non-metric* MDS would be used. Non-metric is where the calculation between rank orders of similarity Euclidean distances and rank orders in the original space are computed to produce a set of metric co-ordinates which most closely approximates their non-metric distances.

The application of MDS for image database display and navigation was first proposed by Rubner (Rubner *et al.* 1997). Rubner produced a way of not only visualising the retrieved images in terms of decreasing similarities but also according to their common similarities. By using non-metric MDS to implant all images by their similarities in a two or three dimensional Euclidean space, these calculated distances could be preserved.

For non-metric calculations a similarity matrix need be obtained from the CBIR techniques previously calculated. Euclidean distances are calculated and initially compared using Kruskals' [5] '*stress formula 1'*.

$$\text{STRESS} = \frac{\sum_{i,j} (\hat{S}_{i,j} - S_{i,j})}{\sum_{i,j} S^2_{i,j}} \quad (1)$$

This algorithm expresses the difference between the similarity values 'S' and the Euclidean values 'Ŝ' between all images. The aim of non-metric MDS is to assign locations to the input data so that the overall stress is minimal. Typically an initial configuration is found through PCA as described in the previous section. While the degree of goodness-of-fit after this is in general fairly high it still can be improved. To do so the locations of the points are updated in such a way as to reduce the overall stress. If for instance the distance between two specific samples has been overestimated it will be reduced to correct this deviation. It is clear that this modification will have implications for all other distances calculated. Therefore, the updating of the co-ordinates and the recalculation of the stress is being

performed in an iterative way where during each iteration the positions are slightly changed until the whole configuration is stable and the algorithm has converged into a minimum where the distances between the projected samples correspond accurately to the original distances. Several termination conditions can be applied such as an acceptable degree of goodness-of-fit, a predefine maximal number of iterations or a threshold for the overall changes in the configuration. Once the calculation is terminated the points can then be mapped onto the screen.
Navigation through this program starts initially with a global display of the entire database with images positioned in relation to how similar they are with all others. From here the user has the ability to zoom into certain regions of interest to enlarge and allow for further querying. For each localised visualisation occurrence, the images selected in the area have their similarity distances recalculated and projected back into two-dimensional format. This accommodates for the enlargement so to occupy the entire screen when displayed. When a region is localised on the distances need to be altered in accordance to the new screen co-ordinates to give maximum visibility of the images. Along with this the images need to be sized, both of these are time abundant relating in a time delay each time an area is localised. This is not a desirable attribute of the system, as normally it would be expected to happen at real time.
All other querying methods are still achievable as long as the appropriate CBIR techniques have been implemented. Retrieval of images from either a sketch or an example, results in the appropriate images being displayed around the selected image in accordance to their similarity. Normally only a certain amount of images, e.g. 15, will be retrieved as too many causes clustering to occur and the display of insignificant results.
Disadvantages of this accurate positioning system are the computation time needed to re-calculate the stress value to obtain the best available configuration of points. As it uses a quadratic approach to compute distances etc. it means it is computationally expensive, '$O(N^2)$', where 'O' represents the object and 'N' the number of items, making interactive visualisation of a large number of images unsuitable.

4. FastMap

Another system in which proposes mapping points from a one-dimensional k-d space such that the dissimilarity distances have little discrepancies, is the FastMap algorithm [1]. In general this is a computational simplification of the MDS procedure based on the geometrical reflection. Vector projections and distances are updated to a degree of accuracy to discover the best configuration of points by iteratively discovering the direction of the strongest component vector.
FastMap aims to display images in a global manner, similar to that of MDS and PCA, where the positioning of images depends on the dissimilarities between all pairs of images. Additionally it attempts to improve on existing methods by computing the results and then displaying them in a more realistic time scale. CBIR techniques are not required for this algorithm to compute; instead the only input required for this to work is the distance function. The FastMap algorithm automatically extracts suitable distance features from each object, approximations are made between the interpoint distance scores with the results being estimated. This is an unbiased approach which calculates extremely quickly especially in comparison to that of completing a full multipoint matrix.
Linear mapping is used to produce results; the idea behind this is to calculate the properties of two objects (pivot objects) that parse through a carefully selected line in n-d space using the cosine law. Pivot objects are ideally two objects that are as far apart as possible and are chosen using a linear heuristic algorithm. This process chooses one object at random and another by finding the point furthest from it, this found object is then set as the furthest from the arbitrary one and both are returned as the pivot objects. This heuristic algorithm is completed until all objects have been mapped onto lines.
When a query by example is performed the pivot objects values are required for knowing the lines of appropriate points so the query can be mapped into a point in k-d space. For this reason storage of each pair of pivot objects is required after each recursive call. Querying methods are performed at a faster rate than other systems due to its integration with highly fine-tuned Spatial Access Methods (SAM) such as R-trees and R*-trees. All of this are computed linearly to keep the computation time down as if it was done by finding the maximum value of the two objects, there would be '$O(N^2)$' computations performed instead of just '$O(N)$' in turn slowing the algorithm down. The results can be extended causing the mapping of two objects on a line in 2-dimensional space whilst still preserving some of the distance information. For mapping to occur in k-d space, projection of all other calculated distances need to be estimated. These can then be placed onto several lines in n-d space by construction, by recursively repeating this procedure results in the ability to project these points into k-d space. Computation time is reduced as it is linear on the number of objects it requires only '$O(N)$' calculations, meaning time taken for the algorithm to map a new object onto the display is reduced dramatically without significant loss of output precision.
Using this approximation procedure can give way to some of the approximations being considerably different from their acceptable value. This algorithm reduces this degree of inaccuracy but unfortunately results in some discrepancies appearing.

5. picSOM

A system designed for use with the internet, retrieval of images is conducted through a web browser on databases exceeding one million pictures in size.
T. Kohonen [3] developed a neural network algorithm using Self-Organising Maps (SOM) to manage images into map units. Its uniqueness comes from the ability to run SOM's concurrently, both to reduce the amount of data by clustering, and to construct a non-linear projection of the data onto a low dimensional display.
Using slightly more advanced Tree Structured Self-Organising Maps (TS-SOMs) as the image similarity ranking method can be used for creating a hierarchical representation of the images in a particular database has been approached by Laaksonen, Koskela and Oja [6] in the creation of the picSOM system. This prototype system utilises numerous content-based image retrieval techniques, either individually or in parallel, to adapt to all different types of database from large to small and domain specific to random. For each statistical feature vector used to retrieve images, an additional two-dimensional TS-SOM is created; resulting in numerous TS-SOM's being grouped in parallel for calculating the best similarity results. If necessary, additional feature vectors can be introduced and integrated with ease. The main advantage is that the user can specify different queries for different features with the system automatically computing the input data and retrieving the results dependent on the queries selected.
TS-SOM's are a vector quantization algorithm which uses a hierarchical structure as its indexing method where each level in the structure contains its own Self Organising Map (SOM). Tree structure consists of an increase in complexity the further down you go with each level, the space available is constrained in relation to the content in the above and below SOM's. Using TS-SOM's instead of SOM's reduces the complexity dramatically.
A novel approach used by this system, the picSOM engine tries to learn progressively what the user wants from the interaction from previous searches performed. Training the system this way means it can predict, to a certain extent, what type of images the user is after. The rationale for this is for each image that is selected as relevant whilst searching, similar ones are marked positively and dissimilar ones marked negatively, similar images selected have an increase in their relevance weighting. This is done both automatically by the computer and manually where the iterative process of selecting or rejecting images is performed by the user. Over a number of searches some image weights are increased to an extent that images retrieved on the forthcoming queries should contain these neighbouring images with increased weights from the training scheme. Applying a weighting system that helps the user track down images of significance before they have been viewed is a great feature not available on other retrieval systems.
Visualising both the results of querying and general browsing is not very appealing as it uses the one-dimensional linear approach to display. For an image retrieval system that deals with significantly large collections of images, analysing the database properly is not realistic or will be extremely time consuming.

6. MARS 3D

A novel prototype interaction visualisation approach to displaying image databases is presented by M. Nakazato [8] in the system *3D MARS*. This system displays the images in a projection-based immersive virtual reality or non-immersive desktop virtual reality manner named *imageGrouper*. This either allows the user to control and navigate through the database in a large three-dimensional VR *CAVE*, by the use of a wand allowing the control of image selection and retrieval from the database. This concept allows the viewer to see a stereoscopic view of the space by displaying the images on 4 walls, top, bottom, left and right, encompassing the user. There are only so many images that can fit on a screen at any one time as the screen size is normally limited e.g. 15" - 21" monitor. With large screens surrounding the user on four different sides means more images can be visualised at one time in all three of the x, y, and z axis than traditional methods (just x and y) as well as images able to be displayed at a greater size. This approach could be expanded to accommodate six sides completely immersing the user as if inside a dice and all walls of the dice were screens showing the image database. This would allow all visual limitations present elsewhere in other systems as it would be like walking through an art gallery. Using a *CAVE* virtual reality system like this is very unique and expensive with hardware equipment not feasibly available by the general user; instead it would be used by specific companies with a real need for image database navigation interaction.
Concept of retrieving the images comes as a result of calculating a similarity matrix for each of the CBIR techniques adopted, then finding the distances from a selected image to all other images available. Biased Discriminant Analysis [12] is used in calculating the CBIR feature weights. These feature weights are combined and an overall weight is assigned to each image with the smaller results perceived as more similar to that of the query image.
Browsing is available through this system in two ways, either the user has to initially manually search through the image database in hope that they will come across a similar image, this occurs when no querying has taken place and as the images are randomly positioned there is no structure to where images will initially be placed. Alternatively browsing can be accomplished after a query has taken place where images will have some spatial structure. The system retrieves all similar images in respect to the query and displays them in accordance to their similarity using colour axes stating the directions of the most similar colours. Number of results retrieved can either be specified e.g. 30, or the entire rearranged database can be returned. From this point browsing becomes a lot easier due to the colour axes as the

three primary colours are used for each direction. If the user browses in one of these directions then images relating to that colour can be viewed, the further away they venture from the axis means the images decrease in that particular colour. An advantage of this system is the allowance of numerous selections of images for querying performed for one particular search. In selecting several images for querying causes the rearrangement of the entire database, hopefully, depending on the CBIR techniques chosen, will yield a more accurate representation of the database for viewing. A better structure needs to be established to achieve the initial display for browsing as the likelihood of coming across an image sought after in a database of e.g. 500,000 is very unlikely.

7. Hierarchical Clustering

Using an automated tree structure and clustering scheme this approach devised by Krishnamachari and Abdel-Mottaleb [4] stores groups of similar images at different levels of a hierarchical tree. Top level images of this tree are fairly dissimilar allowing the user to choose from a diverse range of images, hence narrowing down the types of images sought after. Increasingly going further down the tree refining your search, the images become more alike, this eliminates the linear browsing method so abundant on other existing systems.
Images are initially arranged by assigning similarities between all pairs of images through automatically calculated local histograms consisting of 16 rectangular regions. Then, by using a histogram intersection algorithm, computation of a single weight for all inter-relationships can be conducted.
Clustering is organised such that initially each image from the database is allocated as an individual cluster then all are indexed sequentially. Two clusters are selected and merged as one by locating the two having the largest similarity difference; this reduces the number of unmerged clusters. One of the two is assigned to the left of this newly created cluster and named as the '*left child*' and the other to the right as the '*right child*'. Re-calculation of the similarities between merged and unmerged clusters is performed, but not between merged clusters, this reduces computation time. Similarity weight of a cluster is calculated from the average of all pairs of images in that particular cluster. This sequence of operations is recursively computed until only a single cluster remains. Each cluster ranges in size where the further down the tree, the larger the number of images held by each. For every one a representative image is used for navigation around the tree, these are selected by choosing the most diverse images from the sub-groups representing as many clusters as possible so not to eliminate any further paths down the tree. Browsing takes place by the user selecting representative image which in turn returns the lower level cluster of images. If these images retrieved were not sought after then navigation back up the tree is possible. Query-by-example still exists in this system with the query histogram being compared against each clusters combined histogram finding the best group of images, certain images with similar weights from that cluster can then be compared individually to find the similar images. This technique of querying is considerably quicker than previous methods due to only a subset of the images being compared for similarity instead of the entire database. Results using databases greater than 3500 in size have shown the retrieval accuracy based on this approach is high as well as the computational time to complete low. Although browsing of the database is possible, the images returned are displayed in a one-dimensional manner causing problems when it comes to large databases. Clusters of images can become larger in size and the number of levels in the hierarchical tree becomes so large that searching through it becomes a burden. Also it is quite feasible to suggest that some images will be grouped incorrectly, mainly because of human perception being different to that of the automated similarity weights that will be assigned to it. This results in the image becoming lost unless accidentally stumbled upon while searching in other clusters for different images.

8. Conclusions

Mars 3D CAVE system is one novel approach to displaying databases in an interesting and exciting way which can make the user feel as if they are part of the database, from this they are more able to interact with it. Ventures like this are less available and researched due to the cost of specialist hardware and the impracticality of their size. Searching using this method, after training, would be easier, quicker and more productive as the images are displayed like pictures in an art gallery, but the problems of global browsing are still apparent as there is no spatial arrangement. Querying results from this system are very similar in the layout used by PCA/MDS and FASTMAP making it very easy to use and understand but the global view of the database lets the system down significantly. PicSOM is let down by the way their results are displayed, one reason for this is it deals with online users who aren't willing to wait for the optimal display methods to compute. Time to compute a query, even in large datasets, is very low and the accuracy is increased with each search due to the difference in the way it indexes images. The relevant feedback algorithm has obviously given it a competitive advantage of progressively weighting images not seen by the user but similar to those that were. Hierarchical clustering allows for browsing in a linear display approach but does not allow global visualisation of the total database. It is quick to compute and retrieves images accurately but becomes increasingly complex with larger databases, with clusters and tree levels becoming much larger. This results in a more time consuming search procedure for the user as navigation through numerous clusters has to be performed. This approach solves a couple of the

problems that are apparent elsewhere but in comparison to the other prototypes researched, the time consideration of searching is not as practical.

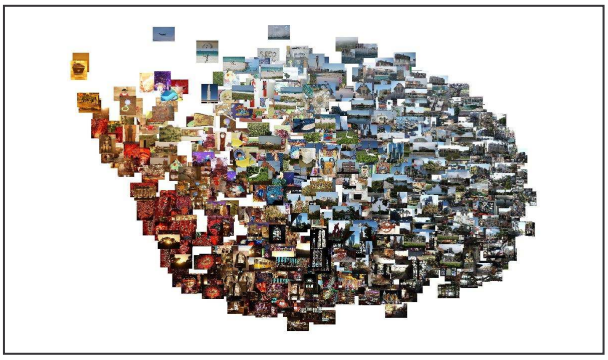


Fig: 1.1 – ***MDS display of entire UCID database images [11]***

Fig: 1.2 – ***Localised display of bottom right region from the UCID database images***

PCA, MDS and FastMap all use the same display technique which can be seen in Fig:1.1, with the difference between them being the time taken to compute and the accuracy at which they do this. PCA is the most inaccurate of the three with an approach that only computes once to gain the desired configuration. As this is a linear approach the time taken to complete the algorithm is very quick and normally can be achieved in next to real time depending on the size of the dataset. MDS goes one step further by re-calculating the configuration of points if it is not at an optimal degree of precision is reached. This process is computationally a lot slower, mainly because it uses a quadratic iterative algorithm approach to its calculations, but the level of accuracy is far greater and acceptable than that of the other methods. Because of the time factor, dealing with simple queries like 'query-by-example' becomes difficult, especially at run time and if dealing with large sets of data. FastMap goes one step further than MDS by eliminating the time problem that burdened it by using linear algorithms to calculate distances between images. The accuracy reached by this algorithm is very good but does not quite meet the level MDS achieves, although the difference is minimal.

There are many desirable features in an image retrieval and visualisation system as seen through the analysis of these six systems, each one having their own drawbacks and sought-after features. Visualising images in the immersive cave way is an intrinsic method that can engross the user making searching a lot more interesting. Realistically at this moment in time they will not be widely used due to cost and size of them, but are more productive in what they can and will be able to do. From systems that can be used on desktop computers by all people, the PCA/MDS/FastMap display method seems to be the way forward, providing global (Fig:1.1) and localised (Fig:1.2) views of databases with spatial relational arrangement. Navigation is considerably easier and the speed at which it can be conducted improves as well. Out of the three methods that have adopted this visualising technique FastMap has conquered the drawbacks of the other two, even though the accuracy is not as precise as MDS. MDS would only be used over FastMap if accuracy was the driving force behind the arrangement of data and the highest degree of accuracy was sort after.

References

[1] Faloutsos, C. and K. Lin. 1995. " FastMap : A fast algorithm for indexing, datamining and visualization of traditional and multimedia datasets." *Proc. Of SIGMOD95*, (May), 163-174.

[2] Flickner, M. et al., 1995. "Query by image and video content: The QBIC system," *IEEE Computers*.

[3] Kohonen, T. 1990. "The Self-Organizing Map." *Proceedings of the IEEE*, vol.78, 1464-1480.

[4] Krishnamachari, S.; and M. Abdel-Mottaleb, 1999. "Image Browsing using Hierarchical Clustering." *Proc. IEEE int. sys. On Comp. & Comm*., (July), 301-307.

[5] Kruskal J. B., and Wish, M., 1978. "Multidimensional Scaling,". SAGE university paper series on quantitive applications in the social sciences, 07-011.Sage Publications,Newbury Park, CA.

[6] Laaksonen, T.; J. Koskela; P. Laakkso; and E. Oja. 2000. "PicSOM – content-based image retrieval with self organising maps." *Pattern Recognition Letters* vol.21, 1199-1207.

[7] Ma, W. Y. and Manjunath, B. S., 1999. "Netra: A toolbox for navigating large image databases." In *Multimedia Systems*, Vol. 7, No. 3, 184-198.

[8] Nakazato, M. and Huang, T. S. 2001. "3D MARS: Immersive Virtual Reality for Content-based Image Retrieval." In *Proceedings of IEEE International Conference on Multimedia and Expo 2001*.

[9] Rubner, Y., Guibas, L., Tomasi, C., 1997. "The earth mover's distance, multi-dimensional scaling, and color-based image retrieval." In *Proceedings of the ARPA Image Understanding Workshop*, (May).

[10] Smith, J. R. and Chang S-F., 1996. "VisualSEEk: a fully automated content-based image query system," In *ACM Multimedia'96*.

[11] Schaefer, G., and Stich, M., 2003. "UCID – An Uncompressed Colour Image Database", *In Proc of SPIE*, 5307, 472-480.

[12] Zhou, X. and Huang, T. S, 2000. "A Generalized Relevance Feedback Scheme for Image Retrieval," In *Proceedings of SPIE* Vol. 4210: Internet Multimedia Management Systems, 6-7 November 2000, Boston, MA, USA.

ACTIVE CONTOURS MULTIOBJECTIVE OPTIMISATION BY HYBRIDS ALGORITHM - APPLICATION TO LIPS CONTOUR EXTRACTION

N. Cladel
IETR/SUPELEC Rennes- Team ETSN
Avenue de la Boulaie BP 81127
35511 Cesson-Sévigné Cedex
France
nicolas.Cladel@supelec.fr

R. Séguier
IETR/SUPELEC Rennes - Team ETSN
Avenue de la Boulaie BP 81127
35511 Cesson-Sévigné Cedex
France
renaud.seguier@supelec.fr

Abstract
In this paper we propose an evolution of Multiobjective Genetic Snakes (MGS) [12] by adding a new genetic hybrid algorithm and local search technique in a multiobjective context. We propose to use the finite difference method [7] for the local method to keep the energy multiobjective optimisation. The application context of this work is the noised and bad segmented images segmentation.
We apply this new algorithm on the lip's contours extraction in real images. The internal and external contours of the lips are coded according to the model of double concentric snakes. Two energies are used to deform the snakes, the first one is a distance map based on gradient energy. The second one is region based and is used to control the deformation.
This local search algorithm is implement in the classical multiobjective genetic algorithm NSGA2 [14] with the representation of MGS. It has been tested on noised images of lips.

Keywords: active contours, hybrid algorithms, labial contours, multiobjective optimisation, Pareto set.

1 Introduction

Since their creation [7], active contours have been much modified to respect researchers requirements [6]. This evolution has gone on in particular since the creation of levels sets [3,10]. However the problems of active contours initialisation is left [4], as well as the energies coefficients determination [8]. The Genetic Snakes (GS) [1] represent a solution to the initialisation problem through the global analysis of the image. The Multiobjective Genetic Snakes (MGS) [12] have been proposed to improve convergence speed and to make energies determination easier. However, one of the main problems of genetics algorithms is the convergence precision. Consequently, the MGS doesn't allow a good segmentation of lips contours in noised images. A common solution used to improve this precision is hybrid technique GA and local search [13]. The implementation of this kind of algorithm within multiobjective context is a partial one because it only concerns the GA [2].Thus we propose an hybrid genetics snakes method with a complete multiobjective implementation by adding an algorithm of classical snakes, the finite difference method (FDM) [7].This paper is organized in three others sections. The second section describes the genetics snakes chromosome coding and the energies evaluation within the Pareto's principle. The third one presents the local search method, the snakes finite difference method, of our hybrid algorithm and its implementation. Section four describes the application of our method to the lips contours extraction and shows results.

2 Multiobjective Genetics Snakes

In this section we will briefly present here the representation of the Multiobjective Genetic Snakes. This technique introduce the multiobjective optimisation into a genetic snakes algorithm and the double snakes coding.
The genetic algorithm principle consists in applying genetic operators (cross over and mutation) on a population of chromosomes. Then these chromosomes are evaluated according to the energies of the problem and represent the new generation of the algorithm. Each chromosome is composed by genes which are the variables of the problem. The values of these variables compose the set of candidates. In the MGS, candidates are the gradient and skin image edges obtained by the filter of Canny-Derriche. The chromosomes of the algorithm are a set of randomly initialized mouth contours (figure 1). They are composed by the edges candidates and coded according to the double snakes model which we'll present in the next section. The chromosomes evaluation is performed by Pareto multiobjective optimisation of gradient and region based energies (section 2.3).

2.1 Double snakes coding

The aim of the double snakes coding is to represent a surface with a hole in a two dimensions approach. Indeed the region enveloped by a single contour is not homogeneous its association to a global minimum of region based energy is difficult. With the double snakes model, the region of interest can be extracted alone.
Within this principle, the mouth contours are represented by the inner and the outer contours :

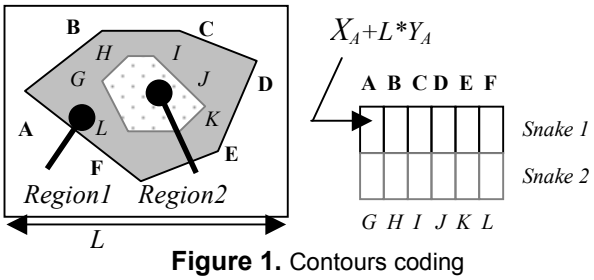


Figure 1. Contours coding

The position of the nodes of the contours are represented by their index in the image. Both contours envelop two regions, the centre of the mouth and the lips. These regions are respectively a minimum and a maximum of the red chromatic component. These region energies will be associated to a gradient based energy.

2.2 Multiobjective evaluation

Chromosomes are ranking on the basis of Pareto non-domination. If E_i are the active contours energies, a contour C is called "non dominated" if it doesn't exist an other contour such all its energies are better. With this principle a group of optimal contours (Pareto's set) can be defined at each generation and a rank can be calculated for each population contour [15]. At the end of the algorithm the Pareto's set is obtained. The final step consist in selecting the optimal bi-snakes configuration. This configuration can be obtained by minimizing the Euclidian distance between the selected contours and the origin of their set.

2.3 Energies

The region based energies are determinate from the image red chromatic component (*ImChromatic*) and from a binary region image (*ImBinary*). This binary image is calculated from *ImChromatic* during the preprocessing.
In the MGS, the region based energies are obtained by filling contours with a morphological algorithm. This method needs a large computation time, thus we use here the Green-Riemann's theorem to estimate the surface (defined by the contour {x,y}) integration of the region descriptor D [16].

The contour is obtained by the Bresenham's lines computation between each node. This contours discretization induce errors. All pixels belonging to the contours are not taken in account contrary to some pixels not belonging to the region. Moreover, information about the number of pixels belonging to the region is useful to avoid contours collapsing. Thus we use the discrete Green's theorem to fill the region [11] to extract pixels $\{p_i\}$ inside the contours. We also define the number of binary pixels N_1 (pixels at 1 in *ImBinary*) inside the contours and respectively the number of non binary pixels N_0 (pixels at 0 in *ImBinary*). We use two regions descriptors, one relative to the homogeneity (equation 2-a) and one relative to the accumulation of pixels (equation 2-b).

$$a : E^R_{homogenity} = var(R) = \frac{1}{N_i}\sum_{k=1}^{N_i}\{Mean(ImChromatic(p_i)) - ImChromatic(pi)\} \quad - b : E^R_{accumulation} = \alpha.N_1 - \beta.N_0$$

Equation 1. Region descriptors

In equation 2 α and β are weighting coefficients. Within the double snakes model, the region of lips and the region of the centre of the mouth each characterized by the two descriptors. The lips region (region 1 in figure 1) is obtained by removing the centre region (region 2 in figure 1) from that one inside the outer contour.
The gradient information of edges candidates is not always sufficient and so we define a gradient based energy to keep the contour along the lips. We use the classical gradient energy with a density coefficient to favor continuous succession of edges :

$$E_{gradient} = \exp\left(-\left(N_{edges}/N\right)\cdot\sum_{k=1}^{N}\left|\nabla ImChromatic(x(k), y(k))\right|\right)$$

Equation 2. Gradient based energy

N is the contour size $\{x,y\}$ and N_{edges} the gradient edges belonging to the contour number.

3 Local search algorithm

With the MGS representation, we can obtain contours near the energies minima. To improve the convergence of theses contours we propose a new hybrid algorithm in a multiobjective context. Our aim is to make converge the chromosomes on local minima. Applying active contours during the genetic algorithm could give better configurations and so help the global convergence of the algorithm. Thus, the classical snakes algorithm will be applied on a little neighbourhood of each selected chromosome.

3.1 Model

It is usual to apply Hill Climbing Operator (HCO) as local search method in hybrid algorithms. We propose to use the classical snakes algorithm [7] to keep a multiobjective deformation. Indeed this active contours algorithm make it possible to use several image energies and integrate cohesion energy into the deformation equation. This method is based on the resolution of the Euler-Lagrange equation by partial differential equation.
Thus the total energy is :

$$E_{total}(v) = \int_0^1 \left(E_{int}(v(s)) + E_{ext}(v(s))\right)ds$$

E_{int} represents energies of curvature and tension. The external energy represents image energies. Then the deformation is performed by the Euler-Lagrange equation (v(s) are the pixels contours $\{x(s),y(s)\}$) :

$$\frac{\partial}{\partial t}\left(\mu\frac{\partial v}{\partial t}\right) + \gamma\frac{\partial v}{\partial t} - \frac{\partial E_{totale}(v)}{\partial t} = f(v)$$

3.2 Energies

The determination of internal energy coefficients, α and β, is difficult. Some authors have proposed some approaches to calculate them in accordance with the first and second contour derivate. Considering the local convergence and the computation time constraints we will manually determine the coefficients internal energy.
External energies are the image forces and so are submitted to the image noise. In order to control snake deformation and to improve robustness we use two external energies. The first one deforms the active contours and the second one controls the deformation. In noisy images, edges are often fuzzy so it is hard to exploit their direction. Thus we define a distance energy (equation 4) between the current node and edges candidates. To satisfy a multiobjective energy evaluation the selected edges candidates must not deteriorate the region based energy E_{region} corresponding to the current contour. At this step of the algorithm our aim is to find the local minimum so we search candidates inside a little neighbourhood V of the current node.

$$E_{v=x,y} = t_{node} - t_G \text{ with } G = \underset{v \in V}{Max}(\nabla I(v).Cr(v)) \qquad \text{with } E_{region}(node) \begin{cases} E_{region1} \text{ if node} \in \text{outer contour} \\ E_{region2} \text{ if node} \in \text{inner contour} \end{cases}$$

$$Cr(v) \begin{cases} 1 \text{ if } E_{region}(v) \le E_{region}(node) \\ 0 \text{ else} \end{cases}$$

Equation 4. External energy

The region energy E_{region} can be the homogeneity one or the accumulation one. We prefer using the energy of homogeneity cause it's more precise and the risk of collapsing is minimal during this step. This multi energies representation is more efficient than a simple weighted sum but its adjustment is more difficult than the Pareto representation.

3.3 Implementation

The local search method is used to improve the exploitation characteristic of GA. This improvement is performed by finding the local minimum of chromosomes at each generation of the GA.
Thus we apply the finite difference method on the chromosomes of the Pareto's frontier at certain iterations to let the genetic algorithm converge near a minimum and to minimize time computation.
In fact, the principle of the hybrid approach is that GA place contours near the global minimum then the classical snakes algorithm fall them on the local minimum.

4 Application to lips contour extraction

mouth Images and videos are difficult to segment because the region image is often noised. This noise can be due to the tongue, luminosity reflections, etc … . An other noise source is the mouth shape variation. For these reasons, lips contours extraction is difficult to automate and is useful in human and computer communication systems like AVSR (Audi Visual Speech Recognition) systems, avatars.

4.1 Coding and image preprocessing

We use the double concentric snakes with eight nodes for each contour to modelize lips contours. Image preprocessing concerns the candidates definition set and energies determination.
The active contour has to respect the model configuration thus we do not have intersections. To reduce the number of possible configurations, nodes have to be rank in the image. For these reasons we have to determinate outer contour nodes evolvement area. The determination of theses areas is based on the centre of the mouth [12]. The inner nodes evolvement area is a triangle defined by the outer node correspondent, the follow-

ing outer node and the mouth centre. During image preprocessing we construct a skin image of the mouth. This binary image is based on the HUE image.

4.2 Results

We have test our algorithm on European Data Base M2VTS images (Multi Modal Checking for Teleservices and Security applications [9]).Here are some results (figure 22) obtained on fifty iterations with the two region based energies and the gradient based one, and with a population of twenty chromosomes. The local search algorithm is applied every 10 generations. We present contours on the region image obtained during preprocessing. On the first lines we show results of MGS without local search algorithm and on the second line, the results of our hybrid algorithms apply on the external contour.

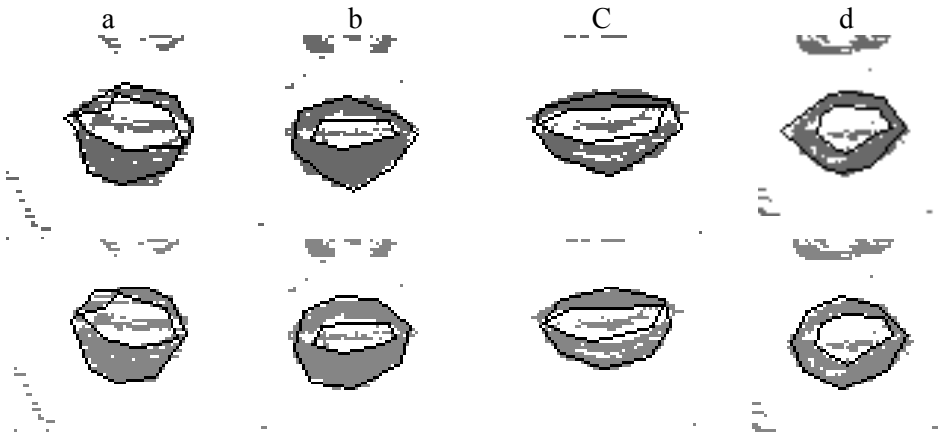


Figure 2. Examples of final results

The weighting coefficient in the region based energies make algorithm robust to noise on the region information. Thus we can see (figures a, b, c, d) that contours can envelop the mouth in spite of the tongue. On the same way figures d and e show that the algorithm is robust to luminosity reflection. On the figures f and g we see that we can extract lips contours in badly segmented images. We show (figure 3) the effect of the classical active contours during the genetic algorithm. The black doted contour is the current contour obtained by genetic algorithms and the white contour is the result oh the local search algorithm. On these examples, the local part algorithm is only apply on the external contour.

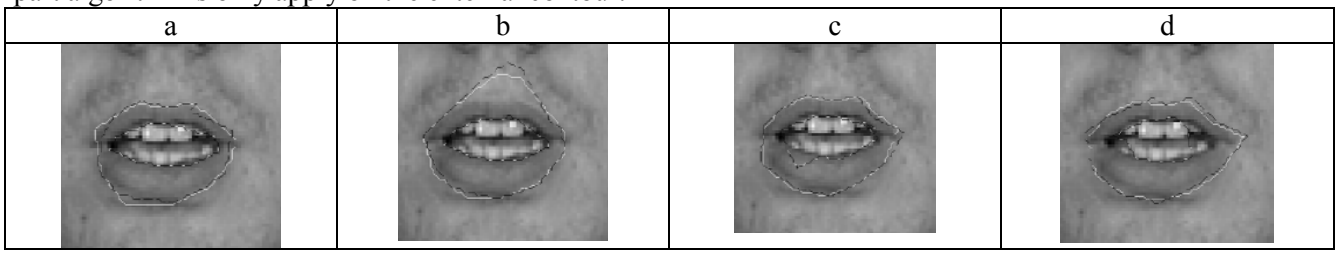


Figure 3 . implementation of classical snakes during GA

We can see that the classical snakes make chromosome converge in a little neighbourhood towards the real contours and so fit better the mouth.

5 Conclusion

In this paper we proposed a new implementation of a hybrid algorithm GA / local search in a multiobjective context. This new algorithm improves convergence of Multiobjective Genetic Snakes quality and reduces the

number of iterations. With this approach we can extract lips contours without being initialised near the lips. Moreover, our algorithm is robust to the tongue presence, luminosity reflections and badly segmentation. Nevertheless, more an image is noised and more the number of generations has to be important.
In our future works , we'll implement our algorithm on all the M2VTS Database to have quantitative comparison between the classical MGS and our hybrid method.

Acknowledgement

This research was supported by Brittany Region ("Région Bretagne") in France.

References

[1] Ballerini, L. (2001). Genetic snakes for color images segmentation. *Lecture Notes in computer sciences 2037*.

[2] Bhanu, B., Lee, S., Das, S. (1995). Adaptive image segmentation using genetic and hybrid search methods. *in IEEE Transactions on Aerospace and Electronic Systems*. 31(4):1268-1291.

[3] Caselles, V., Kimmel, R., Sapiro, G. (1997). Geodesic Active Contours. *in International Journal of Computer Vision*. 22:61-79.

[4] Eveno, N., Caplier, A., Coulon, P.Y. (2003). Jumping Snakes and parametric model for lip segmentation. In *International Conference on Image Processing (ICIP'03),Barcelona, Spain*.

[5] Fonseca, C., Fleming, P. (1995). An overview of evolutionary algorithms in multiobjective optimisation. *in Evolutionary Computation*. 3(1): 1-16.

[6] Hadziavdic, V. (1999). *A comparative study of active contour models for boundary detection in brain images*. MSc thesis in applied physics. University of Bergen, University of Tromsø, Norway, 1999.

[7] Kass, M., Witkin, A., Terzopoulos, D. (1988). Snakes : Active contour models. *International Journal of Computer Vision*. 321-331.

[8] Perrin, D., Smith, C. (2001). Rethinking classical internal forces for active contour models. *In IEEE International Conference on Computer Vision and Pattern Recognition*.

[9] Pigeon, S. (1996). *M2VTS*. www.tele.ucl.ac.be/PROJECTS/-M2VTS/m2fdb.html.

[10] Precioso, F., Barlaud, M., Blu, T., Unser, M. (2003). Smoothing B-spline active contour for fast and robust image and video segmentation. *In International Conference on Image Processing, Barcelona*.

[11] Yang, L.R., Albregtsen, F. (1996). Fast and Exact Computation of Cartesian Geometric Moments Using Discrete Greens Theorem. *In Pattern Recognition* vol.29, No. 7, July 1996, pp. 1061-1073.

[12] Séguier, R., Cladel, N. (2003). Multiobjective Genetic Snakes Application on Audio-Visual Speech Recognition. *In EC-VIP-MC 2003, Zagreb, Croatia*.

[13] Tsakonas, A., Dounias, G. (2002). Hybrid Computational Intelligence Schemes in Complex Domains: An Extended Review. *In Proceedings of SETN-02, 2nd Hellenic Conference on Artificial Intelligence*. 494-512.

[14] Deb, K. (2000). A fast elitist non dominated sorting genetic algorithm for multiobjective optimisation : NSGA II. *In Parallel problem solving form nature – PPSN VI, Springer lecture notes in computer science.* pp. 849-858.

[15] Goldberg, D.E. (1989). Genetic Algorithms in Search Optimisation and Machine Learning. in *Addison-Wesley, Reading*, Massachussets.

[16] Jehan-Besson, S. (2003). Modèles de contours actifs basés régions pour la segmentation d'images et de vidéos. *phd thesis, Nice University - Sophia Antipolis*.

An Iterative Method for Euclidean Shape Using MMSE Cameras and Maharanobis Distance

Hiroyasu Sakamoto,
Department of Design
Kyushu University
4-9-1 Shiobaru, Minamiku, Fukuoka,
815-8540 JAPAN
sakamoto@design.kyushu-u.ac.jp

Azusa Kuwahara and Takashi Noyori
Graduate School of Design
Kyushu University
4-9-1 Shiobaru, Minamiku, Fukuoka,
815-8540 JAPAN

Abstract
This paper proposes an iterative 3-D Euclid reconstruction method by using a series of linear camera models and generalised inverse (g-inverse) matrices. The method starts with a conventional linear 3-D reconstruction such as the factorisation method, and iteratively corrects reconstruction error due to nonlinearity of captured images under Maharanobis distance error criterion. The linear camera model reduces reconstruction error by attaining the minimum mean square error (MMSE) between captured image and the image of currently reconstructed shape. The g-inverse matrices also reduce the reconstruction error by compensating for variance and covariance of nonlinear distortion in perspective images of the current shape. This paper also shows a simple design method of the MMSE camera as a technique for camera calibration with results of performance evaluation of the camera model. Numerical simulation results of the proposed method show considerable reduction of 3-D reconstruction error. The proposed method can also be effective for reducing reconstruction error due to other nonlinear image components such as lens distortion.

Keywords : 3-D reconstruction, MMSE camera model, g-inverse, linear framework.

1. Introduction

Among many 3-D reconstruction schemes from 2-D images, methods employing Moore-Penrose (MP) g-inverse matrix[1] and the factorisation method[2] have beneficial property of averaging and reducing reconstruction error due to random noise components in images. These original methods are based on linear camera models such as scaled orthographic or paraperspective cameras[3], and do not deal with nonlinear distortions such as perspective projection and lens distortion. The distortions do not show random property but have particular property or tendency described in the next paragraph. Later, the camera model of the factorisation method has been extended to perspective camera or projective camera at the cost of somewhat decreased effect of random noise cancellation because depth of each feature point of each image is separately estimated [4,5].

Figure 1 shows an example of the tendency or property of nonlinear distortions by the arrows (i.e. different magnitudes and correlations including their direction) in a perspective image (shaded faces with white lines) deviated from an orthographic image (black dotted lines). Because every 3-D reconstruction method requires feature point correspondences between 2-D images, it is very usual that all 2-D images have a common set of visible feature points (i.e. a common aspect) and therefore a common tendency of nonlinearity. The proposed method in this paper will flatten the

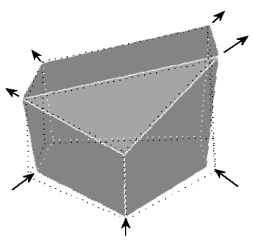


Fig.1. An example of nonlinear components of a perspective image which have magnitude differences and directional correlations. Dimensions of the pentagonal prism in [cm] are 10/6 (max/min H), 10(W) and 7(D).

magnitude differences and will de-correlate these nonlinear components to offer an optimum 3-D reconstruction within the framework of linear processing.

In previous reports, the authors of this paper have proposed a 3-D reconstruction and correction method by using a series of g-inverse matrices with error weighting[6], and have extended the method by employing the MMSE camera models[7]. These methods minimise 3-D reconstruction error criterion measured by the Maharanobis distance.

In this paper, we show further extension of our previous method where it can be incorporated with the factorisation method. In section 2, we formulate the MMSE camera model and show its performance. Section 3 summarises conventional 3-D reconstruction methods. We illustrate the proposed method in section 4 with its algorithm and an estimation method for covariance matrices of nonlinear distortion. In section 5 and 6, we show numerical simulation results of 3-D reconstruction using synthetic images and real images, respectively.

2. The MMSE camera model

2.1. Necessary condition for the MMSE camera

Let $\boldsymbol{V}_k = [X_k, Y_k, Z_k, 1]^{\mathrm{T}}$, ($k = 1, 2, \cdots, K$, and $^{\mathrm{T}}$ is transpose) be homogeneous coordinates of the k-th feature point, and $\boldsymbol{C}$ be 2×4 affine camera matrix, then 2-D image $\boldsymbol{u}_k$ of $\boldsymbol{V}_k$ is given by $\boldsymbol{u}_k = \boldsymbol{C}\boldsymbol{V}_k$. A necessary condition for minimising the MSE $\overline{e^2}$ between $\boldsymbol{u}_k$ and a captured image $\boldsymbol{q}_k = [x_k, y_k]^{\mathrm{T}}$,

$$\overline{e^2} = \frac{1}{K}\sum_{k=1}^{K} \left\| \boldsymbol{u}_k - \boldsymbol{q}_k \right\|^2 , \qquad (1)$$

is given by the following normal equation.

$$\begin{bmatrix} \overline{X^2} & \overline{XY} & \overline{ZX} & \overline{X} \\ \overline{XY} & \overline{Y^2} & \overline{YZ} & \overline{Y} \\ \overline{ZX} & \overline{YZ} & \overline{Z^2} & \overline{Z} \\ \overline{X} & \overline{Y} & \overline{Z} & 1 \end{bmatrix} \boldsymbol{C}^{\mathrm{T}} = \begin{bmatrix} \overline{xX} & \overline{yX} \\ \overline{xY} & \overline{yY} \\ \overline{xZ} & \overline{yZ} \\ \overline{x} & \overline{y} \end{bmatrix}, \quad (2)$$

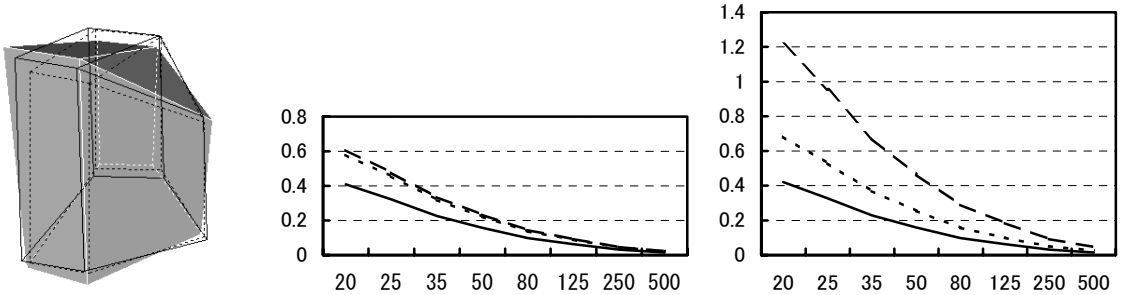
where the bars mean the same averaging operation over $k = 1, 2, \cdots, K$, as Eq.(1).

Once the inverse of the l.h.s. 4×4 matrix in Eq.(2) is calculated, it is easy to obtain $\boldsymbol{C}$ for every captured image. It is possible to regard this method as a camera calibration technique which accounts not so much for physical camera rig but only for captured image and 3-D shape.

2.2. Comparisons with other camera models

Here, we compare 4 camera models with the same magnifying factor 1. Figure 2(a) shows example images of the Fig.1 object captured from a distance 25[cm] by a perspective camera (shaded face with white lines), a paraperspective (dotted lines) and the MMSE (real lines) cameras. Root mean squared error (RMSE) [cm/vertex] from perspective image is 0.76 for paraperspective camera and 0.54 for MMSE camera (71% reduction). More reduction is available by omitting hidden points.

In order to obtain shape independent results, RMSE of very many sets of 10 random dots in



(a) Image examples, gray face is perspective image. (b)MSE[cm] of 10 dots in a cube centred on the optical axis at distances 20~500[cm]. (c) The same result as (b) but the cube edge is on the optical axis.

Fig.2 Comparisons between the MMSE camera( ) with orthographic(— —), paraperspective(---) cameras.

10^3 [cm^3] cube is obtained up to 3 decimal digits precision and is depicted in Fig.2(b),(c). They show RMSE vs. the distance between camera and the cube centre, where in Fig.(b) and (c) the cube centre is 0[cm] and 5[cm] off the optical axis, respectively. Error of the MMSE camera is about 1/3 smaller than paraperspective camera without error increase as the cube goes off the optical axis.

3. Summary of conventional reconstruction methods

3.1. Method using MP g-inverse matrix

Relative camera motion (i.e. angles and distances) is calculated using K (≥ 4) feature points' coordinates $\boldsymbol{v}_{m,k}$ of M (≥ 3) captured images, (m=1,2,$\cdots$,M; k=1,2,$\cdots$,K) with known feature point correspondences to obtain 2×4 camera matrices $\boldsymbol{R}_m$. Using 3-D point coordinates $\boldsymbol{V}_k$, we have

$$\boldsymbol{v}_k = \boldsymbol{R}\,\boldsymbol{V}_k, \qquad \boldsymbol{v}_k = \begin{bmatrix} \boldsymbol{v}_{1,k} \\ \vdots \\ \boldsymbol{v}_{M,k} \end{bmatrix}, \quad \boldsymbol{R} = \begin{bmatrix} \boldsymbol{R}_1 \\ \vdots \\ \boldsymbol{R}_M \end{bmatrix}. \tag{3}$$

The MP g-inverse matrix $\boldsymbol{R}^+ = (\boldsymbol{R}^{\mathrm{T}}\boldsymbol{R})^{-1}\boldsymbol{R}^{\mathrm{T}}$ brings about the MMSE Euclidian 3-D reconstruction.

$$[\boldsymbol{V}_1,\cdots,\boldsymbol{V}_K] = \boldsymbol{R}^+ [\boldsymbol{v}_1,\cdots,\boldsymbol{v}_K], \tag{4}$$

3.2. The factorisation method

The original factorisation method decomposes an observation matrix $\boldsymbol{Q}$ into an approximated product of affine shape matrix $\boldsymbol{S}_{\mathrm{A}}$ and motion matrix $\boldsymbol{M}_{\mathrm{A}}$ by using singular value decomposition (SVD) and truncating its rank to four largest positive singular values σ_1,..., σ_4 and eliminating the others[3].

$$\boldsymbol{Q} \cong \boldsymbol{U}\,\boldsymbol{\Sigma}\,\boldsymbol{V}^{\mathrm{T}} = (\boldsymbol{U}\boldsymbol{\Sigma}^{1/2})\,(\boldsymbol{\Sigma}^{1/2}\,\boldsymbol{V}^{\mathrm{T}}) = \boldsymbol{M}_{\mathrm{A}}\boldsymbol{S}_{\mathrm{A}}, \tag{5}$$

where $\boldsymbol{\Sigma}$ = diag(σ_1,$\cdots$,σ_4,0,$\cdots$,0). In order to modify $\boldsymbol{S}_{\mathrm{A}}$ and $\boldsymbol{M}_{\mathrm{A}}$ to Euclidian shape matrix $\boldsymbol{S}_{\mathrm{E}}$ and camera motion matrix $\boldsymbol{M}_{\mathrm{E}}$, a regular matrix $\boldsymbol{A}_{\mathrm{R}}$ is determined so that each pair of camera coordinate vectors satisfies orthonormal condition and the centre of the object's gravity is at the origin.

$$\boldsymbol{Q} \cong \boldsymbol{M}_{\mathrm{A}}\boldsymbol{S}_{\mathrm{A}} = (\boldsymbol{M}_{\mathrm{A}}\boldsymbol{A}_{\mathrm{R}})(\boldsymbol{A}_{\mathrm{R}}^{-1}\boldsymbol{S}_{\mathrm{A}}) = \boldsymbol{M}_{\mathrm{E}}\boldsymbol{S}_{\mathrm{E}}. \tag{6}$$

In order to obtain absolute reconstruction error from the true shape, metric reconstruction[8] is carried out by an Euclidian transformation matrix $\boldsymbol{A}_{\mathrm{S}}$ to have the following camera matrix $\boldsymbol{M}_{\mathrm{S}}$ and shape matrix $\boldsymbol{S}_{\mathrm{S}}$.

$$\boldsymbol{Q} \cong \boldsymbol{M}_{\mathrm{E}}\boldsymbol{S}_{\mathrm{E}} = (\boldsymbol{M}_{\mathrm{A}}\boldsymbol{A}_{\mathrm{S}}^{-1})(\boldsymbol{A}_{\mathrm{S}}\boldsymbol{S}_{\mathrm{A}}) = \boldsymbol{M}_{\mathrm{S}}\boldsymbol{S}_{\mathrm{S}}. \tag{7}$$

The factorisation method is extended to include projective transformation and perspective camera[4,5]. Because depth for each feature point in each image must be determined, the beneficial effect of random noise cancellation would become weaker.

4. The proposed method

4.1. Formulation of the method

By using g-inverse matrices, we can flatten and decorrelate aspect specific property of nonlinear distortions (like arrows in Fig.1) to reduce total reconstruction error. In order to introduce covariance of feature points, we use $\boldsymbol{v} = \left[\boldsymbol{v}_1^{\mathrm{T}} \;\cdots\; \boldsymbol{v}_K^{\mathrm{T}}\right]^{\mathrm{T}}$, $\boldsymbol{V} = \left[\boldsymbol{V}_1^{\mathrm{T}} \;\cdots\; \boldsymbol{V}_K^{\mathrm{T}}\right]^{\mathrm{T}}$. Then for all k, Eq.(3) is

$$\boldsymbol{v} = \boldsymbol{B}\,\boldsymbol{V}, \tag{8}$$

where $\boldsymbol{B}$ is a block diagonal matrix obtained by arranging $\boldsymbol{R}$ in main diagonal blocks for K times. Here, all elements $\{\boldsymbol{R}_m\}$ of $\boldsymbol{R}$ should be superseded by the MMSE cameras $\{\boldsymbol{C}_m\}$. In our previous paper[6], we adopted the exact $\boldsymbol{R}$ of Eq.(3).

Because real images $\overline{\boldsymbol{v}}$ usually carry observation error $\boldsymbol{e}$, we modify Eq.(8) to $\boldsymbol{e} = \overline{\boldsymbol{v}} - \boldsymbol{B}\boldsymbol{V}$ and introduce a weighting matrix $\boldsymbol{W}$ to form a criterion for the reconstruction error.

$$\boldsymbol{J}(\boldsymbol{V}) = \boldsymbol{e}^{\mathrm{T}}\boldsymbol{W}\,\boldsymbol{e}. \tag{9}$$

When the matrix $\boldsymbol{W}$ is an inverse of covariance matrix of $\boldsymbol{e}$, $\boldsymbol{J}(\boldsymbol{V})$ represents the Maharanobis distance and the effects of flattening and de-correlating the reconstruction error are available.

A necessary condition for minimising Eq.(9) is given by,

$$\boldsymbol{V} = (\boldsymbol{B}^{\mathrm{T}} \boldsymbol{W} \boldsymbol{B})^{-1} \boldsymbol{B}^{\mathrm{T}} \boldsymbol{W} \bar{\boldsymbol{v}} . \qquad (10)$$

If $\boldsymbol{W}$ is set to identity matrix, the solution of Eq.(10) is equivalent to that of section **3.1**.

4.2. An iterative reconstruction algorithm

Using the formulation of **4.1**, the following 6 steps algorithm is devised.

1) In the first step, calculate 3-D reconstruction $\boldsymbol{V}^{[1]}$ by using the MP g-inverse method of **3.1** where $\boldsymbol{R}_m$ should be obtained by the conventional manner[1], or by using the factorisation method where $\boldsymbol{R}_m$ is simultaneously obtained as $\boldsymbol{M}_{\mathrm{E}}$ or $\boldsymbol{M}_{\mathrm{S}}$ of **3.2**.
2) At the j-th step, compute MMSE camera model $\boldsymbol{C}^{[j]}$ from the shape $\boldsymbol{V}^{[j-1]}$ of the former step and the captured images by perspective cameras.
3) Detect nonlinear distortion of perspective images synthesised by perspective cameras whose optical axes have the same directions as $\boldsymbol{C}^{[j]}$ of step 2. Regarding image observation error as the sum of the nonlinear distortion and random noise, estimate variance-covariance matrix $\boldsymbol{D}^{[j]}$ of the error. Set the weighting matrix $\boldsymbol{W}^{[j]}$ to the inverse of $\boldsymbol{D}^{[j]}$.
4) Obtain updated shape $\boldsymbol{V}^{[j]}$ by using g-inverse matrix of Eq.(10) constructed from $\boldsymbol{C}^{[j]}$ and $\boldsymbol{W}^{[j]}$.
5) If $\| \boldsymbol{V}^{[j]} - \boldsymbol{V}^{[j-1]} \| / \| \boldsymbol{V}^{[j]} \| \leq \varepsilon$ (ε is tolerance of conversion), terminate the algorithm.
6) Increase j by 1, and go to the step 2.

The estimation method for nonlinear distortion component in the step 3) is one of the most important keys in our algorithm. For the best precise estimation, we employ orthogonal projection matrices to the complementary subspace of the 3-D reconstruction $\boldsymbol{V}^{[j]}$ of the current step. The matrices can be constructed mainly from three different coordinate vectors of feature points generated by linear cameras with Gram-Schmidt orthogonalisation technique. For details of the orthogonal projection matrices, see the reference [7].

4.3. Estimation method for covariance matrix of nonlinear distortion

In order to estimate covariance matrix of nonlinear distortion, we use the following procedure. The 3-D shape $\boldsymbol{V}^{[j]}$ is rotated on each camera's optical axis, and at each angle $2\pi\, n/N$ [rad] (n = 1,...,N) of rotation, a perspective image is generated by using the camera matrix $\boldsymbol{R}_m$ or $\boldsymbol{C}_m$ of **4.1**. Direction of the optical axis is determined by a vector product of rotation part (i.e. left 2×3 part) of $\boldsymbol{R}_m$ or $\boldsymbol{C}_m$. When the factorisation method is used for the first iteration, each 2×3 part of $\boldsymbol{M}_{\mathrm{S}}$ decides the optical axis. The depth of each point measured along the optical axis gives perspective image.

Multiplying the orthogonal projection matrix of **4.2** to the perspective image data, nonlinear component $\boldsymbol{e}_n$ is extracted. Covariance matrix is estimated by

$$\boldsymbol{U} = \frac{1}{N} \sum_{n=1}^{N} \boldsymbol{e}_n \boldsymbol{e}_n^{\mathrm{T}} + c\, \boldsymbol{I} . \qquad (11)$$

where the 2-nd term stands for random noise component and c its variance and $\boldsymbol{I}$ is identity matrix. In Eqs.(9),(10), we set $\boldsymbol{W} = \boldsymbol{U}^{-1}$ in order to obtain Maharanobis error measure.

5. Numerical simulations by synthetic images

In this section, all 3-D reconstructions are obtained from M = 20 synthetic images from random camera positions giving the same viewing aspect of the object in Fig.1. Some error measures are given in equivalent image pixel size where entire image plane is 640×480 pixels.

5.1. Models of error in image analysis

We pick up two types of error : (i) feature point detection error in $\boldsymbol{v}_k$ and (ii) camera angle estimation error in $\boldsymbol{R}_m$ and $\boldsymbol{M}_{\mathrm{E}}$ or $\boldsymbol{M}_{\mathrm{S}}$, and suppose that their probability distributions are uniform (for point error) and normal (for angle error), respectively. In the following subsections, we will invest two cases : (I) two types of error vary independently and (II) they vary dependently.

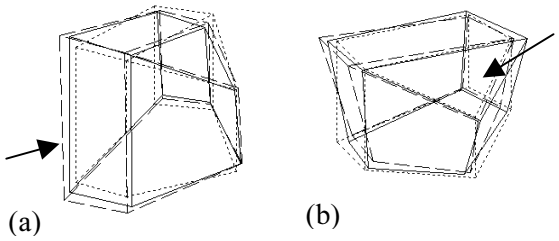


Fig.3. Orthographic images of reconstructed 3-D shapes by 20 perspective images captured from the directions of arrows at the distance 28[cm]. Mean error[cm/vertex] of 3-D shape by the former method [6] (— —) and the proposed method (—) from the true shape (---) are : (a) 0.92 and 0.63 (68% reduction); (b) 0.92 and 0.72 (79%), respectively.

5.2. Two examples

Here, the range of error (i) is set to ±0.5 pixel, and the error (ii) has standard deviation (SD) of 2 degrees. Figure 3(a),(b) show 3-D reconstructed shapes by the proposed method and the former method [6]. It is clear that significant error reduction is available by the proposed method. If the hidden points were omitted, better camera calibration and reconstruction results are available.

5.3. Systematic experiments

In this subsection, we synthesise 500 random sets of $M = 20$ synthetic images of the same aspect of the object in Fig.1 captured from a sphere of radius r [cm] and obtain statistical (averaged) results.

5.3.1. Comparisons with a conventional method

Figure 4 shows averaged reconstruction error [cm/vertex] of the proposed method (—) and the conventional (g-inverse) method [1] (— —) vs. the radius r [cm]. In Fig.4(a), error model (I) is employed, where the point error (i) is fixed to ±0.5 pixel and the angle error (ii) has SD of 2(curve □), 4(curve ×) and 6(curve Δ) degrees, respectively. In Fig.4(b), we use error model (II), where error(i) and error(ii) have the relation that (ii) = 2×(i). The pair of (error(i), error(ii)) is set to (±0.5, ±1)(□), (±1.5, ±3)(×) and (±2.5, ±5)(Δ), respectively.

5.3.2. Comparisons with the former method

In Fig.5(a), we compare averaged 3-D reconstruction error [cm/vertex] by the proposed method (—) and the former method (— —) [6]. Employed noise model and noise level settings are the same as Fig.4(b). Fig.5(b) shows averaged iteration counts required to obtain the results of Fig.5(a).

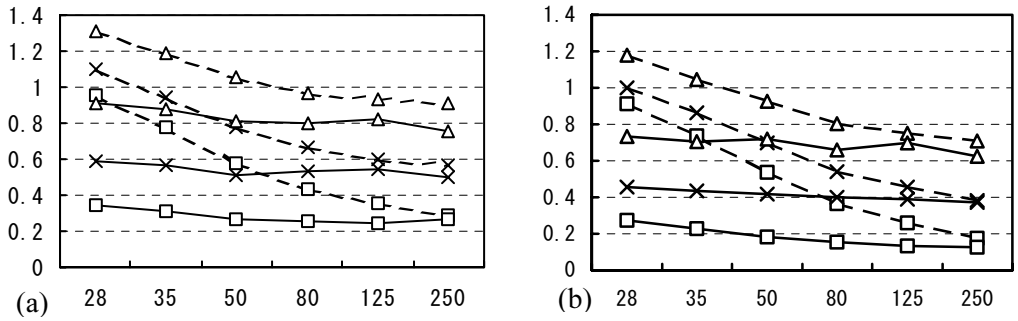


Fig.4. Mean reconstruction error [cm/vertex] vs. distance r [cm] between camera and object.

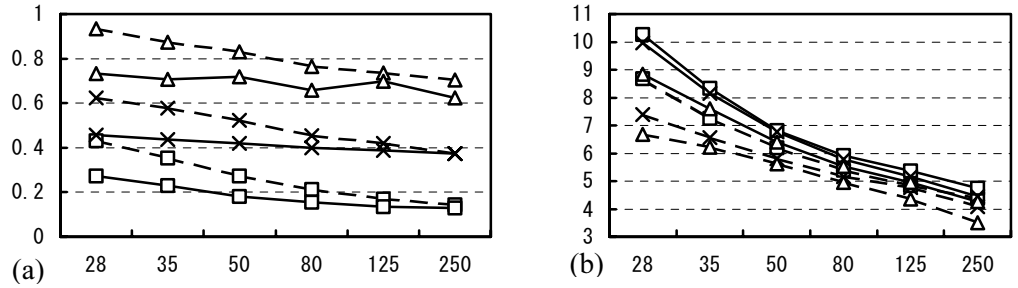


Fig.5 (a). Comparisons with the former method, (b) averaged iteration counts.

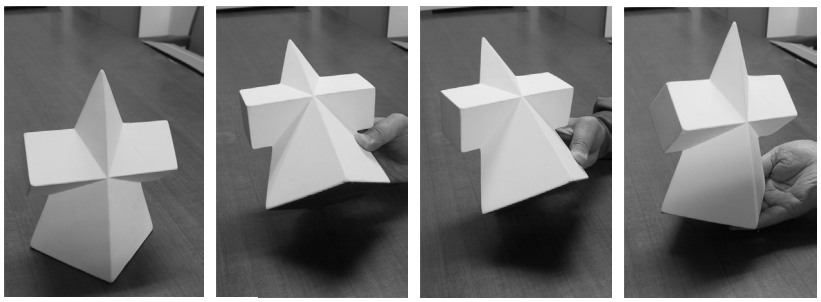


Fig.6. Real images of the object employed (height = 15[cm], bottom =10^2[cm^2] square) .

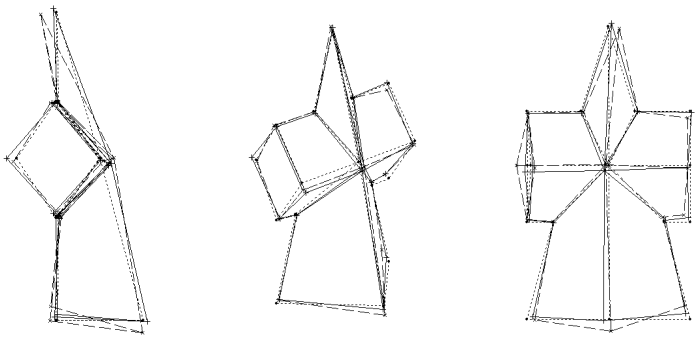


Fig.7. Three views of a metric reconstruction. The true 3-D shape(---), reconstruction by the factorisation method[2] (— —) and by the proposed method (real line). Broken lines show bended peak and bottom parts due to nonlinear distortions of perspective images of this aspect.

6. Numerical simulation by real images

Here, we capture $M = 33$ real images of Fig.6. Using the factorisation method[2], 3-D shape is obtained by Eq.(7) in the step $j = 1$ of **4.2**. 3-D shape of this step is depicted by broken lines in Fig.7, where construction error is 0.8100[cm/vertex]. The true shape is drawn by dotted lines.

Figure 8 shows an error converging process of the proposed method. Only 4 iterations are required for the 4 digit convergence to 0.5433[cm/vertex]. Error reduction rate is about 2/3.

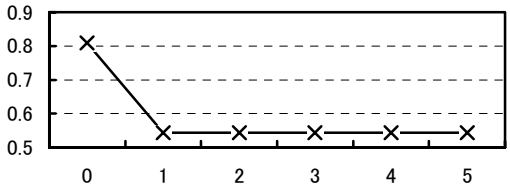


Fig.8. An example of error convergence property of the proposed iterative method. Ordinate : averaged reconstruction error [cm/vertex], abscissa : iteration count.

7. Conclusions

In this paper, an iterative method is proposed for reconstructing 3-D shape from many 2-D images. Starting with a conventional reconstruction method, the method employs a series of the MMSE linear camera models and a series of g-inverse matrices with Maharanobis distance. The numerical simulation results show that the proposed method reduces reconstruction error significantly.

The merit of the proposed method is that the method is effective not only to perspective nonlinear distortion but also to other nonlinear effects such as lens distortion.

References

[1] Xu, G. and Sugimoto, N. (1999). A linear algorithm for motion from three weak perspective images using Euler angles. *IEEE Trans. Pattern Anal. Machine Intell.*, 21:54-57.

[2] Tomasi, C. and Kanade, T. (1992). Shape and motion from image stream under orthography: a factorization method. *Int'l J. Computer Vision*, 9:137-154.

[3] Poleman, C. J. and Kanade, T. (1994). A paraperspective factorization method for shape and motion recovery. *Computer Vision — ECCV 94, Proc. 3-rd Conf. Computer Vision,* 2:97-108.

[4] Christy, S. and Horaud, R. (1996). Euclidian shape and motion from multiple perspective views by affine iterations. *IEEE Trans. Pattern Anal. Machine Intell.*, 18:1098-1104.

[5] Heyden, A., Berthilsson, R. and Sparr, A. (1999). An iterative factorization method for projective structure and motion from image sequences. *Image and Vision Computing*, 17: 981-991.

[6] Sakamoto, H. and Nishikawa, Y. (2003). A correction method for 3-D reconstructed shapes by error weighting, (in Japanese). *J. Inst. Image Information and Television Engineers*, 57:142-148.

[7] Noyori, T., Kuwahara, A. and Sakamoto, H. (2004). An iterative 3-D shape reconstruction method using MMSE camera model and error weighted g-inverse. *Proc. FCV2004*, 214-219.

[8] Hartley, R. and Zisserman, A. (2000). *Multiple View Geometry in Computer Vision*. Cambridge UK: Cambridge Univ. Press.

AUTOMATIC SCORING OF THE SEVERITY OF PSORIASIS SCALING

David Delgado*
IMM,
Denmark
email: ddg@imm.dtu.dk

Bjarne Ersbøll
IMM,
Denmark
email: be@imm.dtu.dk

Jens Michael Carstensen
IMM
Denmark
email: jmc@imm.dtu.dk

Abstract

In this work, a combined statistical and image analysis method to automatically evaluate the severity of scaling in psoriasis lesions is proposed. The method separates the different regions of the disease in the image and scores the degree of scaling based on the properties of these areas. The proposed method provides a solution to the lack of suitable methods to assess the lesion and to evaluate changes during the treatment. An experiment over a collection of psoriasis images is conducted to test the performance of the method. Results show that the obtained scores are highly correlated with scores made by doctors. This and the fact that the obtained measures are continuous indicate the proposed method is a suitable tool to evaluate the lesion and to track the evolution of dermatological diseases.

Keywords*: psoriasis, exploratory data analysis, segmentation, decision trees, classification.*

1 Introduction

One of the main problems in the treatment of dermatological diseases is the difficulty of tracking the evolution of the disease. Physicians are visited by the patients several times to control the evolution of the disease. However, due to the fact that no objective methods to summarize the lesion exist, physicians make scorings and take notes to document the actual condition of the patient. A drawback of this method is the dependency on the individual physician.

The advances in image analysis during the last decade have lead to the development of different methods to deal with related problems in the dermatological field. Engström [1] observed the effect of a new enzymatic debrider observing the evolution of the lesion area and the lesion color. Later, Hansen [2], developed an image system that included calibration for increasing the quality of the images. The system diagnoses burns and pressure ulcers in animals but the possibility of being used in humans was mentioned. In a recent paper, Hillebrand [3] used computer analysis in high resolution digital images to compare the skin condition of a group of females.

In this work, a method to objectively score the degree of scaling in psoriasis is proposed. The method realises a hierarchical segmentation to isolate the different structures present in the image. Different values are obtained from these areas and a classification tree is built to correlate these measurements with the doctor scorings.

2 Segmentation of the areas present in the lesion

Psoriasis is a dermatological disease characterized by red, thickened areas with silvery scales. In order to score the degree of scales in psoriasis, the first step is to segment the different areas in the lesion.

*The dermatologists Lone Skov and Bo Bang of Gentofte Hospital of Denmark and the anonymous patients

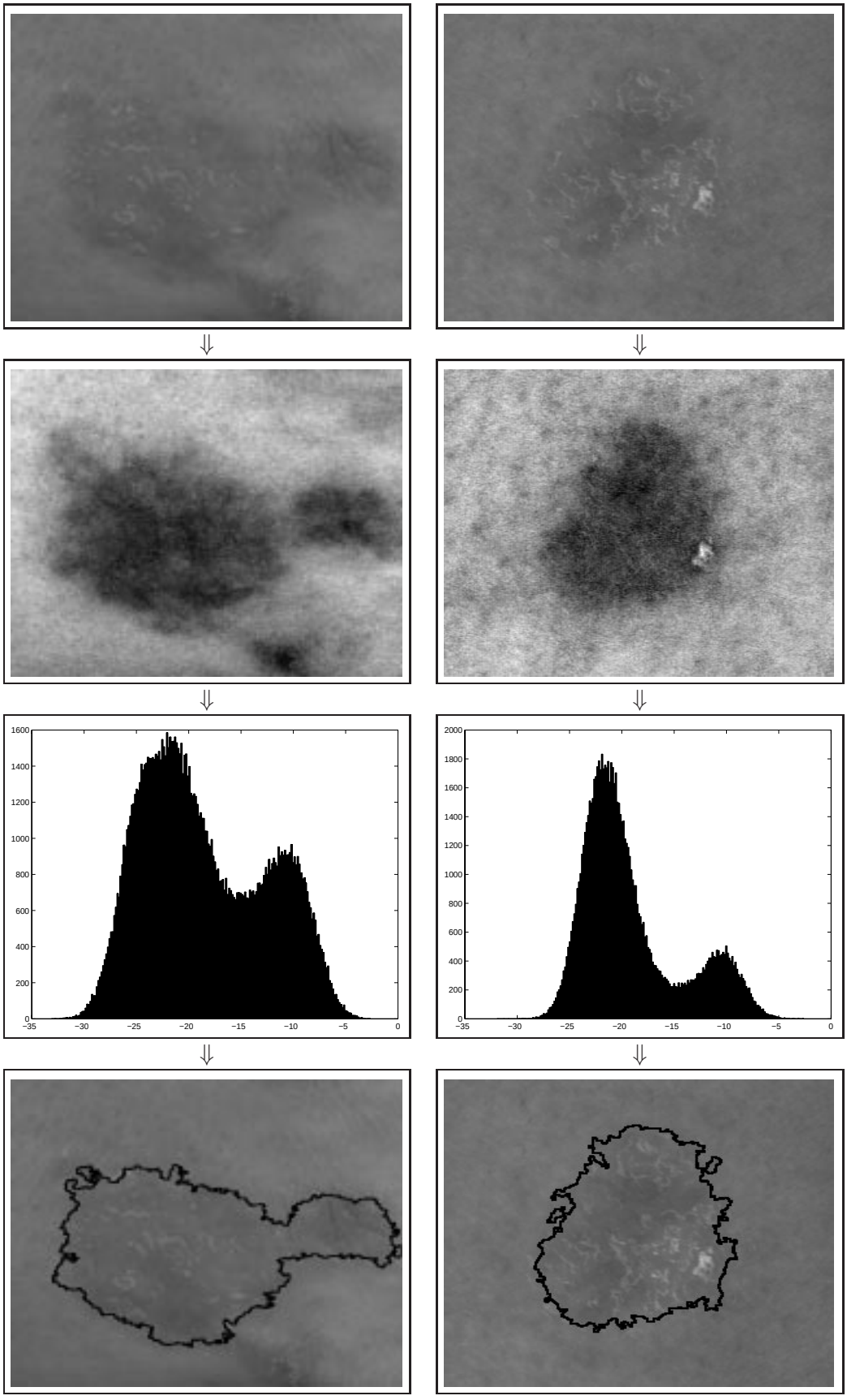


Figure 1: Top row: Two psoriasis images. Second row: Difference between the blue and green bands. Third row: Histogram of the blue minus green bands. Bottom row: Lesion segmentation result.

2.1 Segmentation of the lesion

The segmentation of the disease with respect to the healthy area is based on the assumption, that under a suitable projection, both the normal skin and the lesion are distributed approximately as a Gaussian distribution. This assumption was supported by an exploratory data analysis of a small set of psoriasis images where several projections were considered. Furthermore, a principal component analysis [4] and an independent component analysis [5] on a dataset of 115 images indicated that the difference between the green and the blue band exhibits a good contrast to discriminate between the lesion and the normal skin. The distribution of this difference approximately follows a linear mixture of two Gaussians. The estimation of their means and variances makes it possible to identify the lesion by means of discriminant analysis. The parameters of the gaussians were estimated according to Taxt [6]. Figure 1 shows the segmentation of the lesion.

2.2 Extracting the scales

Segmentation of the scales is complicated by the fact that scales may or may not appear in the image. If they appear they may range from a few spots to a large area. Moreover, non-uniformity of the areas with redness (ranging from red to brown) makes the task even harder. This variability implies that the lesion has to be considered in small areas where the change in redness is not significant. This can be accomplished with watersheds [7] to mark the different scales and then locally use a clustering algorithm to segment them. This approach requires specifying the number of watersheds. In this work, the number of watersheds is determined in two steps. First a new image is created based in the watershed regions. Each watershed area is replaced by the minimum value of this area. This new image is then thresholded and the watersheds with values less than the threshold are the areas where the scales are detected. The method was tested on a set of psoriasis images and it demonstrated a good performance. However, the method had difficulties with some images that had problems during acquisition (especially shadows), so the number of watersheds was not found correctly. To solve this problem, the number of watersheds was fixed visually by a tuning parameter. The blue band was used to find the watersheds because a canonical analysis had shown that this band is the best to separate the scales from the red area. Figure 2 displays the segmentation of the scales.

2.3 Scoring the disease

Once the different areas have been segmented, a decision tree is created to automatically classify the different images approximating the scorings made by the physicians. Three variables are used as input to the model: the area of the scaling, the ratio between the area of scaling and the area of the lesion, and the ratio between the area of scaling and the area of redness. The whole procedure is shown in Figure 3.

2.4 Experiment: Scoring the disease

In collaboration with the dermatological department of Gentofte Hospital in Denmark an experiment was conducted. The goal of the experiment is to objectively score the severity of the scaling in psoriasis images. To accomplish this goal, a set of 46 psoriasis images was selected from a database of psoriasis collected from different patients. The physicians scores of these images was also available. The images were selected to cover the maximal possible diversity. The different areas of each image were extracted according to the procedure described in the previous sections and the above mentioned three summary values were obtained. A cross-validation process was used to build 23 decision trees. These decision trees utilized 44 data points to build the tree and two for testing it. Results showed that the first variable, the area of the scaling, is enough to explain the physicians scoring. The automated scoring with our method has proven reliable, and on several occasions even allowed for corrections of physicians' mistakes. In these cases, the physicians were asked to re-score their previous judgements, and in all cases

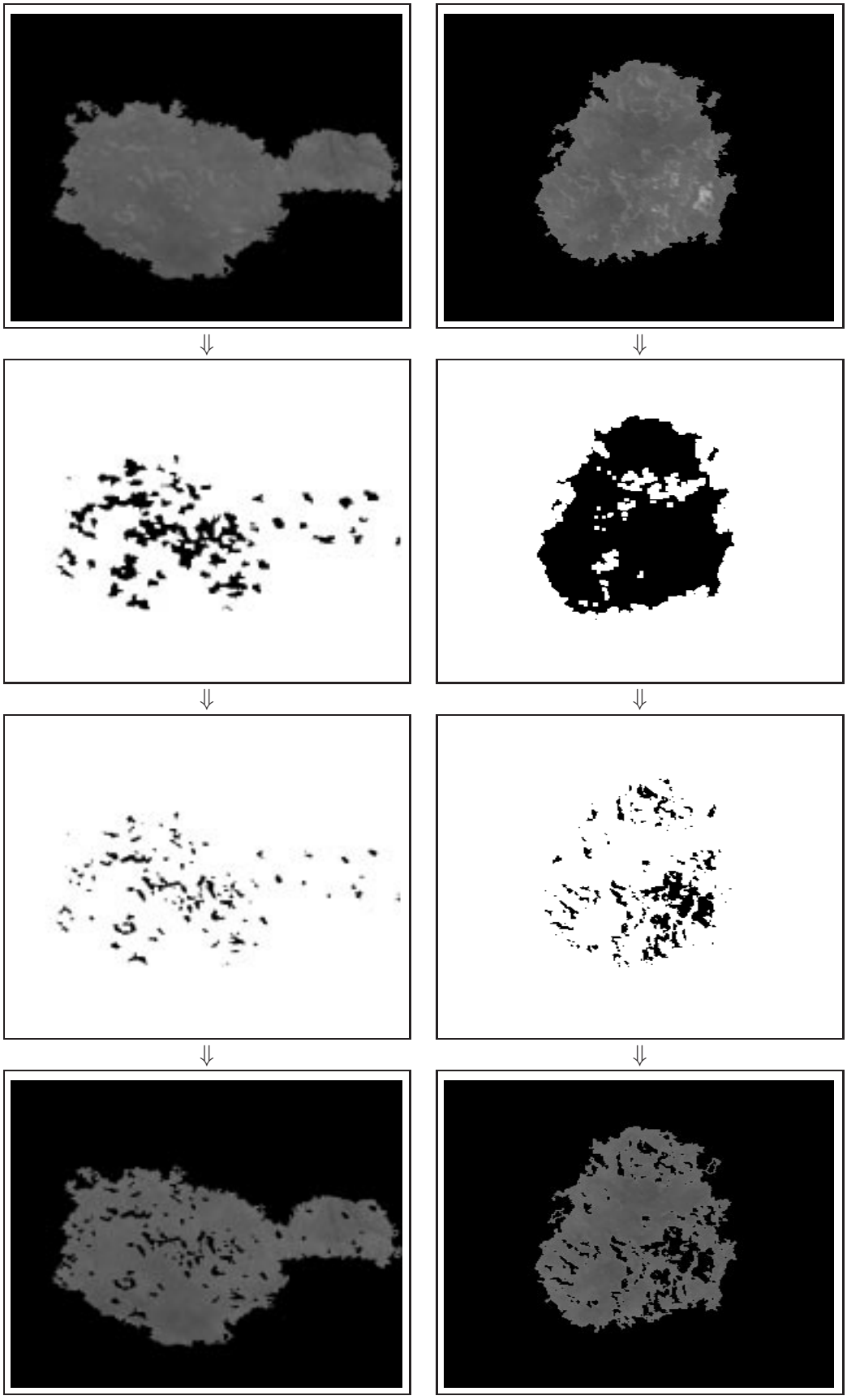


Figure 2: Top row: The original lesion. Second row: Scaling markers. Third row: Scaling segmentation result. Bottom row: A clear display of segmentation on top of the original image.

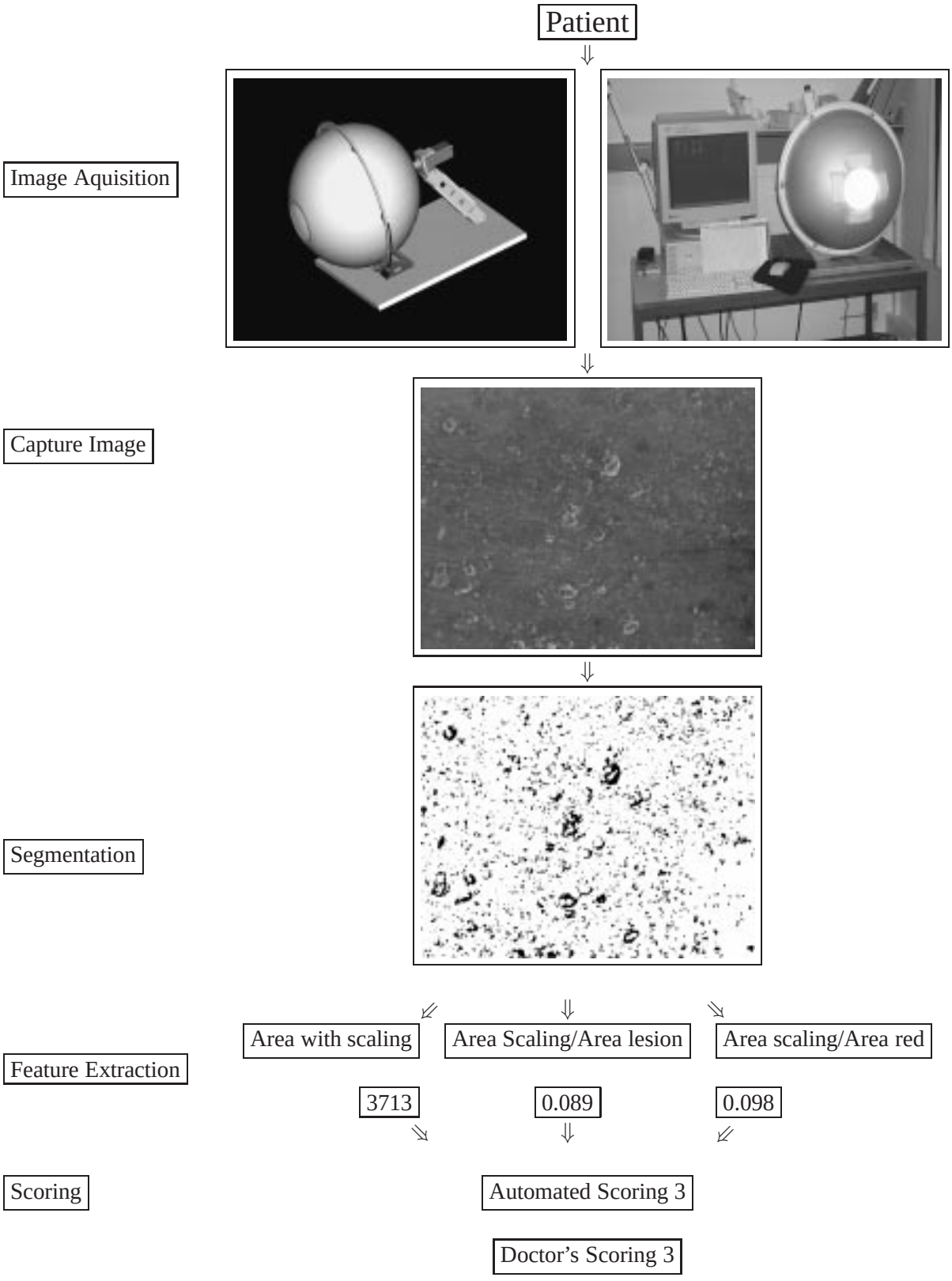


Figure 3: diagram of the method.

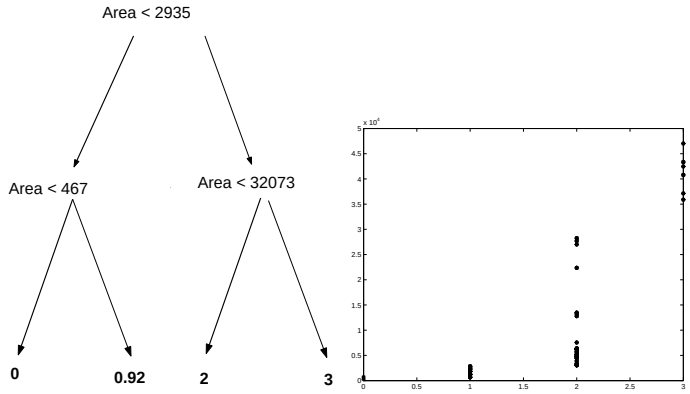


Figure 4: Left: Decision tree for the scoring given the parameters of the segmentation. Right: Dependency of lesion area on physicians' scoring of the lesions

the assessment was changed. Figure 4 left shows the final tree generated using all the points. Figure 4 right plots the area of the scaling versus the physicians' scoring.

3 Summary and conclusion

In this work, a procedure to evaluate the severity of the scaling in psoriasis has been developed. The method automatically separates the different parts and extracts different parameters. In certain difficult cases such as uneven illumination it has been noticed that, allowing a manual interaction increases the accuracy notably. The method provides objective measures that avoid the dependence of the physician in the tracking of dermatological diseases. It has been shown that one of the provided measures is highly correlated with the doctor scoring. Together with the other two measures we expect to be able to provide a better lesion description.

References

[1] Engström N., Hansson F., Hellgren L., Vincent J., Nordin B. and Wahlberg A. Computerized Wound Image Analysis In Pathogenesis of Wound and Biomaterial-Associated Infections, Springer-Verlag, p 189-193, 1990.

[2] Hansen G., Sparrow E., Kokate J., Leland K., Iaizzo P. Wound Status Evaluation using Color Image Processing IEEE Transactions on Medical Imaging, vol. 16, no.1, February 1997

[3] Hillebrand G., Miyamoto K., Schnell B., Ichihashi M., Shinkura R., Akiba S. Quantitative evaluation of skin condition in an epidemiological survey of females living in northern versus southern Japan Journal of Dermatological Science, vol. 27, p 42-52, 2001.

[4] Johnson R., Wichern D. Applied Multivariated Statistical Analysis. Chapter 8. Prentince-Hall, 1995.

[5] Hyv*ä*rinen A., Karhunen J., Oja E. Independent Component Analysis Wiley publications, 2001.

[6] Taxt T., Hjort L., Eikvik L. : Statistical Classification using a Linear Mixture of two Multi-normal Probability Densities. Pattern Recognition Letters.(1991) 12 731-737

[7] Vincent L., Soille P. Watersheds in digital spaces: an efficient algorithm based on immersion simulations, IEEE Trans. Pattern Anal. Mach. Intell. 13 (6) (1991) 593-598.

A Global Analysis of Optical Snow for Arbitrary Camera Motions

Vincent Chapdelaine-Couture[1] and Sébastien Roy[2]

University of Montreal (DIRO), Montreal, Quebec, H3T 1J4 Canada
[1] **email: chapdelv@iro.umontreal.ca**
[2] **email: roys@iro.umontreal.ca**

Abstract

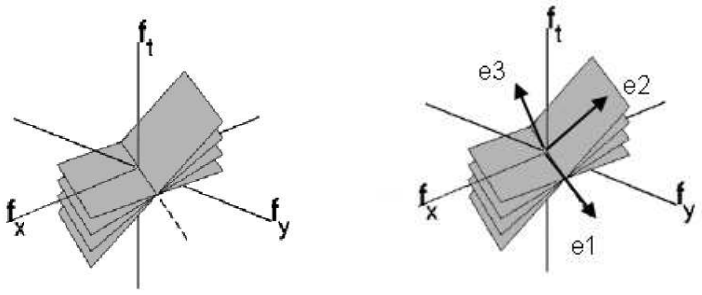
Optical snow, introduced in [4], is a new category of motion for highly cluttered scenes in which no spatial continuity can be assumed. Since no smoothness constraint can be imposed on the velocity field, traditional optical flow methods can no longer be used [1]. However, a model of optical snow has been proposed in [5] and algorithms based on this model were suggested using an analysis in the spatio-temporal frequency domain [5, 2]. This model assumes lateral motion and can be used to solve the 3D camera motion problem by decomposing sequences in sufficiently small patches [7]. We would like to use the same model to find arbitrary camera motions globally instead of using patches. In the present paper, we introduce a complementary model for purely non-lateral optical snow. The standard optical snow model and this complementary form could lead to a new global approach for solving the general egomotion problem. We show how non-lateral optical snow sequences can be rectified such that standard methods to analyze optical snow can be applied. The effectiveness of the method is shown for both real and synthetic sequences.
Keywords*: Optical snow, Egomotion, Fourier transform, Motion analysis, Optical flow*

1 Introduction

Most studies assume a unique velocity at each point in the visual field [1]. This assumption is only valid if the depth map is continuous. If an observer moves in a 3D highly cluttered scene, a forest for instance, this assumption no longer holds; branches and leaves at many depths cause discontinuities in the motion field. Such cases can be solved by a human observer [10]. However, these scenes are hard to solve since feature points cannot be tracked and since traditional optical flow methods cannot be expected to recover an image velocity.

Recently, [4] introduced a new category of movement called optical snow which generalizes optical flow by abandoning assumptions of spatial continuity. A model to analyse optical snow induced by all one-parameter set of velocities has been proposed in [5]. Taking the whole sequence in consideration, this model allows for *lateral observer motions* only [5]. Since the image velocity field can locally be approximated as a sum of two fields, a parallel field due to camera translation and a constant field due to camera rotation [9, 7], it was showed in [7] that this model could be used to recover 3D egomotion by subdividing the image sequence in sufficiently small patches.

We suggest a more global approach without the use of patches. In this paper, we analyze purely *non-lateral observer motions* and present a method to rectify these sequences such that standard optical snow methods can be applied. Finally, we give some experimental results.



(a) (b)

Figure 1: (a) Bowtie signature in the frequency domain (b) Eigenbasis of bowtie signature

2 Previous works

2.1 Optical Snow

The model of optical snow defined in [4, 5] is an extension of the motion plane property [11] which states that an image pattern translating with uniform image velocity produces a plane of energy in the frequency domain. Formally, let $I(x, y, t)$ be a time varying image. If an image patch is translating by (v_x, v_y), we know from [3] that this velocity is constrained by

$$v_x \frac{\partial I}{\partial x} + v_y \frac{\partial I}{\partial y} + \frac{\partial I}{\partial t} = 0 \tag{1}$$

This constraint, transposed in the Fourier domain, yields the following equation:

$$-2\pi * (v_x f_x + v_y f_y + f_t) * \hat{I}(f_x, f_y, f_t) = 0 \tag{2}$$

where $\hat{I}(f_x, f_y, f_t)$ is the Fourier transform of $I(f_x, f_y, f_t)$. As noted in [5], equation 2 implies that all frequencies $\hat{I}(x, y, t) \neq 0$ lie on the plane

$$v_x f_x + v_y f_y + f_t = 0. \tag{3}$$

This model was extended in [4] for the case in which there is a one-parameter set of velocities within an image region, i.e. where velocities vary according the following equation:

$$(v_x, v_y) = (u_x + \alpha t_x, u_y + \alpha t_y) \tag{4}$$

where u_x, u_y, t_x, t_y are constants and α depends on the visible depth at point (x,y) [5].
Substituting Eq. 4 into Eq. 3 yield a family of planes in the frequency domain,

$$(u_x + \alpha t_x) f_x + (u_y + \alpha t_y) f_y + f_t = 0. \tag{5}$$

Thus, this model produces a set of planes in the frequency domain. This set of planes forms a *bowtie* (see Fig. 1) and follows the two following propositions:

Proposition 1: The planes of the bowtie intersect at a common line, called the *axis of the bowtie*, that passes through the origin.

Proposition 2: The axis of the bowtie is in direction $(-t_y, t_x, u_x t_y - u_y t_x)$[1]. By normalizing (t_x, t_y), this direction becomes $(-t_y, t_x, |U|sin(\phi))$, where ϕ is the angle between vectors (t_x, t_y) and (u_x, u_y).

Applied to the egomotion problem, it was noted in [5] that Eq. (4) corresponds to

$$(v_x, v_y) = (-\omega_y + \alpha t_x, \omega_x + \alpha t_y) \tag{6}$$

In other words, camera rotation generates a constant velocity component $(-\omega_y, \omega_x)$ for small field of views ($\pm 20°$) and lateral translation (t_x, t_y) generates a velocity inversely proportional to depth. Components ω_z and t_z were assumed to be 0. Therefore, this model cannot recover arbitrary camera motions.

The main contribution of this paper is to show that complementary camera motions, i.e. following components ω_z and t_z (all other components assumed to be 0), produce non-lateral optical snow sequences that can be rectified to be analyzed by existing optical snow methods. We will describe how to find components ω_z and t_z.

3 Motion Field

The motion field equation contains 7 variables which are the depth P_z at each pixel, the translation vector (t_x, t_y, t_z) and the rotation vector $(\omega_x, \omega_y, \omega_z)$.

More precisely, the velocity field for a pixel (x,y), as defined in [6], is:

$$(v_x, v_y)^T = \begin{pmatrix} p_x p_y \omega_x - (1 + p_x^2)\omega_y + p_y \omega_z + \frac{p_x t_z}{P_z} - \frac{t_x}{P_z} \\ (1 + p_y^2)\omega_x - p_x p_y \omega_y - p_x \omega_z + \frac{p_y t_z}{P_z} - \frac{t_y}{P_z} \end{pmatrix}$$

By assuming that only P_z, t_z and ω_z are non-zero, we are left with:

$$(v_x, v_y)^T = \begin{pmatrix} p_y \omega_z + \frac{p_x t_z}{P_z} \\ -p_x \omega_z + \frac{p_y t_z}{P_z} \end{pmatrix}$$

Note that rotation component ω_z is perpendicular to translation component t_z for each pixel (x,y), and that ω_z is independent of depth. From this observation, rectification is performed on image sequences to obtain optical snow motion $(0, \omega_z) + \alpha(t_z, 0)$.

3.1 Motion Field Rectification

The rectification is a polar transformation around the FOE [8] which is the image center in our case. First, we rectify the motion field induced by pure ω_z rotation. Consider a point p located in the image at (1,0) on the camera projection plane (see Fig. 2-a). The path followed through time by p for a pure ω_z movement is a circle of radius 1, and its speed is exactly equal to $|\omega_z|$. The rectification is done by "unfolding" images such that this circle becomes straight and vertical, as shown in Fig. 2-b. For a field of view of 90° and images of size $N \times N$, the vertical length of the rectified image is $2\pi \frac{N}{2} = \pi N$ pixels. Note that flow lines of a pure forward motion become horizontal. The velocities in the rectified motion field correspond to $(v'_x, v'_y)^T = (\alpha t_z \sqrt{p_x^2 + p_y^2}, \omega_z)^T$.

Standard optical snow produces velocites that only depend on depth. However, the velocities in rectified non-lateral motion sequences vary according to depth and to image position. Therefore, horizontal lines in the rectified sequence must be resampled according to factor $\frac{1}{\sqrt{p_x^2 + p_y^2}}$, where ($p_x$,$p_y$) is the position of a pixel on the camera projection plane in the original sequence. In theory, we get infinite sampling at center (0,0). In practice, we do not rectify near the image center as illustrated in Fig. 2-c. Notice that image corners are cut to remove empty spaces in the rectified sequence.

[1]For simplicity, all equations in this paper assume image sequences of equal dimension in space and time.

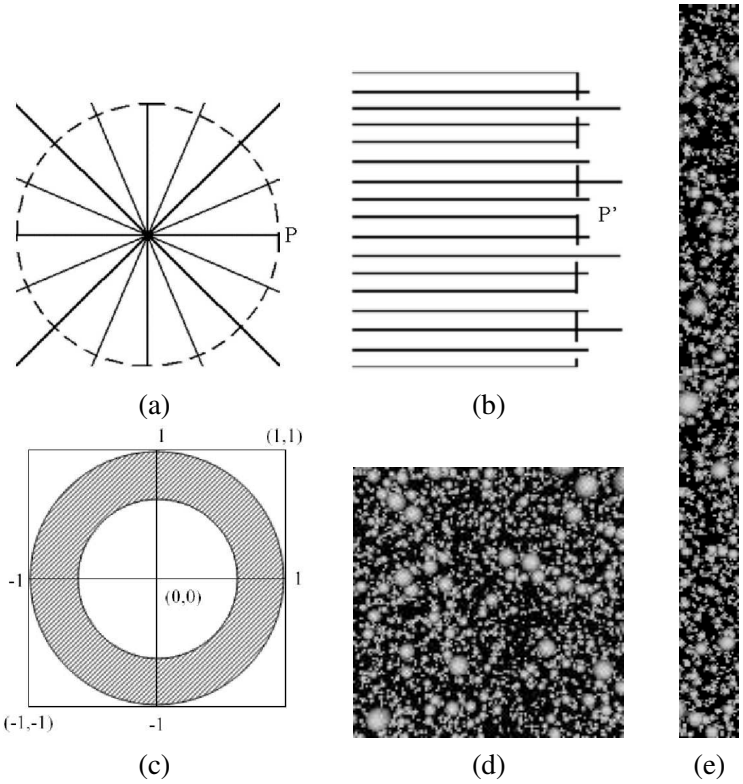


Figure 2: (a) Original motion field (b) Rectified motion field (c) Rectified region of the original sequence (d) Original frame (e) Rectified frame

3.2 Finding rotation ω_z

The bowtie axis can be computed from the rectified sequence using Principal Components Analysis as described in [2]. For standard optical snow, the angle ϕ in the bowtie axis equation (see Proposition 2) is unknown. For non-lateral optical snow, however, we know that $\phi = 90°$. Hence, the bowtie axis equation becomes $(0, 1, \omega_z)$. Thus, the third component of the bowtie axis gives us ω_z directly with speed given in pixels/frame in the rectified sequence. The rotation given in degrees, for a field of view of $90°$, is then $360\frac{\omega_z}{\pi N}$.

The bowtie axis can also be found using the best fit plane [2]. Let (n_x, n_y, n_z) be the normal of the best fit plane π. Since t_z is the only component affected by depth, the bowtie axis is the line on the best fit plane in direction $(0, 1, -\frac{n_y}{n_z})$.

3.3 Finding translation t_z

From Eq. 5, and since t_z generates only horizontal velocities and ω_z only vertical velocities, planes forming the bowtie have equation $(\alpha t_z, \omega_z, 1)$. The best fit plane π is defined as $(\frac{n_x}{n_z}, \frac{n_y}{n_z}, 1) = (\overline{\alpha} t_z, \omega_z, 1)$, where $\overline{\alpha}$ is the weighted "average" of the slopes of the motion planes that compose the bowtie. The weights depend on depth distribution in the scene as well as image contrast contributed by each object. Since this information is unknown, we can only compute t_z up to a scale factor $\overline{\alpha}$.

4 Experimental results

To evaluate our method, we rendered several synthetic image sequences of scenes containing lambertian spheres (see Fig. 3-a). Image motion was generated by moving a camera ($90°$ field of view) through the scene with various t_z and ω_z parameters.

Table 1 shows results for various rectified sequences. The last two rows correspond to real image sequences, respectively the lab sequence (see Fig. 3-b) and the plants sequence (see Fig. 3-c). The ω_z component is found almost exactly. The running time is about 1.6 seconds for 128x128x32 sequences on a 1.3GHz AMD Athlon machine.

(a) (b) (c)

Figure 3: (a) Synthetic scenes were constituted of small balls at different depths (64 frames of size 128x128 pixels). (b) Lab sequence taken from a camera rotating around the z-axis (40 frames of 128x128 pixels). (c) Plants sequence taken from a camera making a forward motion (32 frames of 128x128 pixels).

Table 1 : Results for synthetic and real scenes

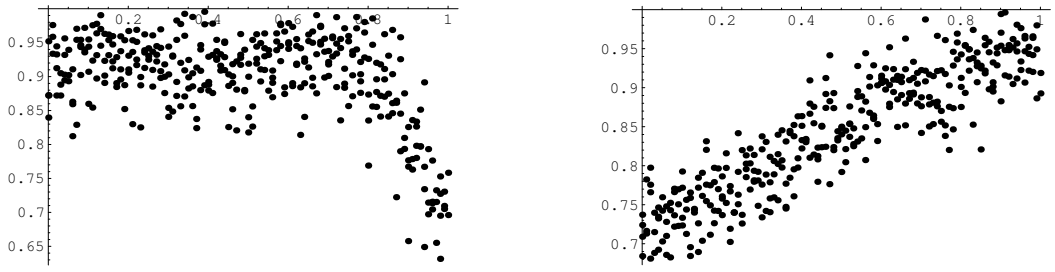
True Rotation (degrees/frame)	**True Translation** (pixel/frame)	**Rotation Found**	**Translation Found** (up to a scale factor)
0.00	1.00	0.00	0.15
1.80	0.00	1.78	0.00
1.80	1.00	1.81	0.22
≈ 0.50	≈ 0.00	0.35	-0.05
≈ 0.00	≈ 0.80	0.00	0.31

4.1 Comparing the eigenvalues

When analyzing image sequences globally, we would like to estimate how well the motion fits the optical snow model, i.e. if a bowtie is present. For instance, an unrectifed forward motion or a rectified lateral motion do not produce a bowtie signature. For such cases, the motion field features velocities oriented in all directions. Detecting these situations would be a great benefit.

In [2], depth range is evaluated by comparing the two largest eigenvalues of the Principal Components Analysis method. We can adapt this measure to detect the presence of a bowtie. The ratio of the first two eigenvalues λ_2 and λ_1 has a maximum of 1 and should decrease as we get closer to a bowtie signature. In fact, the presence of a bowtie creates a high power concentration along its axis which increases the first eigenvalue. The absence of a bowtie is caracterized by a uniform power distribution which makes λ_1 and λ_2 almost equal. Fig. 4-a shows a plot of $\frac{\lambda_2}{\lambda_1}$ as a function of $\frac{|t_z|}{|T|}$ for rectified sequences. As expected, the ratio falls off as the bowtie takes shape in the frequency domain. Fig. 4-b shows $\frac{\lambda_2}{\lambda_1}$ as a function of $\frac{|t_z|}{|T|}$ for non-rectified sequences. As expected, it increases linearly up to 1 as bowtie signature disappears. The sequences used for these graphs have random translation vectors of unit length.

Notice that the curve in Fig. 4-a decreases non-linearly while the other one is linear. This might be caused by the subsampling during rectification which accentuates the effect of any lateral motion. This is still under investigation. These curves, if modelled correctly, would allow merging non-lateral and lateral motions analysis into a general egomotion estimation algorithm.



(a) (b)

Figure 4: (a) $\frac{\lambda_2}{\lambda_1}$ as a function of $\frac{|t_z|}{|T|}$ for rectified sequences. It starts at 1 and decreases as a bowtie signature takes shape. (b) $\frac{\lambda_2}{\lambda_1}$ as a function of $\frac{|t_z|}{|T|}$ for non-rectified sequences. It increases linearly up to 1 as bowtie signature disappears.

5 Conclusion

This paper presented a method to analyze purely non-lateral optical snow by introducing a rectification process. Results show its accuracy and efficiency. We hope to solve the general egomotion problem in a global approach using this scheme.

References

[1] J.L. Barron, D.J. Fleet, and S.S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–77, February 1994.

[2] V. C.-Couture, S. Roy, M. S. Langer, and R. Mann. Principal components analysis of optical snow. In *British Machine Vision Conference*, September 2004.

[3] B. Horn and B. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[4] M. S. Langer and R. Mann. Dimensional analysis of image motion. In *IEEE International Conference on Computer Vision*, pages 155–162, 2001.

[5] M.S. Langer and R. Mann. Optical snow. *International Journal of Computer Vision*, 55(1):55–71, 2003.

[6] H.C. Longuet-Higgins and K. Prazdny. The interpretation of a moving retinal image. *Proceedings of the Royal Society of London B*, B-208:385–397, 1980.

[7] R. Mann and M. S. Langer. Estimating camera motion through a 3d cluttered scene. In *Canadian Conference on Computer and Robot Vision*, London, Canada, May (to appear) 2004.

[8] M. Pollefeys, R. Koch, and L. Van Gool. A simple and efficient rectification method for general motion. In *International Conference on Computer Vision*, pages 496–501, Corfy, Greece, 1999.

[9] J. H. Rieger and D. T. Lawton. Processing differential image motion. *J. Opt. Soc. Am. A*, 2:254–260, 1985.

[10] W.H. Warren and D.J. Hannon. Eye movements and optical flow. *Journal of the Optical Society of America A*, 7(1):160–169, 1990.

[11] A. Watson and A. Ahumada. Model of human visual-motion sensing. *Journal of the Optical Society of America*, 2(2):322–342, 1985.

DIRECTION OF CAMERA BASED ON SHADOWS

Darren Caulfield
Department of Computer Science
Trinity College
Dublin 2
Ireland
Darren.Caulfield@cs.tcd.ie

Kenneth Dawson-Howe
Department of Computer Science
Trinity College
Dublin 2
Ireland
Kenneth.Dawson-Howe@cs.tcd.ie

Abstract
For surveillance systems that use data from multiple cameras the compass direction of each camera is a useful piece of information. It provides constraints on the topology of the camera network, and can limit the search space of algorithms looking for object correspondences across different cameras. This paper presents an approach for inferring the compass direction of a camera from the shadows of moving objects in a video sequence. The Sun's position is calculated using celestial mechanics, which requires the user to provide the date, time and geographic location at which the video was shot. The point on the horizon towards which all shadows on the ground appear to converge is located robustly. By combining these two pieces of information it is possible to determine the camera's compass direction. The technique has been successfully tested on a number of video sequences of people walking.

Keywords: Camera orientation, shadows, surveillance.

1 Introduction

An emerging area in the field of Computer Vision is the automatic correlation of information obtained from multiple cameras with non-overlapping fields of view [2][7]. This task can be simplified if something is known about how the various cameras are positioned relative to one another. This paper describes an approach for finding the compass direction of a camera using the shadows cast by moving objects in a video sequence of an outdoor scene. Knowing the direction in which different cameras are facing, e.g. North, South, East or West, allows us to relate the information obtained from these cameras. For example, if two cameras are positioned as shown in Figure 1(a) a person appearing in camera 1 will move into shot of camera 2 from the left, whereas in Figure 1(b) they would be expected to enter camera 2's shot from the right. Having such information available could be used, for example, to limit the search space when looking for corresponding objects, e.g. people, across multiple cameras.

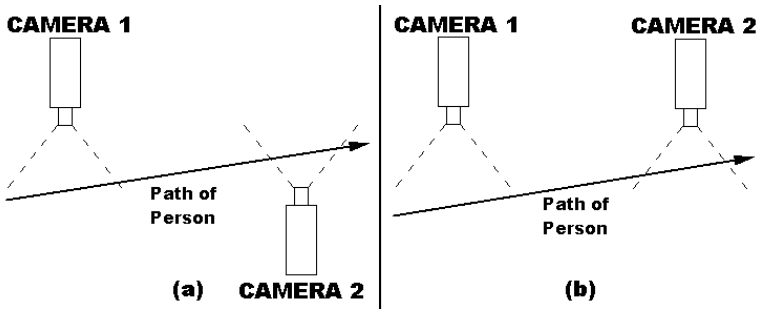


Figure 1: Difficulty of tracking when camera directions are unknown

1.1 Related Work

A substantial amount of research has been undertaken in the area of identifying shadows in images and video sequences [6][5][3]. A comparative evaluation of the main techniques can be found in [10]. The task of finding the camera's orientation relative to the scene has also been explored [8][1]. These techniques are, however, restricted to man-made scenes in which a large number of straight edges, e.g. buildings, are present. Our goal is to infer the camera's orientation to the scene from the shadows we observe. Combining this information with techniques from Astronomy for calculating the Sun's position will yield the camera's compass direction.

2 Astronomy

In order to infer the camera's compass direction it is necessary to know the position of the Sun in the sky for a given date, time and geographic location (specified by latitude and longitude). For our purposes we need the Sun's position in *horizontal co-ordinates*. These are specified by its angle above the horizon (altitude) and its angular displacement from South, travelling "around" the horizon (azimuth). (The Sun's azimuth is 90° when it is in the West.)

The position of the Sun must first be calculated in *equatorial co-ordinates*, which reference a point on the celestial sphere, an imaginary sphere that has the Earth at its centre. The first step is to calculate the Julian date (JD) for the date and time of interest. This is simply a continuous count of days and fractions of days since noon Universal Time on 1 January 4713 B.C. (on the Julian calendar). The Sun's equatorial co-ordinates, its *right ascension* (RA) and *declination* (d), are determined as follows [13]:

$$\tan RA = \cos E \sin L / \cos L \quad (1)$$

$$\sin d = \sin E \sin L \quad (2)$$

where

$$E = 23.439 - 0.00000036\ D$$
$$D = JD - 2451545.0$$
$$L = q + 1.915 \sin g + 0.020 \sin 2g$$
$$g = 357.529 + 0.98560028\ D$$
$$q = 280.459 + 0.98564736\ D$$

The observer's latitude (Lat) and longitude are then used to find the Sun's altitude (Alt) and azimuth (Az) [9]. Note that longitudes East of Greenwich are taken as positive.

$$LST = 280.46061837 + 360.98564736629\ D + \text{longitude}$$
$$HA = LST - RA$$

$$\sin Alt = \sin Lat \sin d + \cos Lat \cos d \cos HA \quad (3)$$

$$\tan (Az + 180) = (-\sin HA) / (\cos Lat \tan d - \sin Lat \cos HA) \quad (4)$$

The algorithms presented above are accurate to approximately $1/30^{th}$ of the diameter of the Sun's disk, which is more than acceptable for our purposes.

3 Shadow Identification

Shadows of moving objects are identified using the technique described in [12]. Firstly, both moving object *and* shadow pixels are identified using background subtraction in RGB colour space. Next, some of these pixels are classified as shadow. The criteria that such a pixel must meet are:

- its luminance must drop (by a limited amount) AND
- its saturation may rise only very slightly

Two parameters are required to use this algorithm. The first, required by the background subtraction technique, specifies the maximum amount by which the pixel under consideration can differ (in each channel) from the background before it is regarded as an object or shadow pixel. The second gives the maximum percentage amount by which the luminance may drop for a shadow

pixel. In order to remove isolated noise points produced by the shadow detection algorithm it was necessary to blur both the background image and the video sequence. A simple averaging operation with a neighbourhood size of 3 was found to remove the vast majority of noise points (Figure 2).

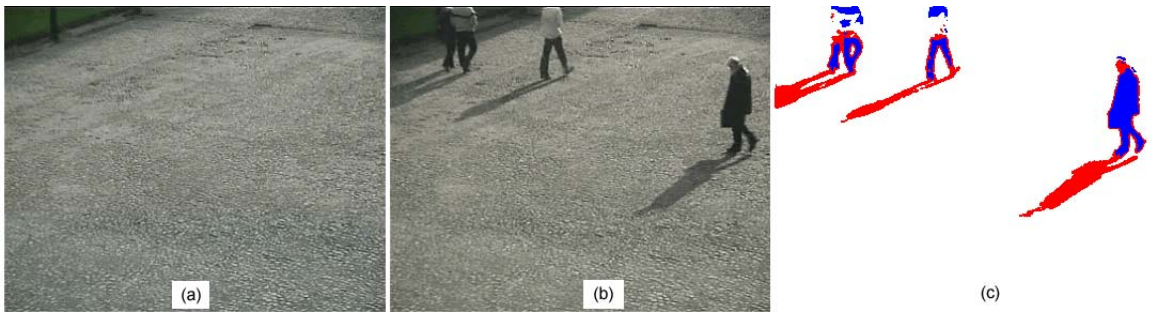


Figure 2: Background (a), current frame (b) and objects and shadows found (after blurring) (c)

A further problem was the occurrence of "false shadow" pixels surrounding the moving objects. Because such areas were very thin they were easily removed by applying an "opening" operation to the image of shadow pixels (Figure 3(b)).

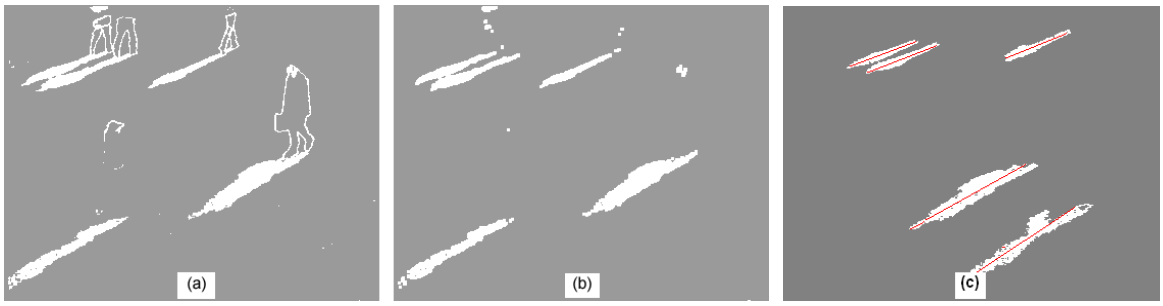


Figure 3: (a) Original binary shadow image and (b) results of opening to remove "false shadow"

(c) Shadow regions (from a different frame) overlaid with their associated directions

4 Shadow Direction

All shadows cast onto the ground by vertical objects are essentially parallel, by virtue of the fact that the Sun is at a very great distance from the Earth. Such shadows have the same compass direction (azimuth) as the Sun. They appear in a 2D image, if extended along their direction, to meet at a point on the horizon. This point V (Figure 4) is here referred to as the "vanishing point". The mathematics of the following section reveals that finding the vanishing point is necessary in order to calculate the camera's compass direction.

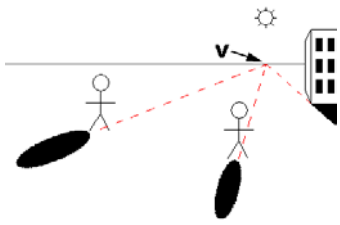


Figure 4: All ground shadows meet at a point V on the horizon

Connected Component Analysis was performed on the image of shadow pixels to yield a collection of shadow regions. For each such region a direction θ was required in order to find V (Figure 3(c)). This was achieved through the calculation of spatial and central moments for each region [11]:

$$m_{pq} = \sum_{row} \sum_{col} col^{p} row^{q} f(col, row) \tag{5}$$

where

$$f(col, row) = \begin{cases} 1 \text{ if pixel at (row, col) is part of current region} \\ 0 \text{ otherwise} \end{cases}$$

$$x_C = \frac{m_{10}}{m_{00}} \qquad y_C = \frac{m_{01}}{m_{00}} \tag{6}$$

$$\mu_{pq} = \sum_{row} \sum_{col} (col - x_c)^{p} (row - y_c)^{q} f(col, row) \tag{7}$$

$$\theta = \frac{1}{2} \tan^{-1}\left(\frac{2\mu_{11}}{\mu_{20} - \mu_{02}} \right) \tag{8}$$

4.1 Excluding Unreliable Shadow Regions

Because of failings of the shadow detection algorithm many small shadow regions were produced for which very inaccurate directions were calculated (Figure 5). An excess of such bad data would make it impossible to locate the vanishing point robustly. In order to overcome this problem a shadow region tracker was implemented. Only regions that appeared in several successive frames with similar position direction and length were considered reliable. (Equation 6 gives the centre of gravity of each region as (x_c, y_c).) Furthermore, all regions below a certain size were discarded, as were regions that did not move over time (these were often caused by specular highlights in the scene). This technique resulted in a large number of shadow regions being discarded, thus improving the quality of the data (Figure 6).

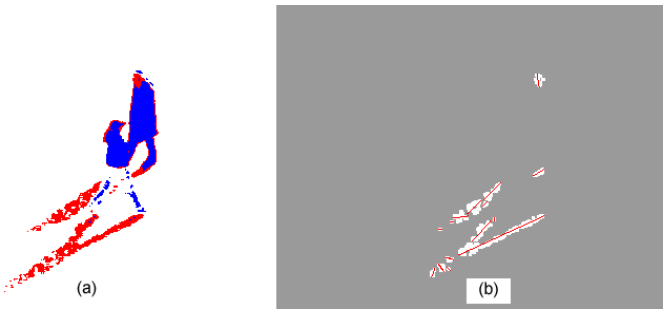


Figure 5: Poor shadow detection (a) producing unreliable line segments (b)

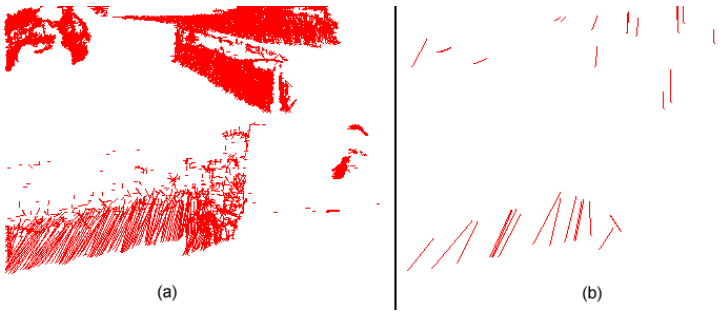


Figure 6: All line segments from a video (a) and reliable line segments only (b)

5 Camera Direction Inference

Before the camera's compass direction can be found we must find the relative orientation S of the camera and the shadows (Figure 7(a)). For this purpose we use the pinhole camera model [11].

If $\mathbf{X_W}$ represents a point in world space, then its projection **u** in the image is given by the formula:

$$\mathbf{u} = [\ KR \mid -KRt\ ]\ \mathbf{X_W} \tag{9}$$

where both **u** and $\mathbf{X_W}$ are in homogeneous co-ordinates (which allow directions as well as points to be expressed). Both K and R are 3 × 3 matrices, **u** and **t** are 3-vectors and $\mathbf{X_W}$ is a 4-vector.

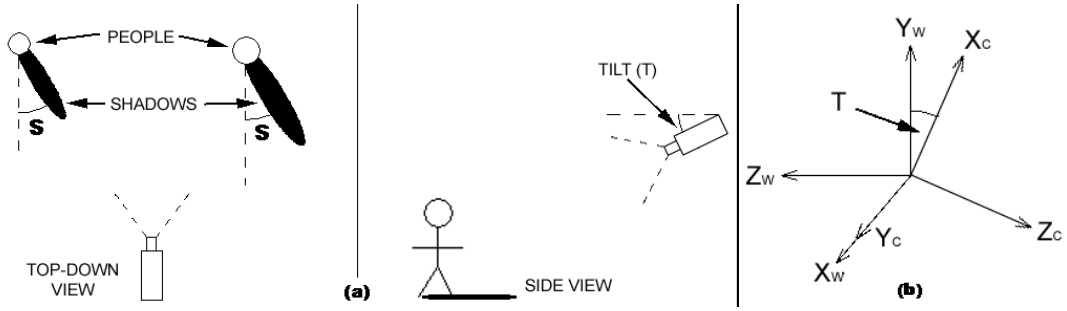


Figure 7: (a) Arrangement of camera illustrating relative orientation to shadows (S)

(b) Alignment of world and camera co-ordinate spaces for our purposes

5.1 Intrinsic Parameters

We assume the following about the camera's intrinsic parameters:

- the principal point corresponds to the centre of the image
- the camera has zero skew
- the camera has a 1:1 aspect ratio

The intrinsic parameters are given by the matrix K. The assumptions given above result in K having the following form (note that all image points must be translated to the image centre before being used in calculations):

$$K = \begin{bmatrix} \alpha & 0 & 0 \\ 0 & \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{10}$$

5.2 Extrinsic Parameters

The matrix R represents the camera's orientation in world space. We assume that the camera is level, i.e. has zero roll. It is hoped that this assumption can be relaxed in future versions of the work. In order to simplify the mathematics we align the origins of the world space and the camera space. This means that the translation vector **t** (Equation 9) is the zero vector. We also align the camera's Y-axis with the world X-axis (Figure 7(b)). The angle T is the camera's downward tilt. The structure of the matrix R is [4]:

$$R = \begin{bmatrix} 0 & \cos T & -\sin T \\ 1 & 0 & 0 \\ 0 & -\sin T & -\cos T \end{bmatrix} \tag{11}$$

5.3 Back-projection

In order to find S, the relative orientation of the camera and the shadows, we must have a correspondence between a point in the image and a point in 3-space. The vanishing point V in the image (see Figure 4) corresponds to the common direction of all the shadows. This direction can be represented in homogeneous 3-space as:

$$\mathbf{X_W} = [\ \tan S, 0, 1, 0\ ]^T \tag{12}$$

If the 2D co-ordinates of the vanishing point are measured as **u** = (u_{vp}, v_{vp}) Equation 9 expands to:

$$\beta\begin{bmatrix} u_{vp} \\ v_{vp} \\ 1 \end{bmatrix} = \left[\begin{bmatrix} -\alpha & 0 & 0 \\ 0 & \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & \cos T & -\sin T \\ 1 & 0 & 0 \\ 0 & -\sin T & -\cos T \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \right] \begin{bmatrix} \tan S \\ 0 \\ 1 \\ 0 \end{bmatrix} \tag{13}$$

It was necessary to change the sign of one element in the matrix K to account for the inversion of image space. The variable β is needed because of the use of homogeneous co-ordinates.

We wish to solve Equation 13 for S. Multiplying out the matrices yields the equations

$$\beta u_{vp} = \alpha \sin T \tag{14}$$

$$\beta v_{vp} = \alpha \tan S \tag{15}$$

$$\beta = -\cos T \tag{16}$$

Equation 16 can be used to eliminate β from Equations 14 and 15:

$$-\cos T\, u_{vp} = \alpha \sin T \tag{17}$$

$$-\cos T\, v_{vp} = \alpha \tan S \tag{18}$$

Doing so reveals that, in order to find S, either α or T is required. At the present time no other information has been extracted from the video sequence. We intend in future work to utilise the information about the Sun's altitude to provide another constraint. For the moment, however, one of the unknowns must be assumed. The scaling factor of the camera, α, is an extremely unintuitive quantity, whereas the camera's downward tilt, T, is much more meaningful. Therefore, the value of T must be provided by the user, allowing α to be eliminated from Equations 17 and 18.

$$\frac{-\cos T\, u_{vp}}{\sin T} = \frac{-\cos T\, v_{vp}}{\tan S} \quad \Rightarrow \tan S = \frac{\sin T\, v_{vp}}{u_{vp}} \tag{19}$$

Once S has been determined the camera's compass direction is found as follows:

compass direction of camera = azimuth of Sun + S (20)

5.4 Vanishing Point Estimation

The previous section shows that, in order to calculate S, the relative orientation of the camera and the shadows, the vanishing point V must be found. Given a collection of shadow regions and their associated directions (as discussed in Section 4), we must find V. Theoretically, the vanishing point is found as the intersection of any two line segments characterising each shadow region's position and direction. However, because of the existence of noisy data, such a simple technique will not work reliably. In its place we have developed a robust iterative algorithm that finds the vanishing point in spite of a high proportion of noise. The algorithm is discussed below.

Because we are assuming that the camera has zero roll we can constrain the horizon line to be horizontal in the image.

(Begin with a horizontal line far above the image.)

1. For a given horizontal line record the distribution of points where the shadow line segments intersect it.
2. Calculate the least square error for this distribution of points.
3. Repeat steps 1 and 2, moving the horizontal line down the image in small steps at each iteration, until it is far below the image.

4. Pick the horizontal line with the smallest least square error. It is an approximation to the horizon line.
5. Record the point on this horizon line that minimises the least square error. It is an approximation to the vanishing point.
6. Find the shadow line segment that intersects the horizon line furthest from the vanishing point. Remove this line segment from the data.
7. Repeat steps 1 to 6 until only two line segments remain. Their intersection represents the best approximation to the vanishing point.

6 Results

Experiments were performed using two video sequences, each approximately one minute long, of people walking across an open area on a sunny day. The videos were shot at 12.15p.m. and 12.20p.m. on 6 Nov. 2002 in Central Dublin. The downward tilt of the camera in both cases was taken as 30 degrees. In each instance the algorithm for finding the vanishing point converged to a result consistent with that found by manual calculation using "good" data (Figure 8), in spite of a high proportion of incorrectly detected shadow regions. The direction of the camera in video 1 was found as South 43 degrees East (Figure 9), and for video 2 as South 10 degrees East (Figure 10).

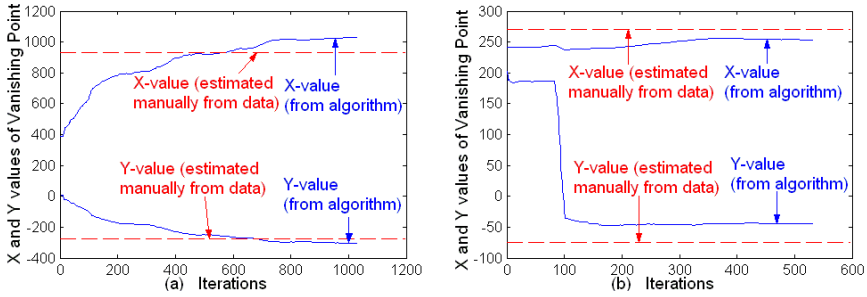


Figure 8: Convergence of algorithm towards vanishing point for (a) Video 1 and (b) Video 2

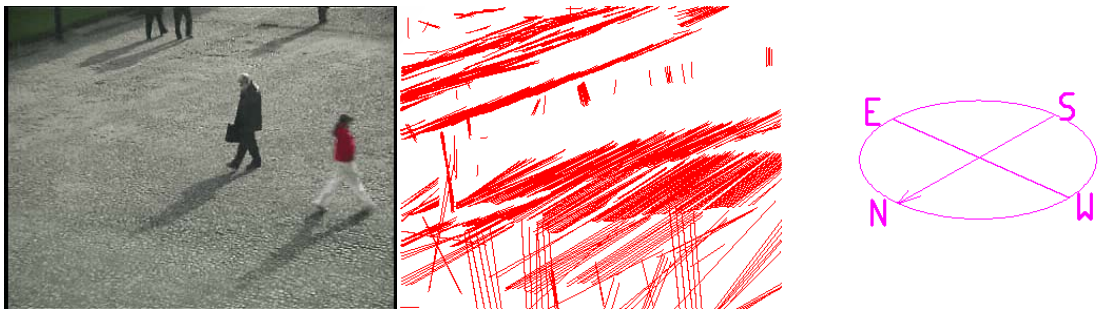


Figure 9: A frame from video 1, the line segments used to find the vanishing point and a virtual compass depicting the camera's orientation

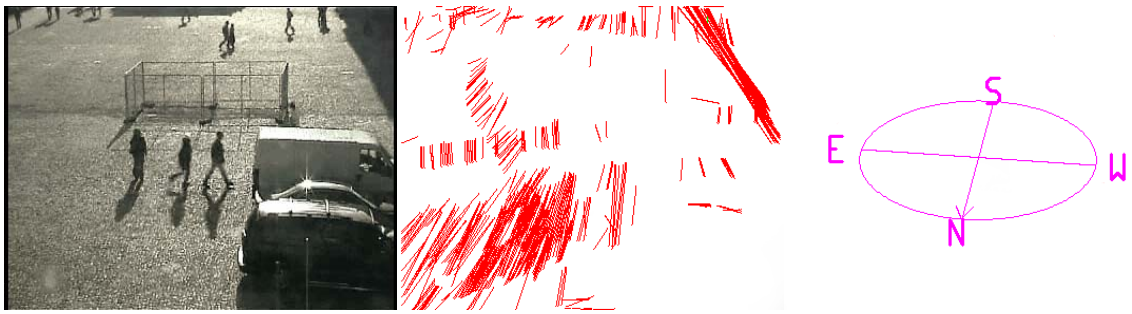


Figure 10: A frame from video 2, and its associated line segments and virtual compass

7 Future Work

We hope to extend this work to eliminate the need for the user to provide the downward tilt of the camera as a parameter. This could be accomplished by utilising the information available about the Sun's altitude and by finding correspondences between people's heads and the heads of their shadows in the video sequences. It may also be possible to extract direction information from time-lapse footage of the shadows cast by fixed objects such as buildings.

8 Conclusions

We have successfully determined the compass direction of a camera based on the shadows cast by moving objects in a video sequence. The automatic extraction of this information has applications in a multi-camera network, where a system may seek to relate the data obtained from different cameras. Knowing each camera's compass direction would constrain the search for corresponding objects across different views.

Acknowledgements

This work was in part supported by a grant from the Irish Research Council for Science, Engineering and Technology: funded by the National Development Plan.

References

[1] Coughlan, J.M., Yuille, A.L. Manhattan World: Compass Direction from a Single Image by Bayesian Inference. *Proc. International Conference on Computer Vision*, 1999.

[2] Ellis, T.J., Makris, D., Black, J.K. Learning a Multi-Camera Topology. *Proc. Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, 2003.

[3] Finlayson, G.D., Hordley, S.D., Drew, M.S. Removing Shadows from Images. *Proc. European Conference on Computer Vision*, 2002.

[4] Foley, J., van Dam, A., Fiener, S., Hughes, J. *Computer Graphics: Principles and Practice.* Addison Wesley, 1990

[5] Fung, G.S.K., Yung, N.H.C., Pang, G.K.H., Lai, A.H.S. Effective moving cast shadow detection for monocular color image sequences. *Proc. International Conference on Image Analysis and Processing*, 2001.

[6] Horprasert, T., Harwood, D., Davis, L.S. A Statistical Approach for Real-time Robust Background Subtraction and Shadow Detection. *Proc. Frame-rate Applications, Methods and Experiences with Regularly Available Technology and Equipment*, 1999.

[7] Javed, O., Rasheed, Z., Shafique, K., Shah, M. Tracking Across Multiple Cameras With Disjoint Views. *Proc. International Conference on Computer Vision*, 2003.

[8] Kosecka, J., Zhang, W. Video Compass. *Proc. European Conference on Computer Vision*, 2002.

[9] Meeus, J. *Astronomical Algorithms.* Willmann-Bell, 1991.

[10] Prati, A., Cucchiara, R., Mikic, I., Trivedi, M.M. Analysis and Detection of Shadows in Video Streams: A Comparative Evaluation. *Proc. IEEE Computer Vision and Pattern Recognition*, 2001.

[11] Sonka, M., Hlavac, V., Boyle, R. *Image Processing, Analysis, and Machine Vision.* PWS Publishing, Second Edition, 1999.

[12] Tattersall, S, Dawson-Howe, K. Adaptive shadow identification through automatic parameter estimation in video sequences, *Proc. Irish Machine Vision and Image Processing Conference (IMVIP 2003),* Coleraine, Ireland, 3rd-5th September 2003, pp.57-64.

[13] U.S Nautical Almanac Office, Her Majesty's Nautical Almanac Office. *The Astronomical Almanac.* U. S. Government Printing Office, Washington, D. C. 20402, 2004.

COMPARISON OF TWO ALGORITHMS FOR ROBUST M-ESTIMATION OF GLOBAL MOTION PARAMETERS

Rozenn Dahyot
Department of Statistics
Trinity College Dublin, Ireland
email: dahyot@mee.tcd.ie

Anil Kokaram
Electronic & Electrical Engineering Dpt
Trinity College Dublin, Ireland
email: akokaram@tcd.ie

Abstract

The estimation of Global or Camera motion from image sequences is important both for video retrieval and compression (MPEG4). This is frequently performed using robust M-estimators with the widely used Iterative Reweighted Least Squares algorithm. This article presents an investigation of the use of an alternative robust estimation algorithm and illustrates its improved computationnal efficiency. The paper also introduces two new confidence measures which can be used to validate camera motion measurements in the context of information retrieval.
Keywords*: Camera motion, M-estimators, Video analysis.*

1 Introduction

Content based information retrieval has been a highly active research area during the last decade [14]. The motivation has been that access via keywords allows only a primitive interaction with non-text media in general. To allow access on the basis of content (e.g. responses to questions like "find all aces in a game of tennis") is the key to efficient exploitation of these kinds of information. Typically the process begins by identifying and extracting feature primitives (colour, shape, motion,...) relevant to the content and these features might be related in some way to the human perception of the data. The features are then manipulated jointly in response to user queries or in order to identify events.

Motion is clearly an important feature in retrieval from video media, and Global Motion in particular captures the movement of the camera operator. This motion is well correlated to important events in video for example sport broadcasts [11] and can be used as a preliminary task before local motion analysis [9]. In this paper, global motion is considered to be that single motion representation which accounts for the largest moving area in the image.

A 6 parameter affine model is introduced in section 2 to represent this displacement. Those parameters are then estimated by minimising an energy function. It is local motion that complicates the estimation of global motion. In effect, that area of the image which undergoes local motion (or discontinuity) acts as an outlier in the global motion model. Standard estimation methods as presented in section 3, are sensitive to the presence of such outliers. By using instead, robust estimation processes [4] (M-estimators), it is possible to handle the presence of contaminated data in the observations i.e. the effect of global motion in this case. M-estimation has been applied to many problems in computer vision such as regularisation [6], motion optical flow estimation [1], object tracking [2], or object recognition and detection [8].

This paper considers two algorithms used for performing robust Global motion estimation with M-estimators. The first is the so-called *Iterative Reweighted Least Squares* (IRLS), and the second is the little known *Iterative Modified Residuals* (IMR) [10]. This article quantitatively analyses how well they perform with respect to the accuracy of the motion estimate itself and it compares their computation times. The paper shows that in fact, the two algorithms lead to the same performances for accuracy, the IMR is more computationally efficient.

Of special importance in any estimation problem on real data is to measure the confidence of the estimates. The paper also introduces two new confidence measures which can be use to validate camera motion measurements in the context of information retrieval.

2 Image sequence modelling

Motion estimation techniques presented in this article rely on the following image sequence model:

$$I_n(\mathbf{x}) = I_{n-1}(F(\mathbf{x},\Theta)) + \epsilon(\mathbf{x}) \tag{1}$$

where $I_n(\mathbf{x})$ is the grey level of the pixel at the location given by position vector $\mathbf{x}$ in the frame n. The vector function $F(\mathbf{x},\Theta)$ is the transformation of image coordinates induced by the motion between time $n-1$ and n. Θ is the vector formed by the motion parameters. In other words, it means that the current frame can be created by rearranging the position of the intensities from the previous frame.

Camera Motion Model. To represent motion such as zooming, rotation and translation between the current frame n and the previous frame $n-1$, a 6-parameter affine transformation $F(\mathbf{x},\Theta) = \mathrm{A}\mathbf{x} + \mathbf{d}$ is used where A is a 2×2 matrix for affine transformation and $\mathbf{d}$ is the displacement vector, as below:

$$\begin{aligned} F(\mathbf{x},\Theta) &= \mathrm{A}\mathbf{x} + \mathbf{d} \\ &= \begin{pmatrix} a_1 & a_2 \\ a_3 & a_4 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} d_x \\ d_y \end{pmatrix} \\ &= \begin{pmatrix} a_1\, x + a_2\, y + d_x \\ a_3\, x + a_4\, y + d_y \end{pmatrix} = \mathrm{B}(\mathbf{x})\ \Theta \end{aligned} \tag{2}$$

where $\mathrm{B}(\mathbf{x}) = \begin{pmatrix} x & y & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x & y & 1 \end{pmatrix}$ and $\Theta = [a_1, a_2, d_x, a_3, a_4, d_y]^T$.

Toward a linear residual. The parameter Θ can be estimated by minimising some function of the residual $\epsilon(\mathbf{x}) = I_n(\mathbf{x}) - I_{n-1}(F(\mathbf{x},\Theta))$. This residual $\epsilon(\mathbf{x})$ is however not linear in Θ. A Taylor series expansion around the motion parameters Θ is used to linearise the parameter estimation problem as follows:

$$I_n(\mathbf{x}) - I_{n-1}(\mathrm{B}(\mathbf{x})\ \Theta) = \nabla I_{n-1}(\mathrm{B}(\mathbf{x})\ \Theta) \cdot \mathrm{B}(\mathbf{x}) \cdot \delta\Theta + \varepsilon(\mathbf{x}) \tag{3}$$

$\epsilon(\mathbf{x})$ and the higher order terms of the expansion are lumped together in the new residual $\varepsilon(\mathbf{x})$ linear with respect to the update $\delta\Theta$. The ∇ operator is the usual multidimensional gradient operator. The estimation proceeds by recursive estimation of $\delta\Theta$ and updating of $\Theta \leftarrow \Theta + \delta\Theta$. This simple idea unifies all previous approaches to Global Motion estimations.

The equation (3) is expressed at each location $\mathbf{x}$. Considering all the pixels, it can be rewritten using vectors and matrices such as:

$$\mathbf{z} = \mathrm{G} \cdot \delta\Theta + \boldsymbol{\varepsilon} \tag{4}$$

with the vectors (limiting the notation to the first two locations $\mathbf{x_1} = (x_1, y_1)$ and $\mathbf{x_2} = (x_2, y_2)$):

$$\mathbf{z} = \begin{bmatrix} I_n(\mathbf{x_1}) - I_{n-1}(\mathrm{B}(\mathbf{x_1})\ \Theta) \\ I_n(\mathbf{x_2}) - I_{n-1}(\mathrm{B}(\mathbf{x_2})\ \Theta) \\ \vdots \end{bmatrix}, \quad \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon(\mathbf{x_1}) \\ \varepsilon(\mathbf{x_2}) \\ \vdots \end{bmatrix}$$

and the following matrix:

$$\mathrm{G} = \begin{bmatrix} x_1 \cdot I^x_{n-1}(\mathrm{B}(\mathbf{x_1})) & y_1 \cdot I^x_{n-1}(\mathrm{B}(\mathbf{x_1})) & I^x_{n-1}(\mathrm{B}(\mathbf{x_1})) & x_1 \cdot I^y_{n-1}(\mathrm{B}(\mathbf{x_1})) & y_1 \cdot I^y_{n-1}(\mathrm{B}(\mathbf{x_1})) & I^y_{n-1}(\mathrm{B}(\mathbf{x_1})) \\ x_2 \cdot I^x_{n-1}(\mathrm{B}(\mathbf{x_2})) & y_2 \cdot I^x_{n-1}(\mathrm{B}(\mathbf{x_2})) & I^x_{n-1}(\mathrm{B}(\mathbf{x_2})) & x_2 \cdot I^y_{n-1}(\mathrm{B}(\mathbf{x_2})) & y_2 \cdot I^y_{n-1}(\mathrm{B}(\mathbf{x_2})) & I^y_{n-1}(\mathrm{B}(\mathbf{x_2})) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

3 Maximum likelihood estimation

A maximum likelihood approach to estimation of Θ would choose an estimate $\hat{\Theta}$ which maximises the likelihood of $\epsilon(\mathbf{x})$ as all the sites in the image simultaneously $\mathcal{P}(\varepsilon)$. This is equivalent to minimising the log liklelihood $-\log \mathcal{P}(\varepsilon)$. In practice, by exploiting the linearisation of the log-likelihood (as in the previous section) around a current estimate of Θ called Θ_i; it is possible to generate an estimate by successively updating $\hat{\Theta}$ through $\hat{\Theta}^{i+1} = \Theta^i + \hat{\delta\Theta}$. Where $\delta\Theta^i$ is the update to be estimated.

Assuming the distribution of the residual $\boldsymbol{\varepsilon}$ is spherical (i.e. multidimensional gaussian with a covariance matrix proportional to the identity matrix), the algorithm can be written as :

$$\begin{array}{l} \text{Do} \\ \quad \widehat{\delta\Theta} = \arg\min_{\delta\Theta}\{\mathcal{J}(\boldsymbol{\varepsilon}^{(i)}) = \sum_{\mathbf{x}}[\varepsilon^{(i)}(\mathbf{x})]^2\} \\ \quad \widehat{\Theta}^{(i+1)} = \widehat{\Theta}^{(i)} + \widehat{\delta\Theta} \\ \text{Until convergence at final step } \boldsymbol{i}\ (\widehat{\Theta} = \widehat{\Theta}^{(\boldsymbol{i})}) \end{array} \tag{5}$$

At each step i, $\widehat{\delta\Theta}$ is estimated by Least Squares:

$$\widehat{\delta\Theta} = [\mathrm{G}^{(i)\,T}\mathrm{G}^{(i)}]^{-1}\mathrm{G}^{(i)\,T}\mathbf{z}^{(i)} \tag{6}$$

4 Robust M-estimation

Least Square estimation is sensitive to gross errors (or outliers) due to, for instance, local motion in the video different from the global motion or occlusion effects. M-estimators [10] are now widely used to perform robust estimation of global motion parameters [4, 15, 13]. The underlying assumption is that the probability density function of the residuals is no longer gaussian, and can be written as:

$$\mathcal{P}(\boldsymbol{\varepsilon}) \propto \exp\left[-\frac{1}{2}\sum_{\mathbf{x}} \rho\left(\frac{\varepsilon(\mathbf{x})}{\sigma_\rho}\right)\right] \tag{7}$$

Several functions ρ, convex or non-convex, have been proposed in the literature [4, 8, 13]. In our experiments (section 6), we have chosen the convex function $\rho(t) = 2\sqrt{1+t^2} - 2$, in order to avoid problems with local minima occurring with non-convex ones which could disturb the comparison of the algorithms. σ_ρ is the *scale parameter* that controls the limit where the influence of the outliers begins to decrease [10]. This parameter is fixed offline [4] but to simplify, it is set to 1 in the following equations.

The corresponding energy to minimize is non-quadratic and requires specific algorithms. Two have been proposed in the literature. The first is widely used and is called *The location step with modified weights* [10] in the robust statistic framework, or more commonly the *Iterative Reweighted Least Squares* [12], or as ARTUR in the Half Quadratic (HQ) formulation [5]. The IRLS algorithm is reviewed in paragraph 4.2.

The second algorithm, little known in computer vision literature, has first been called *The location step with modified residuals* in robust statistics [10] and as LEGEND in the HQ framework [5]. This algorithm is referred as Iterative Modified Residuals (IMR) and is explained in paragraph 4.3 for global motion parameter estimation. The section 4.1 briefly explains the origins of the two algorithms and they are compared in section 4.3.

4.1 Half-Quadratic Theory

Several explanations can account for both algorithms [10, 5, 8], but for simplicity we choose here the HQ framework. Maximising $\mathcal{P}(\boldsymbol{\varepsilon})$ in Θ is equivalent to minimize $\mathcal{J}(\boldsymbol{\varepsilon}) = \sum_{\mathbf{x}} \rho(\varepsilon(\mathbf{x}))$ iteratively in $\delta\Theta$. HQ theory defines an augmented energy $\mathcal{J}^*$ with the same global minimum:

$$\mathcal{J}(\boldsymbol{\varepsilon}) = \min_{\mathbf{b}} \left\{ \mathcal{J}^*(\boldsymbol{\varepsilon}, \mathbf{b}) = \sum_{\mathbf{x}} \rho^*(\varepsilon(\mathbf{x}), b(\mathbf{x})) \right\} \tag{8}$$

$\mathcal{J}^*$ is minimized iteratively in $\delta\Theta$ and $\mathbf{b} = \{b(\mathbf{x})\}_{\mathbf{x}}$:

Do
  Do
    $\delta\Theta^{(j)} = \arg\min_{\delta\Theta} \left\{ \mathcal{J}^*(\boldsymbol{\varepsilon}^{(j)}, \mathbf{b}^{(j)}) \right\}$
    $\mathbf{b}^{(j+1)} = \arg\min_{\mathbf{b}} \left\{ \mathcal{J}^*(\boldsymbol{\varepsilon}^{(j+1)}, \mathbf{b}^{(j)}) \right\}$
  Until convergence at final step $\boldsymbol{j}$ $(\widehat{\delta\Theta} = \delta\Theta^{(\boldsymbol{j})})$
  $\widehat{\Theta}^{(i+1)} = \widehat{\Theta}^{(i)} + \widehat{\delta\Theta}$
Until convergence at final step $\boldsymbol{i}$ $(\widehat{\Theta} = \widehat{\Theta}^{(\boldsymbol{i})})$

The different interactions of the auxiliary variable $\mathbf{b}$ with the residual $\boldsymbol{\varepsilon}$ defines the different robust algorithms of the M-estimation. $\mathbf{b}$ corresponds to weights on the residuals in the IRLS algorithm and is denoted $\mathbf{w}$ in section 4.2.

4.2 Iterative Reweighted Least Squares (IRLS)

The first proposed augmented energy can be written as:

$$\mathcal{J}^*(\boldsymbol{\varepsilon}, \mathbf{w}) = \sum_{\mathbf{x}} w(\mathbf{x}) \, [\varepsilon(\mathbf{x})]^2 + \Psi(w(\mathbf{x})) \tag{9}$$

When the auxiliary variable $\mathbf{w} = \{w(\mathbf{x})\}_{\mathbf{x}}$ is fixed, the update is estimated by weighted Least Squares:

$$\delta\Theta^{(j)} = [\mathrm{G}^{(i)\,T} \mathrm{W}^{(j)} \mathrm{G}^{(i)}]^{-1} \, (\mathrm{G}^{(i)})^T \mathrm{W}^{(j)} \mathbf{z}^{(i)} \tag{10}$$

The diagonal matrix $\mathrm{W} = \mathrm{diag}(\mathbf{w})$ is then updated by $w^{(j+1)}(\mathbf{x}) = \frac{\rho'(\varepsilon(\mathbf{x}))}{2 \cdot \varepsilon(\mathbf{x})}$. The weights act to reduce the effect of large residuals in the estimation process. A parametric expression of the function Ψ is proposed in [8]:

$$\left| \begin{array}{l} w = \frac{\rho'(\varepsilon)}{2\varepsilon} \\ \Psi = \rho(\varepsilon) - \frac{\rho'(\varepsilon)}{2}\varepsilon \end{array} \right.$$

Under some hypothesis on ρ [7, 3], this can be expressed as $\Psi(w) = \phi((\phi')^{-1}(w)) - w \, (\phi')^{-1}(w)$ with $\phi(x^2) = \rho(x)$.

4.3 Iterative Modified Residuals (IMR)

The second augmented energy can be written as:

$$\mathcal{J}^*(\boldsymbol{\varepsilon}, \mathbf{b}) = \sum_{\mathbf{x}} [\varepsilon(\mathbf{x}) - b(\mathbf{x})]^2 + \xi(b(\mathbf{x})) \tag{11}$$

When $\mathbf{b} = \{b(\mathbf{x})\}_{\mathbf{x}}$ is fixed, the update is computed by :

$$\delta\Theta^{(j)} = [\mathrm{G}^{(i)\,T} \, \mathrm{G}^{(i)}]^{-1} \, \mathrm{G}^{(i)\,T} \, (\mathbf{z}^{(i)} - \mathbf{b}^{(j)}) \tag{12}$$

The vector $\mathbf{b}$ is updated by $b^{(j+1)}(\mathbf{x}) = \varepsilon(\mathbf{x}) \left(1 - \frac{\rho^{'}(\varepsilon(\mathbf{x}))}{2\,\varepsilon(\mathbf{x})}\right)$. Here the auxiliary variable acts again to reduce the effect of large residuals but here by subtraction of each outlier. ξ has been defined as [7]: $\xi(b) = \sup_u\{-(u-b)^2 + \rho(u)\}$. As this expression is not analytically exploitable, a parametric expression of ξ has also been proposed [8]:

$$\left| \begin{array}{l} b = \varepsilon \left(1 - \frac{\rho^{'}(\varepsilon)}{2\varepsilon}\right) \\ \xi = \rho(\varepsilon) - \left(\frac{\rho^{'}(\varepsilon)}{2}\right)^2 \end{array} \right. \tag{13}$$

Remarks It has been shown that the IRLS algorithm converges in less steps $\boldsymbol{j}$ to the estimate $\widehat{\delta\Theta}$ than the IMR [10, 8] (cf. figure 1). But in comparing equations (12) and (10), we see that the IMR algorithm involves less computation (product of matrixes $[\mathrm{G}^{(i)\,T}\ \mathrm{G}^{(i)}]$) at each j step than the IRLS ($[\mathrm{G}^{(i)\,T}\mathrm{W}^{(j)}\mathrm{G}^{(i)}]$).

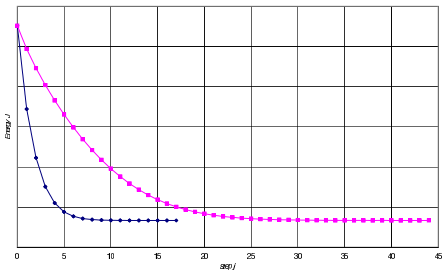


Figure 1: Energy $\mathcal{J}$ with respect to the step j. Convergence to the minimum faster for IRLS (diamond-blue) than for IMR (square-pink).

5 Confidence measures

The local motion behaviour or discontinuity contaminates the observations of the global motion. The level of contamination can be evaluated using the following measures.

Using the residuals. The probability density function of the residuals $\max_\Theta \mathcal{P}(\boldsymbol{\varepsilon})$ (or its corresponding energy defined as $\min_\Theta\{-\log \mathcal{P}(\boldsymbol{\varepsilon})\}$) is directly connected to estimator behaviour. If the estimate exactly accounts for the motion of each pixel within the image that is undergoing global motion, then the residual energy is zero (consequently, the likelihood probability is high). Conversely, high residual energy implies poor estimation and low likelihood probability.

Using the weights. The auxiliary variable used in the IRLS algorithm collects the weights defined on each residual. This weight is close to one when the residual is an inlier for the estimation of the global motion parameter, and close to zero otherwise. The image of the weights can be seen as a confidence map on the data. We propose a measure using those weights: $E[w^2] = \frac{\sum_{\mathbf{x}}[w(\mathbf{x})]^2}{\sum_{\mathbf{x}} 1}$. The measure is the Mean Square weight (MSW) across the entire image, and if most of the image can be accounted for by global motion, one would expect the IRLS MSW to be 1.0 and the IMR MSW to be 0.0. This measure is slightly different from that proposed by Bouthemy et al.[4] in that it does not require any prior thresholding.

6 Experimental results

Artificial sequences Three artificial video sequences (50 images of 360×288 pixels) were generated by applying a motion with known parameters on an original frame. The sequences show accelerating global motion in order to simulate rapid camera action.

Accuracy. As the table 1 shows, the accuracy of both algorithms on each parameter are the same. The error on the translation parameters is bigger than the one of matrix A, but is still very small ($<<$ 1 pel).

	a_1	a_2	a_3	a_4	d_x	d_y
IRLS	4.10^{-4}	3.10^{-5}	3.10^{-5}	4.10^{-4}	1.10^{-2}	1.10^{-2}
IMR	4.10^{-4}	3.10^{-5}	4.10^{-4}	3.10^{-5}	1.10^{-2}	1.10^{-2}

Table 1: Average error on the parameters.

Computation time. The estimation has been computed using a three level pyramid as in [4] both to speed up the computation and to have accurate results. Figure 2 shows the advantage of this approach in terms of computation time: the notations IRLS and IMR (respectively PYR-IRLS and PYR-IMR) mean that the estimation has been performed without (resp. with) the pyramid of resolution. The curves show the computation time (in ms) for each frame of the sequence, and have been computed for the pan-only sequence which presents a decelerating translation for each frame n such that: $d_x(n) = 7 \exp\left[-\frac{n}{25}\right]$. The initial guess of the algorithms for each frame n is the identity transformation (i.e. in particular $d_x^{(i=0)} = 0$). At the beginning of the sequence, the initial guess is far from the solution, and therefore both algorithms (without using the pyramid decomposition) require more time to converge than at the end. This is not the case using the pyramid decomposition where a coarse to fine refinement is used. The

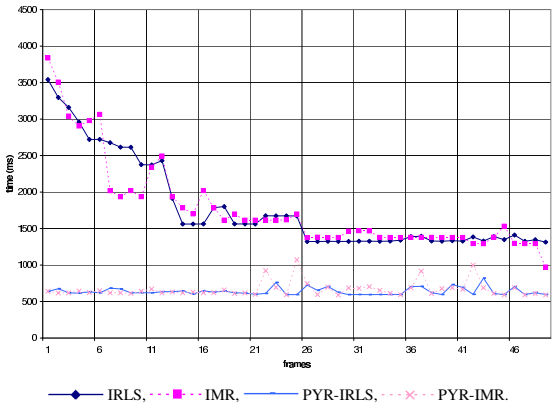


Figure 2: Computation time (pan-only).

table 2 presents the average time of computation on all our sequences using the pyramid decomposition. No obvious difference appears between the performance of IMR and IRLS.

	zoompan	pan only	zoom only
IRLS	641	639	605
IMR	633	666	607

Table 2: Average time (ms) on artificial sequences.

Real video sequences A video sequence of cricket has been processed using both algorithms (1500 images of 720×576 pixels).

Computation time. On the overall sequence, the IMR is reaching the estimate 10% faster than the IRLS (in comparing their average times $5612ms$ for IMR and $6476ms$ for IRLS over the sequence). As noticed in section 4.3, when the pixels in the image are numerous (that increases the size of the matrixes involved in the computation), the computation costs at each j step involving the products of matrixes in the IRLS algorithm can become time consuming.

Confidence measures. The figure 3 shows the confidence measures computed for the global motion parameter over the cricket sequence: there is an obvious correlation between the shot changes and weak values of the confidence measures (high values of the energy using the residuals, and low values of the confidence measure using weights). Abrupt transitions like cuts are well detected by both confidence measures, but not gradual transitions. As noticed in [4], confidence measures on the estimated parameters can also be used to detect shot changes in the video sequences. Figure 6 presents the weights estimated

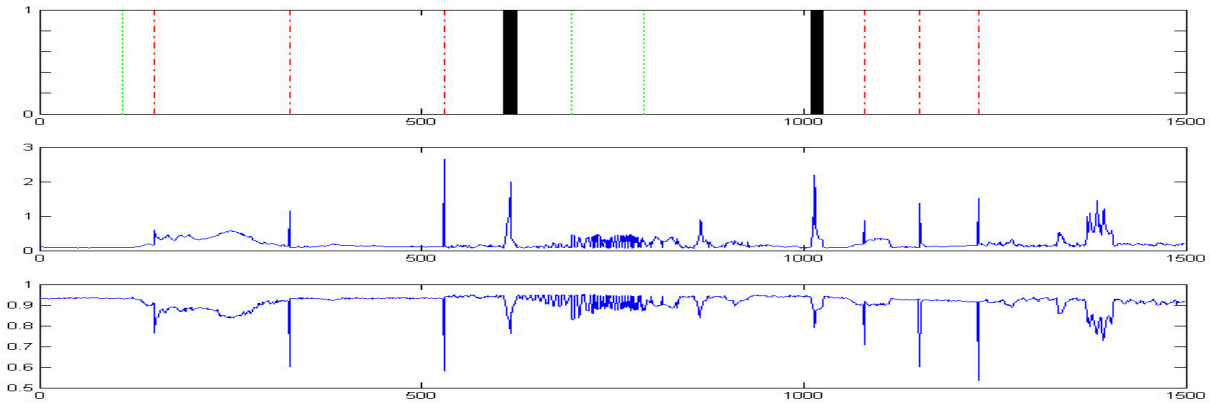


Figure 3: From top to bottom : ground truth of shot transitions (red dashdot lines for cuts, green dot lines for dissolves, black solid lines for wipes), confidence measures using residuals (middle) and weights (bottom).

for some images of the sequence. The weights are presented scaled by 255. High brightness represents pixels with high weights and dark pixels represent those with low weights. Low weights indicate pixels which are *not* part of the area undergoing global motion. This is in effect a representation of objects that are not following the camera motion such as the wipe in image $n = 616$ or people in frame 1400. Strong camera activity is detected in image $n = 251$ (travelling $d_x = 16$) through the presence of outliers on the right and left borders of the weight map. These are caused by off-scene locations that cannot be matched in the successive images because of the large inter-frame motion.

7 Conclusion

We have presented two algorithms performing the robust M-estimation. Depending on the size of the images, we have shown that the IMR algorithm can be faster, for the same accuracy, than the IRLS algorithm usually used to solve M-estimation for the global motion estimation problem. Global motion parameters are used for instance to index sport events [11] since the movement of the camera is highly correlated to the game. The two measures characterising the amount of contaminated data can help to detect shot changes [4]. Finally, weight maps provide a interesting start for local motion analysis and object segmentation in videos.

$n = 616$ $n = 551$ $n = 251$ $n = 1400$

Figure 4: Images from the cricket sequence with their weighting map $\{w(\mathbf{x})\}_{\mathbf{x}}$. Black pixels correspond to weights close to 0 (outliers), and white ones are weights close to 1 (inliers).

References

[1] M. J. Black and P. Anandan. A framework for the robust estimation of optical flow. In *Proceedings of the 4th International Conference on Computer Vision*, pages 231–236, Berlin, Allemagne, May 1993.

[2] M. J. Black and A. D. Jepson. Eigentracking: Robust matching and tracking of articulated objects using a view-based representation. *International Journal on Computer Vision*, 26(1):63–84, January 1998.

[3] M. J. Black and A. Rangarajan. On the unification of line processes, outlier rejection, and robust statistics with applications in early vision. *International Journal of Computer Vision*, 19(1):57–92, July 1996.

[4] P. Bouthémy, M. Gelgon, and F. Ganansia. A unified approach to shot change detection and camera motion characterization. *IEEE Transactions on Circuits and Systems for Video Technology*, 9:1030–1044, 1999.

[5] P. Charbonnier, L. Blanc-Feraud, G. Aubert, and M. Barlaud. Two deterministic half-quadratic regularization algorithms for computed imaging. In *Proceedings of the International Conference on Image Processing ICIP-94*, volume 2, pages 168–172, 1994.

[6] P. Charbonnier, L. Blanc-Feraud, G. Aubert, and M. Barlaud. Deterministic edge-preserving regularization in computed imaging. *IEEE Transactions on Image Processing*, 6(2):298 – 311, February 1997.

[7] Pierre Charbonnier. *Reconstruction d'image: régularisation avec prise en compte des discontinuités*. PhD thesis, Université de Nice-Sophia Antipolis, France, 1994.

[8] R. Dahyot. *Analyse d'images séquentielles de scènes routières par modèles d'apparence pour la gestion du réseau routier*. Editions LCPC, ISBN: 2-7208-2028-1, France, September 2003.

[9] R. Fablet, P. Bouthemy, and P. Perez. Nonparametric motion characterization using causal probabilistic models for video indexing and retrieval. *IEEE Transactions on Image Processing*, 11(4):393–407, April 2002.

[10] P.J. Huber. *Robust Statistics*. John Wiley and Sons, 1981.

[11] A. Kokaram and P. Delacourt. A new global estimation algorithm and its application to retrieval in sport events. In *IEEE International Workshop on Multimedia Signal Processing, MMSP'01*, Cannes, France, October 2001.

[12] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C - The Art of Scientific Computing*. Cambridge University Press, 1995.

[13] Wei Qi and Yuzhuo Zhong. New robust global motion estimation approach used in mpeg-4. *Journal of Tsinghua University Science and Technology*, 2001.

[14] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, December 2000.

[15] A. Smolic and J.-R. Ohm. Robust global motion estimation using a simplified m-estimator approach. In *IEEE International Conference on Image Processing*, Vancouver, Canada, September 2000.

VIDEO SEQUENCE INDEXING THROUGH RECOVERY OF OBJECT-BASED MOTION TRAJECTORIES

A. Naftel
Department of Computation
UMIST
Sackville Street
Manchester M60 1QD, UK
andrew.naftel@co.umist.ac.uk

S. Khalid
Department of Computation
UMIST
Sackville Street
Manchester M60 1QD, UK
s.khalid-2@postgrad.umist.ac.uk

Abstract
In this paper, we present a hybrid approach for tracking multiple objects through occlusion observed by a stationary camera. This tracking data can then be used to generate accurate object motion trajectories that provide an index key into a database of motion sequences. The system is intended for tracking people and objects in crowded environments such as supermarkets or shopping malls. The output of the system can be used for intelligent behavioural analysis or activity-based video indexing and retrieval for security management. The approach starts with a robust foreground object detection and SAKBOT-based shadow suppression stage. It is shown how both static and dynamic occlusions are handled using a first-order Kalman Filter combined with a simple colour model incorporating histogram intersection and back-projection for each tracked object. In the next step we use a RANSAC-type approach to generate smooth motion trajectories for each object modelled with *m*-degree polynomials. Preliminary results are presented to show how our method produces robust trajectory paths insensitive to outliers containing high numbers of mis-detected points. A similarity metric is then defined using polynomial coefficients. This enables a user to construct a motion trajectory query which can be used to index into a database of surveillance clips and retrieve similar results..

Keywords: object tracking, shadow detection, motion trajectory, occlusion handling.

1 Introduction

Intelligent surveillance systems are assuming an increasingly important role in crime detection and prevention as the number of installed camera networks can attest. One of the most important tasks for the next generation of commercial CCTV surveillance systems is to automate the process of tracking people, objects and their interactions in complex and crowded environments. The tracking problem (i.e. establishing inter-frame correspondence for individual objects over a video sequence) has been extensively studied in the computer vision literature [1][2]. However, the issue of how to curate the vast quantities of tracking data collected has only recently been addressed by researchers. One approach is through semantic video interpretation [3] where the system attempts to recognise user-predefined events such as certain types of possible criminal activity. An alternative is to analyse object motion paths [4][5][6] in order to learn and predict patterns of behaviour, or to allow users to create queries about the content of surveillance scenes [7][8][9], e.g. trajectory, colour, type of object, etc. and thereby retrieve useful information.

Our work most closely relates to [7][8], since the aim of the project is to develop a system for indexing and retrieval of relevant video sequences based on object motion paths. The specific application domain addressed is indoor retail store surveillance which offers a number of challenging problems when attempting to automate scene analysis. These are as follows:

- **Static/dynamic occlusion**: Indoor environments such as retail stores and shopping malls are often crowded and hence are full of static and dynamic objects that may partially or totally occlude the

target object. This results in rapid appearance and shape changes which must be dealt with carefully if identified objects are not to be mis-classified. This is an inherent problem when attempting to analyse crowded scenes.

- **Shadows:** Often strong artificial illumination from multiple light sources used in indoor scenes introduces problems in effective foreground detection due to the generation of shadows of varying intensities in different parts of the scene. When the object moves close to the light source, the intensity of the shadow increases rapidly.
- **Background changes:** Previously moving objects that suddenly become stationary in the scene for long periods or sharp changes in lighting conditions cause instability in the background model. As the first step in reliable object segmentation is normally background subtraction, the system must react and adapt to the background changes by frequently updating the model.

1.1 Related work

There have been numerous efforts to robustly track objects in crowded scenes. This work can be categorised into single or multiple cameras, static or moving cameras, colour or grey scale, and single or multiple person tracking systems. The W^4 system [10] tracks people in grey scale video obtained from a static camera. The foreground object is detected using a statistical background model, and a set of features (such as silhouette calculation, body parts localisation) for each object and group is computed. These features are then used to track moving objects through various types of occlusion. An enhancement to the system known as W4S, which integrates real-time stereo computation in order to suppress shadows (previously detected as separate blob or new foreground objects), is described in [11]. Other noteworthy tracking systems working with fixed cameras have been reported [12][13][14].

In [15], tracking is accomplished by decoupling the problem into two parts. Firstly, the object appearance is defined using a colour-based object representation and it then models 2D and 3D velocities of the object. An appearance-based description of moving objects is used for measuring similarity among detected moving objects whereas Kalman Filtering is used for 2D/3D modelling of tracked objects. The SAKBOT [16] approach enables effective tracking of objects, even in the presence of heavy shadows. It uses HSV colour space to improve the accuracy in detecting shadows by exploiting the general effects of shadows on the HSV component of the pixel on which it falls. This effect includes a lowering of brightness value for the pixel (caused by darkening) greater than expected with little effect on the S and V (colour) component.

Here, we propose a simple and effective solution for tracking multiple objects in a busy environment such as a shopping centre or retail store. The solution combines various existing techniques with some modifications.

The remainder of the paper is organised as follows. In section 2, an algorithm to detect foreground objects using an adaptive statistical background model is discussed. A technique to avoid moving shadows being classified as part of a moving object is then described. Section 3 specifies the algorithm used to track objects in various possible scenarios. These scenarios include tracking of multiple objects through both partial and complete static or dynamic occlusions. The algorithm uses a hybrid model comprising a spatial prediction component based on the output of a first order Kalman Filter and appearance model component based on technique of colour indexing proposed by Swain and Ballard [17]. In section 4, we describe the procedure for modelling the motion path generated by the object tracker points. This method is shown to be insensitive to gross outliers in the data, instabilities inherent in the tracking algorithm and is particularly suited to smoothing through occluded sequences. We then show how to use the output for indexing motion histories. Preliminary results are presented in section 5, concluding with a discussion and summary in section 6.

2 Foreground Object Detection

We adopt the adaptive background modelling technique based on [10] which has proved reliable. Before background subtraction can be applied, an initial background model should be learned based on frames with a majority of the background visible. However, the algorithm can create an initial background model even if there are small localised visible objects moving in the scene. A set of masks can be used to neglect the moving objects and we can select only the valid regions to be used to update the background model. It also caters for any object that moves into the scene (then identified as a foreground object) and remains stationary for a long time period. Similarly, any object which is initially assumed to be the part of the background but then starts to move, can cause false background changes for a short period but settles down in a reasonable number of frames.

This is combined with shadow detection based on SAKBOT model [16]. Shadows are detected by assuming that they reduce the intensity of the underlying pixel without having a significant effect on its colour. As background subtraction only takes into account the brightness component of pixel, we need to model hue and saturation pixel components separately for shadow removal. The result of applying background subtraction with shadow suppression is shown in Fig 1.

(a) (b) (c)

Fig. 1. (a) Current frame. (b) Foreground object detected after background subtraction. (c) Foreground object after background subtraction with shadow detection and removal.

3 Tracking via Motion and Appearance Models

This section describes the techniques employed to track labelled moving objects through frames. We deploy a simple motion model based on first order Kalman Filter and an appearance model using colour histogram intersection and backprojection [17]. The advantage is this approach is its speed and simplicity of representation.

The motion model for the object is specified as follows. The bounding box and centroid coordinates of the identified object are used as the state and measurement variables in the Kalman Filter such that:

$$\mathbf{m} = \begin{bmatrix} x_1 & y_1 & x_2 & y_2 \end{bmatrix} \tag{1}$$

$$\mathbf{S} = \begin{bmatrix} x_1 & y_1 & x_2 & y_2 & \Delta x_1 & \Delta y_1 & \Delta x_2 & \Delta y_2 & w & h \end{bmatrix} \tag{2}$$

where **m** and **S** are measurement and state models and (x_1, y_1, x_2, y_2) represents the left, top, right, bottom boundaries of the bounding box. (Δx_1, Δy_1, Δx_2, Δy_2) represent the corresponding change in the values of boundaries in recent frame and (w, h) specify the overall width and height of the bounding box. These variables are used in the case where one bounding edge of a target is observable and its opposite boundary just becomes occluded. The occluded boundary can then be approximated by adding/subtracting the w or h state variable. Since we assume the object moves through image space at constant velocity, the new position at time $t + 1$ is predicted from the position at t by the equation:

$$\begin{pmatrix} x_1^{t+1} \\ y_1^{t+1} \\ x_2^{t+1} \\ y_2^{t+1} \\ \Delta x_1^{t+1} \\ \Delta y_1^{t+1} \\ \Delta x_2^{t+1} \\ \Delta y_2^{t+1} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1^t \\ y_1^t \\ x_2^t \\ y_2^t \\ \Delta x_1^t \\ \Delta y_1^t \\ \Delta x_2^t \\ \Delta y_2^t \end{pmatrix} + \begin{pmatrix} n_1^t \\ n_2^t \\ n_3^t \\ n_4^t \\ n_5^t \\ n_6^t \\ n_7^t \\ n_8^t \end{pmatrix} \tag{3}$$

The appearance model for the object is constructed as soon as the foreground blob is identified as a valid moving object. The object model is obtained by creating a colour histogram for the pixels considered part of the object. Each component of the colour model is quantised using a variable number of bits. Here, we use 5 bits (32 bins) for each colour component (H and S) and 4 bits for brightness (V) component. A smaller number of bins for V component is consistent with the fact that as the object moves, the brightness of the illumination varies according to its distance from the light source. The overall structure of the tracking

algorithm is illustrated in Fig. 2. The tracking algorithm works as follows. For each frame in the video sequence:

Step 1. Predict the new position of each tracked object using eq.(3).
Step 2. Calculate the most likely position of the object based on the prediction and the actual measurement associated with the object. If the measurement obtained varies significantly from the predicted position (e.g. in the case of static occlusion), use the predicted position.
Step 3. Use histogram backprojection technique[17] to identify the location of the object centroid based on colour model. Use the additional information obtained to validate and adjust the object location.
Step 4. Update the object state variable based on the object's most likely position.
Step 5. Update the colour model for the object if it is not subject to static or dynamic occlusion.

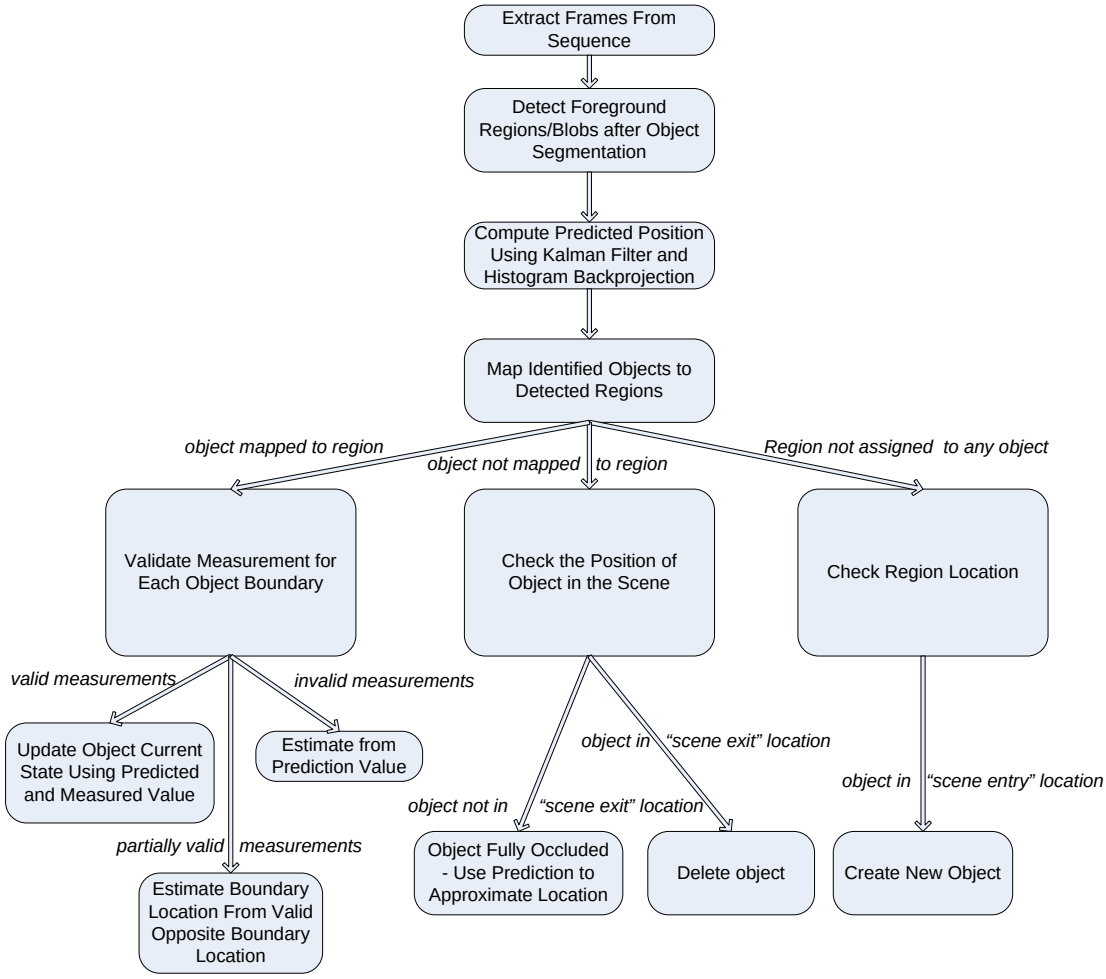


Fig. 2. Block diagram of the tracking algorithm

For objects that are dynamically occluded, extra processing is needed. In this type of situation, an object may partially or fully occlude the other objects. When different objects start to overlap in the scene and appear to move together, then all the constituent objects of the group are tracked as one large blob. Within the blob, the location of the object is approximated by the colour model using a histogram backprojection technique. This is accomplished as follows.

Assuming a pair of multi-dimensional histograms G and H each containing n bins, where G represents the target object model and H the 'background' image, we define a ratio histogram Ω between object and image as

$$\Omega_i = \min\left(\frac{G_i}{H_i}, 1\right) \tag{4}$$

Image values are then replaced by the values of Ω_i which they index. The backprojected image is then convolved with a mask which is approximately the size of the object's bounding box. The index with maximum value in the convolved image is the approximate location of the object. This additional information provides a cue for the object location within the larger blob (representing multiple objects with dynamic occlusion). When the objects separate from each other, the unique identity of each object is verified by intersecting the separated region model (i.e. histogram) with the histogram of all the objects (originally part of the dynamically occluded blob) in the current frame. The histogram intersection measure Ψ is defined as

$$\Psi = \sum_{i=1}^{n} \min(R_i, G_i) \tag{5}$$

where Ψ represents the number of pixels with the same colour in the two reference histograms, R is the colour histogram of the region, and G is the target object histogram. The object with the maximum value for Ψ is assigned to the region that is separated from the dynamically occluded group.

4 Modelling the Motion Path

4.1 Model Fitting

As for most tracking algorithms, the output is a set of (usually noisy) 2-D points representing the frame-to-frame reference location of an object tracked through the image space. We propose to model the overall shape of the resulting tracked points using a low degree polynomial. For more complex motions, the representation could be piecewise but we do not consider that here. The advantages are that a model representation will result in significant compression of the tracked data and it can also be used to index stored video sequences where the generic motion path of an object is of interest, e.g. to a CCTV operator.

We consider a RANSAC implementation [18] for the least squares (LS) approximation of a set of n data points (x_i, y_i) $(i = 1,2,...,n)$ by a polynomial $p_m(x)$ of degree $m < n$. The unknown m+1 coefficients a_k $(k = 0,1,...,m)$ can be determined by minimising the function E with respect to a_0, a_1, ...

$$E(a_0, a_1, \ldots, a_m) = \sum_{i=1}^{n} \left[y_i - \left(a_0 + a_1 x + \ldots + a_m x_i^m \right) \right]^2 \tag{6}$$

This is suitable in the case where x coordinate values are monotonically increasing. Where the values are monotonically increasing in y, we reverse the roles of x and y in eq.(6).

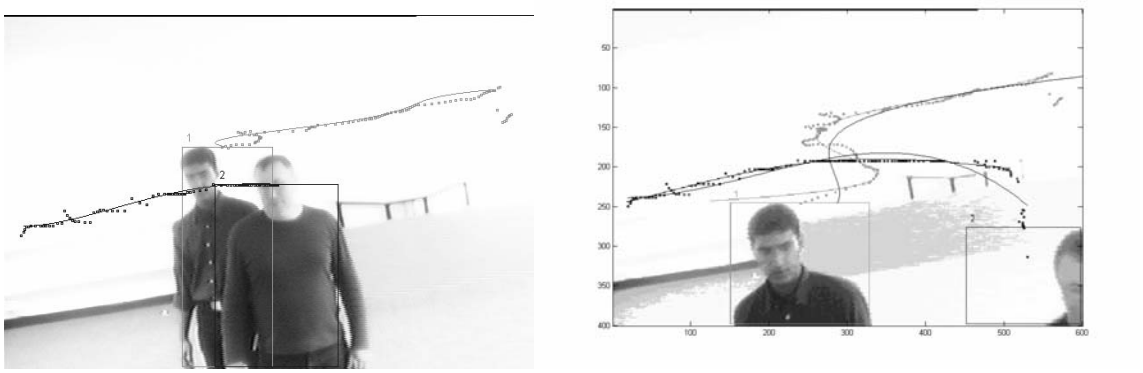
It is well known that least squares is a smoothing technique that is highly sensitive to gross errors. These outliers commonly arise in the tracking process due to object miss-classification and measurement error. RANSAC, on the other hand, is particularly suited to model fitting where the data is highly contaminated by outliers. Instead of using all the points to fit the curve (as in LS), it initialises the model with as small a data set as possible and then enlarges this set with consistent data where possible. When there are sufficient mutually consistent points, RANSAC then employs a smoothing technique such as LS to compute an improved estimate for the fit. This is demonstrated in Fig. 3 where the RANSAC result provides a more faithful representation of the motion path data. The intersection of the curves indicate the position at which object occlusion occurs. In most cases, $m = 3$ provides an adequate representation of the modelled trajectory.

4.2 Similarity Metric for Retrieval of Motion Paths

Since we wish to search and retrieve similar trajectories for tracked objects, it makes sense to index the video motion clips in a database using a model-based descriptor. Each tracked and labelled video object is therefore represented by the set of coefficients $\{a_i\}$ of the interpolated curve through its motion path. When a user invokes a query (motion path) which could be a free-hand sketch or set of trend points marked on a representative background scene, the coefficients are generated and compared to each of those in the database of clips using a Euclidean distance metric. The best matches are then retrieved in order of similarity. The similarity metric d is defined as

$$d(M_q, M_k) = \left\{ \sum_{i=1}^{m} (a_{iq} - a_{ik})^2 \right\}^{1/2} \tag{7}$$

where $M_q = \{a_{iq}\}$ and $M_k = \{a_{ik}\}$ (i = 1,..., m) denote the coefficient set for the query and stored motion path models respectively.



(a) (b)

Fig. 3. Fitting polynomials of degree 3 to motion paths. (a) Tracking up to occluding frames. (b) Comparison of LS and RANSAC model fitting. RANSAC produces tighter fitting curves.

5 Experimental Results

In this section, we present some results to indicate the effectiveness of the proposed techniques for tracking people through static and dynamic occlusions. We then generate motion path models and demonstrate how these can be used for object-based video indexing and retrieval.

The results shown in Fig. 4 demonstrate object tracking and interaction in the presence of static and dynamic occlusion. In Fig. 4(a), objects are tracked independently in the presence of static occlusion (represented by the table). Objects move towards each other and come into contact, thus merging into a single blob, but are still identified as separate objects shown in Fig. 4(b). Object 1 moves behind object 2 and is completely occluded as shown in Fig. 4(c). Two objects then separate and are identified and tracked with the correct label as shown in Fig. 4(d). The results demonstrate usefulness of appearance model since both objects are of similar colour distributions and object 1 is completely occluded by object 2 for some time as shown in Fig. 4(c).

(a) (b) (c) (d)

Fig. 4. Tracking of multiple objects through occlusions. (a) Objects are tracked independently (b) Objects come in contact with each other (c) Object 1 is fully occluded by object 2 (d) Tracking continues with correct labels on the objects.

Fig. 5 illustrates the results of using eq.(7) to search and retrieve object motion paths, similar to a user-defined query, from a surveillance database of motion clips. A partial or complete trajectory has been recovered for each successfully tracked object in the motion clip using the method described in section 4.1. A sample set of trajectories are shown in Fig. 5(a). Where the motion path is more complex and cannot be adequately modelled using a low-order polynomial, either this has been excluded from the database or stored only as a partial trajectory. The same is true of object paths where tracking has been lost due to complete occlusion.

Figs. 5(b)-(d) show the object motions retrieved for various user-specified queries. The stored trajectories (and hence motion clips) are ranked according to their degree of similarity to the query and the results indicate those inter-trajectory coefficient distances lying within a certain tolerance τ, where $d(M_q,M_k) < \tau$. The coefficient distance metric, though simple to compute, appears to give plausible results even in the case of a partial trajectory query, shown in Fig. 5(d). In future work, we intend to compare the performance of several different similarity metrics including Hausdorff distance measures (HDM). HDMs [6] are expensive to compute but have the advantage of working with point sets that are more suited to the case of

complex trajectory shapes. We also intend to investigate the addition of a velocity difference term to the metric since this important information is currently neglected.

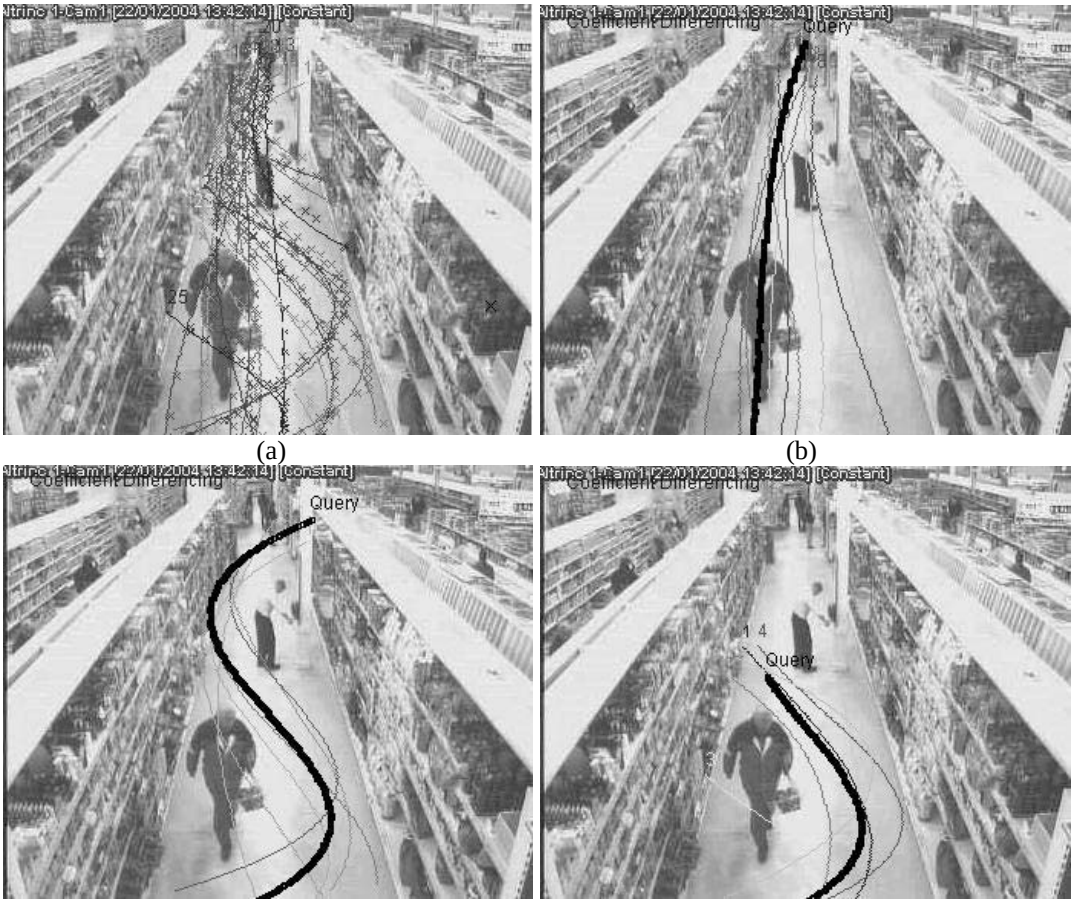


Fig. 5. Using a user-sketched query to retrieve similar motion paths. (a) Database of stored motion paths. (b)-(d) Highest ranked results for various queries based on closest distances in coefficient space- only those trajectories lying within a certain tolerance are displayed.

6 Conclusion

We have presented a simple but effective approach for tracking multiple objects through static and dynamic occlusions. It requires colour images as this is vital for shadow detection and maintaining an object-based appearance model used for disambiguating merged regions during occluding frames. If the predicted object position varies significantly from the measured position based on the current frame, Kalman Filtering is used to estimate the new location. The prediction is then adjusted after performing histogram intersection of the object in the current frame.

Motion trajectories are then modelled via polynomial interpolation adopting a RANSAC approach for ensuring the generated motion paths are resistant to outliers. The coefficient descriptors prove to be a useful index key into a database of video clips representing object motions. A user-defined query can be sketched as a means of retrieving similar motion events which makes this a useful tool for surveillance-based intelligent behaviour analysis.

Acknowledgements

Mr Shehzad Khalid would like to thank the Higher Education Commission of Pakistan for financial support of his postgraduate studies.

References

[1] J. Aggarwal and Q. Cai, "Human motion analysis: A review," Computer Vision and Image Understanding, vol. 73, pp. 428--440, 1999.

[2] L. Wang, W. Hu and T. Tan, Recent developments in human motion analysis, Pattern Recognition, Volume 36, Issue 3, March 2003, Pages 585-601

[3] B. Georgis, M. Maziere, F. Bremond, M Thonnat, "A video interpretation platform applied to bank agency monitoring, Proc. IEE Intelligent Distributed Surveillance Systems (IDSS-04), February 23, 2004, London, UK, pp. 46-50.

[4] D. Makris and T. Ellis, Path detection in video surveillance. Image & Vision computing, 20 (2002) 895-903

[5] N. Johnson and D. Hogg "Learning the distribution of object trajectories for event recognition", Image & Vision Computing, 14 (1996) 609-615.

[6] J. Lou, Q. Liu, T. Tan, Weiming Hu, Semantic Interpretation of Object Activities in a Surveillance System. 16th International Conference on Pattern Recognition (ICPR'02) Volume 3 August 11 - 15, 2002

[7] Y.Jung, K. Lee, Y. Ho, Content-Based event retrieval using semantic Scene interpretation for automated traffic surveillance, IEEE Trans. Intell. Transport. Syst. 2, 151-163, 2001.

[8] W. P. Berriss, W. G. Price and M.Z. Bober, “The Use of MPEG-7 for Intelligent Analysis and Retrieval in Video Surveillance” Proc. IEE Intelligent Distributed Surveillance Systems Symposium (IDSS-03), pp. 8/1 – 8/5, London, February 25, 2003.

[9] A. J. Lipton, J. Clark, P. Brewer, A. Chosak, P. Venetianer, Object video forensics: Activity-Based Video Indexing and Retrieval for Physical security, IDSS-04, February 23, 2004, London, UK, pp. 56-60.

[10] Haritaoglu, D. Harwood, and L.S. Davis. W4: Real-Time Surveillance of People and Their Activities. IEEE Tras. On Pattern Analysis and Machine Intelligence, 22(8):809-830, August 2000.

[11] Haritaoglu, I., D Harwood & L. Davis (1998). W4S: A Real Time System for Detecting and Tracking People in 2.5D. European Conference on Computer Vision, 1998

[12] C. Wren, A. Azarbavejani, T. Darrell, and A. Pentland, “Pfinder: Real-Time Tracking of the Human Body”, IEEE Trans. Pattern Analysis and Machine Intelligence vol.19, no. 7, July 1997.

[13] Robert T. Collins, Alan J. Lipton, Hironobu Fujiyoshi and Takeo Kanade, "Algorithms for Cooperative Multisensor Surveillance," Proceedings of the IEEE, Vol. 89(10), Oct 2001, pp.1456-1477.

[14] Robert T. Collins, Alan J. Lipton, Takeo Kanade, Hironobu Fujiyoshi, David Duggins, Yanghai Tsin,David Tolliver, Nobuyoshi Enomoto, Osamu Hasegawa, Peter Burt1 and Lambert Wixson1. “A System for Video Surveillance and Monitoring”. The Robotics Institute , Carnegie Mellon University. CMU-RI-TR-00-12

[15] J. Kang, I. Cohen and G. Medioni, “Tracking Objects from Multiple Stationary and Moving Cameras”, Proc. IEE Intelligent Distributed Surveillance Systems (IDSS-04), February 23, 2004, London, UK, pp. 31-35.

[16] R. Cucchiara, C. Grana, M. Piccardi, A. Prati, and S. Sirotti, “Improving shadow suppression in moving object detection with HSV colour information ”, in Proc. of the 4th International IEEE Conference on Intelligent Transportation Systems, August 25-29, 2001, Oakland, CA, USA, pp.334-339.

[17] M. J. Swain and D. H. Ballard. Color Indexing. International Journal of Computer Vision, 7(1):11-32, 1991.

[18] M. A. Fischler, R. C. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Comm. of the ACM, Vol 24, pp 381-395, 1981.

Content Based Access for a massive database of human observation video

L. Joyeux[†], E. Doyle[‡], H. Denman[†], A.C. Crawford[†], A. Kokaram[†], R. Fuller[‡]
[†] Dept. of Electronic & Electrical Engineering
[‡] Dept. of Psychology
Trinity College Dublin, Ireland

Abstract

We present in this paper a CBIR system for use in a psychological study of the relationship between human movement and Dyslexia. The system allows access to up to 500 hours of video and is an example of a scientific user context. This user context requires 100% accurate indexing and retrieval for a set of specific queries. This paper presents a novel use of interactive visual and audio cues for attaining this level of indexing performance. Furthermore, the issue of motion estimation accuracy in the presence of compression artifacts is explored as part of the data integrity storage problem. In addition, content based motion analysis techniques accurate enough to parse sequences on the basis of motion and objectively evaluate that motion are investigated. The tool allows Psychologists to undertake a study that would previously be impractical and the paper presents a number of lessons gained from the ongoing work.
Keywords: content retrieval, tracking, video retrieval, dyslexia, human body motion

1 Introduction

Developmental dyslexia (also known as 'Specific Learning Difficulty' or SLD) is a serious societal problem. It affects 8% of the population - that implies 480,000 people in Ireland alone. It is not caused by lack of intelligence, emotional disturbance, poor teaching, family difficulties or social problems. If left untreated, a child can develop poor self-esteem and confidence and fail to master even the basics of reading, writing and arithmetic. These children require a high level of educational resources and have the strong potential to continue causing problems in the school system. The cost of dyslexia to the society infrastructure as a whole is therefore enormous. There is currently no reliable diagnosis available to identify dyslexia until the child has demonstrated a failure to read after persistent attempts (usually at the age of 8 or 9). Remedial therapies are based on intensive practice of basic language skills and so occupy a large amount of teacher resources (often on a one to one basis). More often than not the child never reaches his or her appropriate reading age. McPhillips et al. [4] presented the notion that there is a quantifiable connection between Dyslexia and the retention of certain reflex movements. Dyslexia is now no longer seen solely as a problem generated by a higher-order brain malfunction, but as possibly a treatable disorder with a physiological rationale. Evidence was provided that in Dyslexics, certain *primary reflexes* [3] are retained. In subsequent development, these reflexes become integrated into postural reflexes to allow the child to progress to the next stage of movement. But in dyslexics, early reflexes may persist. The work of McPhillips et al. also indicates that Dyslexia can be treated by retraining the central nervous system by slowly repeating these movements. Hence the connection between the treatment of Dyslexia and a movement therapy. The **DysVideo** project at Trinity College was set up to observe the development of 400 children aged below 6 years. Each child is observed through 3 sessions of 20 minutes, each 6 months

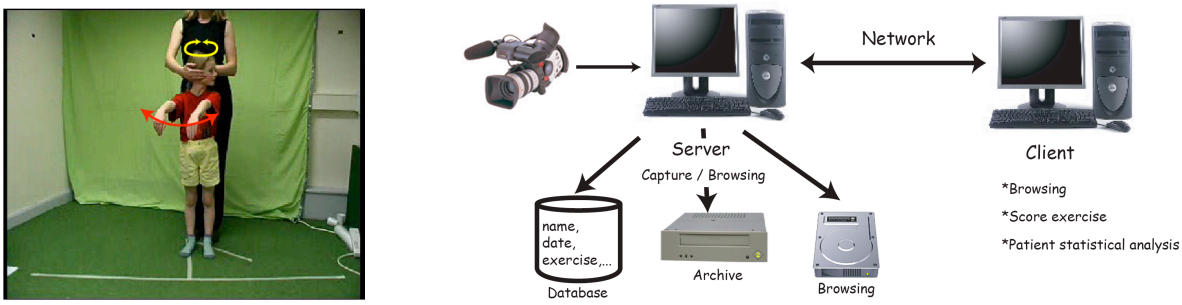


Figure 1: Left: Example of exercise to trigger the ATNR primary reflex. Right: Client/Server System Architecture.

apart. The session is composed of 14 exercises that are designed to trigger each of four primary reflexes. For example, Fig. 1 shows the movement designed to trigger the ATNR[1, 2] primary reflex. In this movement, the child stands with arms held out in front. The supervisor then turns the subject's head to each side for 5 secs. The arms may follow the head movement or drop. The amount of movement made by the arms gives one clue about the severity of the retained reflex. In a non-dyslexic child, the arms should not move.

The idea is to video each session and then to allow the Psychologists access to the recorded sessions for offline subjective assessment of the degree to which each child meets or fails to meet the required movement template. However, there are clear difficulties that can only be addressed by content-based analysis and indexing as follows.

1. Although sessions may last 20 minutes, the actual measurable information may only be about 5 minutes. This is because much of the time is spent making the child comfortable and setting up each test. Furthermore, children under ten years old are not known for good attention spans, thus intrusive behaviour may cause the session to last even longer. Therefore, it is extremely time demanding for Psychologists to manually locate the useful information from the massive amount of data recorded. *A process is needed to index the start and end of each session automatically.*
2. The movement evaluation as is currently carried out is subjective. Furthermore, without a video record there is no way to cross check retrospectively between different evaluators. Indeed, direct observation requires some training and the movement instance can simply be missed by the observer. Consequently, maintaining a database of scores and movement sessions is essential. *This implies identifying the child and each session uniquely.*
3. Objective movement evaluation is required. This could be achieved by automated tracking of the movement of the limb in question and then attempting to correlate these measurements with a predetermined template motion. However, most trackers require human initialisation. Given the huge database of material within which the usable material is just a fraction, this is impractical. *A mechanism must be found to directly index the active portion of each experiment in order to engage an automated tracker.*

Each of these problems is now addressed in turn.

2 System Architecture

Fig. 1 shows that the system architecture has a server/client structure. The server performs the capture, indexing and analysis of video sequences, and can also be used as a browser. The different clients browse the captured video sequences remotely. Analysis includes sequence compression and content retrieval.

2.1 Video streaming and compression

A DV camera with an output at a constant bit-rate of 26.4Mbit/s was used. Given that the total video to be stored is about 500hrs; this is equivalent to about 5.8 Terabytes. To keep storage costs low, the DV video stream is compressed using MPEG4 with a 1Mb/s bit-rate. This setting gives comfortable viewing for the human evaluators. The compressed sequences are stored on a disk for "video on demand". Compressed sequences are easily stored in fast access hard drives, e.g. 500hrs of video require 225GB, which is currently easily obtained. Sequences are compressed in real time at the end of the day's recording sessions. For practical (space) and reliability reasons it is more sensible to restrict the recording sessions to one camera only and to avoid streaming direct to disk. However, 1Mb/s bit-rate does not provide a good enough quality for motion analysis. There are two possible solutions to this problem. (1) The DV media should be processed immediately for motion upon capture. (2) Further attention should be paid to the problem of compressed bit rates required for scientific video analysis. Development of motion analysis techniques is still an active research area and it is not sensible to rely in the future on motion estimates generated once upon capture only. Therefore it is useful to consider the problem of choosing a bitrate which gives little effect on motion estimation, yet yields good enough compression for long term storage. From our experience, the chosen motion estimation process [11] operates properly above a bit-rate of 128Kb/s. Consequently, sessions are compressed at a bit-rate of 2Mb/s, to have a good safety margin, using a MPEG2 codec.

The video from each session is streamed directly into a single file that then must be indexed to indicate the important portion of that file. The system does not create multiple files for each session as it is simpler to maintain a basic database. Thus, key or index files are associated with each session video stream. The creation of the index is discussed below.

2.2 Interactive Audio Markers

The user is asked to use a handheld computer to create tones which are used to indicate the start and end of each exercise (2 digits), as well as the ChildID (6 digits) etc. DTMF tones (Dual Tone Multi Frequency), were used because they are better differentiated from speech and they code 10 digits and two symbols # and *. The symbols are used to mark the exercise end or an error, respectively. In the first recorded sessions, the DTMF sound was played, near to the camcorder. Unfortunately, classification was hampered by noise such as laughter. Nevertheless, the detection was successful in more than 95% of the cases. To achieve 100% accuracy, the DTMF and room audio was recorded on separate channels of the stereo sound camcorder system, thus the detection becomes trivial and 100% accurate. The detection requires the discrimination of two frequencies simultaneously (row/low and column/high) [7]. and consists of 3 steps: 1) measure and threshold of the energy on all DTMF frequencies, 2) identify the key pressed 3) group a set of keys to get the exercise number, the child ID or symbols # and *.

2.3 Browser

The browser allows access to a particular exercise for a given session and child as well as scoring and comparison with other similar sequences. It uses MPEG4 compressed video sequences (1Mbit/sec) and a database (the indexing or key file), which contains time codes to allow random access to particular sessions. The GUI is shown on Fig. 2. It allows the user to watch three different exercises; a window is split horizontally or vertically, when an exercise is added (right-top bottom). Three sliders are used to navigate throughout the exercise, allowing the user to repeatably view the important sections of the session. On the right part of the window, a tree displays information on all exercises taken by a particular child in addition to current user

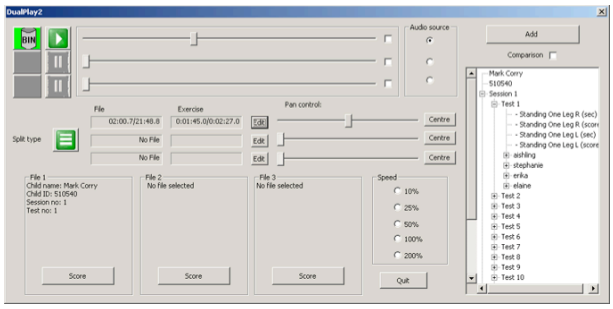


Figure 2: Browser interface. The user can browse and score up to 3 exercises simultaneously.

scores. Other user scores can be displayed depending on access rights. The browser also allows computations to derive score statistics across individuals.

3 Content based analysis

The key file allows the indexing of the start and end of each exercise, but there may be some time between the experimenter inputting audio tones and the actual starting of the movement experiment. In order to attempt to develop automated motion analysis assessment and explore how well this correlates with the subjective assessment of the psychologists, a mechanism must be found to identify exactly when the exercise actions begin. Efforts are currently concentrated on the ATNR (Asymmetrical Tonic Neck Reflex) exercises. The idea is to use skin detection to locate limbs, and then to use the rough flesh information in two ways. Tracking of the centre of gravity (CoG) of the region in the whole frame allows the start of each action to be indexed. Then, closer body localisation can be carried out again using the flesh detector. This time the temporal indicators from the CoG analysis can be used to instantiate a tracker for the relevant limb.

3.1 Skin tone detection

Skin detection is a common technique used, e.g., in face recognition [5, 6]. The idea is to associate pixels containing skin with a particular colour distribution that is empirically built from observed images. The best detection quality was obtained using the skin detector described in [9]: a pixel is flagged if "$(R > 95)$ and$(G > 80)$ and$(B > 40)$ and$(R > G)$ and$(R > B)$ and$(R - \min(G, B) > 10)$ and$(R - G > 15)$". This detector avoids selection of pure red or gray pixels. Just before applying this detector, a global colour adjustment is performed to compensate the global colour variations (for unknown reasons, the image becomes randomly blue). Using the carpet colour as the reference colour is the carpet, we simply subtract the colour reference to the colour estimated. A typical result is shown in Fig. 3. In practice all exposed limbs are detected except in instances where the limb colour is changed due to lighting and shadow. Few false alarms also occur in the presence of rich reds. This problem is resolved simply by recommending that subjects do not wear red clothing. The detector works in at least 95% of the cases on 100 sequences of 90s.

4 Analysis of ATNR (Schilder test) exercise

This exercise is described in Fig. 1. The aim of the analysis is to track hand positions over the sequence. The analysis is challenging because of many degrees of freedom of arms and hands and unreliable framing of the child in the field of view. Moreover, children do not co-operate actively with the exercise and this implies that less than 50% of the sequences correctly match the exercise template.

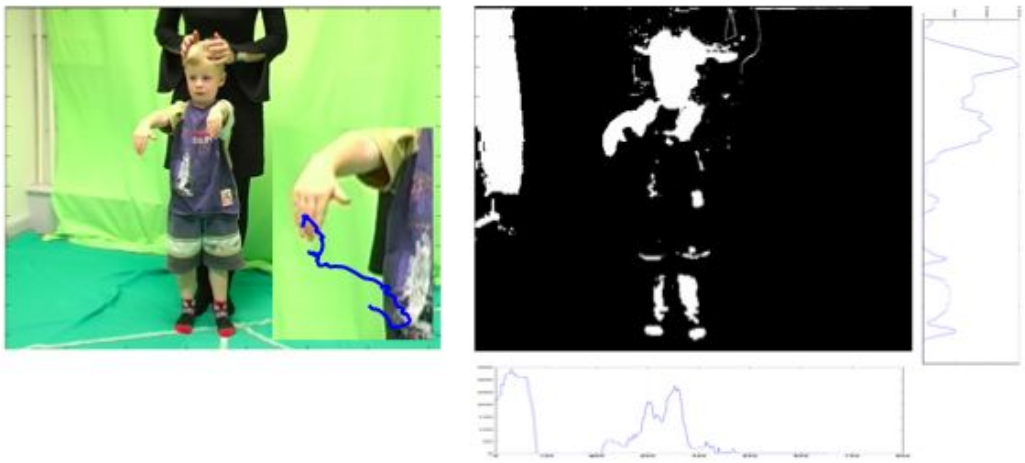


Figure 3: Left: Original with a red indicator showing result of hand detection and an estimated motion track in blue, Right: Result of skin detection and horizontal and vertical projections showing body localisation.

4.1 Hand localisation

To build a rough localisation of hands we exploit the starting conditions of the ATNR experiment: hands down, arms up to shoulders and straight forward (e.g. image Fig. 3). Detected areas of skin can then be associated with limbs. Because both arms are detected, a vertical projection of the skin detection image gives the body range along the X axis. Using this narrowed range, a horizontal projection gives the bottom position of the hands with the search constrained in the top half of the image. Fig. 3 shows both projections: we see immediately that it is easy to locate body boundaries.

Once the top part of the body is localised, hands are associated with the lowest parts of that skin/body mass, with small objects removed using an erosion operator with a mask size of 10×10. All possible point pairs p_l and p_r, for left and right hands respectively, are considered. The pair that maximises $p_l(y) + p_r(y)$ is chosen as the hand detected positions.

We tested the hand localisation on two sequences of duration 1.5 mins which is the total extent of each exercise. During the exercise, there are only a few seconds in which the hands are detectable in the expected pose. The hand position is working in 80% of the cases (both hands are correctly localised).

4.2 Analysis

The localisation feature presented above can then be exploited in two ways to provide the Psychologists with a possible objective measure of motion. First of all, there is a need to locate efficiently in time the start and end of each exercise instance. Having done this, hand detection can then be used to initialise a tracker [8] or optic flow field estimation can be used to generate some index of fit to an expected template optic flow field. This paper does not present any results of motion measurement as the study is still in an initial phase. However, the body and hand localisation feature are important features for content access when coupled with simple motion information.

Again exploiting the user context, the ATNR exercise begins with the experimenter's hands moving between head and arms as this is a training period for the child subject. Thus vertical movement is an indicator of the specific start point of this exercise. A simple feature to index

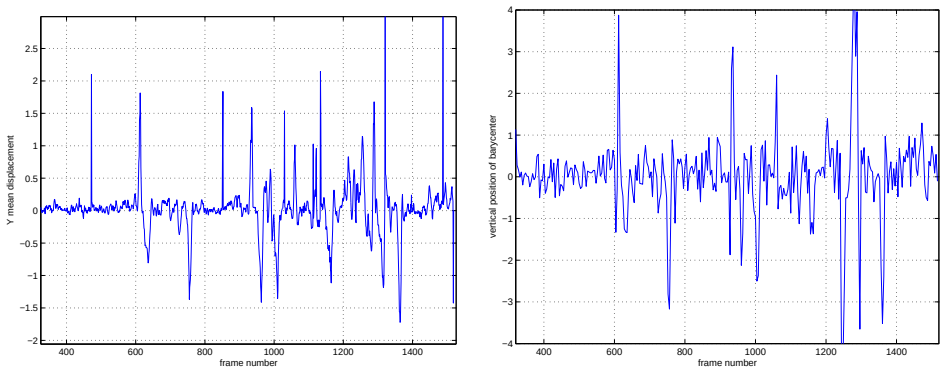


Figure 4: Mean vertical motion of limbs (on the left) and vertical position of the center of gravity (on the right)

this information therefore is the centre of gravity of all the skin detected in a particular frame. This is in fact related to a geometric moment, a feature we have exploited successfully in the past for sport events [8]. A track of the vertical displacement of this Frame-CoG is shown on the right in Fig. 4. Explicit hand tracking can also yield similar information. Experiments were carried out using a primitive tracker. A hand reference point is assigned which is expected to be at the centre of the palm. Optic flow components within a disc of radius 12 pixels around this reference point are then averaged to estimate the motion into the next frame. In the next frame the point is corrected to be at one half disk radius away from the bottom detected hand portion. Furthermore, to avoid lateral drift, the horizontal position of the reference point is corrected to be at the centre of gravity of the detected hand portion within the disk radius. This correction is at most 3 pixels in practice and hence problems do *not* arise with the hand portion moving outside the disk radius. A track of the hand over 100 frames is shown superimposed in Fig. 3. Both these features show points of action indexed by large positive motion followed by large negative motion as expected. They do not agree entirely however, since hand tracking is explicit while Frame-CoG is implicit. In practice, we find that using the Frame-CoG feature the start of 90% of ATNR exercises is successfully located, while yielding 25% false alarm rate. The false alarm rate is high due to the crude feature extraction step. Nevertheless, given a manual initialisation, hand tracking is accurate and over 3 minutes (the full extent of the exercise for two realisations) (as stated in the previous section) there is no loss of lock.

5 Analysis of ATNR (ayres test) exercise

In this exercise, the child is on all fours, head turned to the camera. The supervisor, seated on one side of the child, turns the head of the child left and right for 5s (see Fig. 5). This movement may trigger a tremor or a bend of the arms. The goal is to measure the angle of the forearm as well as the angle variation and speed for each arm. We have to detect the individual realisation of the exercise since the movement is repeated several times (eyes open and then closed), without inserting marks. The following process is illustrated in Fig. 5. As in the previous exercise, we apply first the skin detector to select both arms. This selects arms, hair and supervisor hands. Then a bounding box containing both arms is estimated to localise further processing. The bounding box is estimated in two steps. First, a vertical projection gives the vertical position of arms, we search for the two extrema that correspond to individual arms since they are oriented vertically. Second, using the previous vertical boundaries, a horizontal projection is performed. When this projection is scanned from the bottom to the top, this indicates a direction from fingers to upper arms. The maximum of this curve corresponds to hand location since the width

of the hand is larger than that of the arm in the view. The vertical extent of the bounding box is taken as three times the hand height. This is a weak hypothesis but valid since since body ratio is relatively constant. The accuracy of the bounding box is 80% on 30 sequences of 90s. The next step is to estimate the angle of arms and detect each exercise realisation. To do this we fit a line using Andrew's sine robust estimator [10] with sine width set to estimated arm width. This estimator gives a better line fitting than Hough transform or least squares because the arm is not a straight line (due to geometric projection) and because legs and arms are articulated and may merge as a single region. The minimisation implementation is performed using the bisector method by limiting the angle search to $[-\pi/8, \pi/8]$ and origin to $[-w_h, w_h]$ where w_h is the hand width. Line fitting is performed for both arms with the origin set at the corresponding hand location position estimated during the bounding box step.

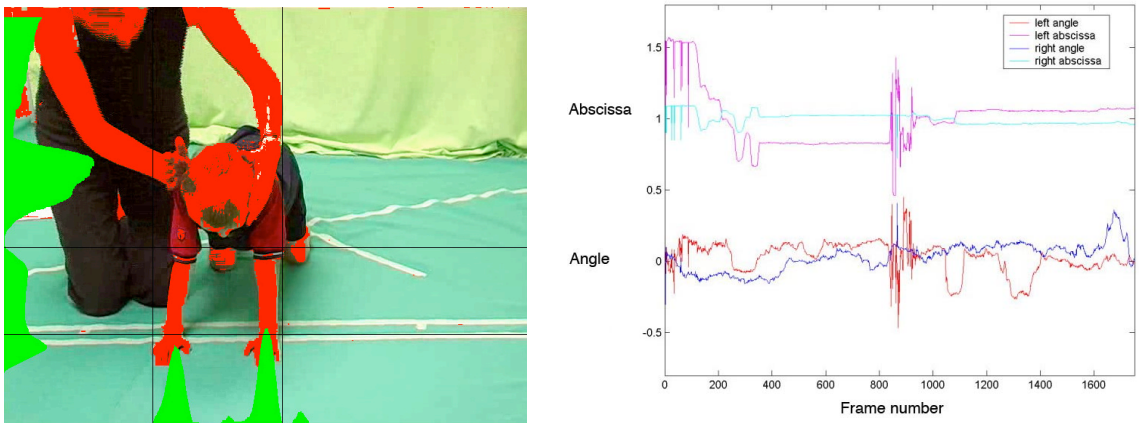


Figure 5: Left: ATNR ayres exercise. Skin detected pixels are in red. The green curves represent the vertical (on the bottom) and horizontal (on the left) projections. Right: result of angle and abscissa estimations for both hands.

In Fig. 5 is shown the result of the estimation. The two lower curves represent the angle and the two upper curves are the horizontal position (normalised to fit in the plot) of both hands. From position curves, we can distinguish the actual conduction of the experiment from the preparation stages. The horizontal hand position has to be constant during experimentation since hands are fixed on the ground. Any variations, during a period of few seconds, indicates preparation of the child and not actual conduction of the experiment. Discrimination between preparation and realisation is therefore performed by fixed threshold on the movement curves (called $mc(t)$) : realisation is when $|\text{median}(mc(t), 10) - mc(t)| < 3.5$ where $\text{median}(x, y)$ is the median on x on a window of length y. From the angle curves are extracted the mean, μ_φ, and standard deviation, σ_φ (preparation stage intervals are ignored). Speed is parameterised with mean, $\mu_{\varphi'}$, and standard deviation, $\sigma_{\varphi'}$, on the absolute value of the derivative of the angle (the angle is filtered to reduce the noise by median filter over 20 images). These features are used for motion assessment and are currently under investigation. The method presented in this section has being ran successfully for 80% of the cases for a set of 30 sequences of 90s each. Failing cases, mainly related to line estimation, are due to bad framing (the child does not fit the image), objects overlapping arms.

6 Final Comments

This paper has presented a new tool for Psychologists that exploits content retrieval technology in research in motor reflexes in Dyslexia. The system allows video on demand as well as

automated indexing and video analysis. As the final users are psychologists and not computer specialists, simplicity and robustness are paramount. The work has highlighted some interesting implications for massive databases for scientific use. First of all, storage requirements may not enable the best quality material to be stored. This limits the quality of scientific analysis of the picture material. Having two streams of data with two levels of compression appears to be the best compromise. We have presented new results exploring what the breakdown level is for motion accuracy applied to compressed sequences. Secondly, by exploiting the user context, the system is able to deploy 100% reliable indexing. This is imperative for use in scientific investigation. The use of interactive audio cues is novel and allows 100% reliability to be achieved. New features that yield position information for identifying the start of stylistic movement have also been presented. In this user context, explicit tracking with automated initialisation is possible and this yields powerful information for indexing. Finally, it is noteworthy that this project has the potential to have a major impact on human observational studies. This project allows for a deep level of data access without the need for 3D observation, by exploiting the user context. Our current work focuses on quantitative evaluation of motion characteristics in dyslexic children. Video sequences showing indexing and parsing output as well as the browser interface are shown at www.mee.tcd.ie/∼sigmedia/dysvideo.

References

[1] Goddard, S., "A Teachers Window into a Child's Mind, A non-invasive approach to solving learning and behaviour problems", Fern Ridge Press, Oregon, 1996.

[2] Holt, K.S. "Child Development: Diagnosis and Assessment", Butterworth-Heineman Ltd, 1991.

[3] Illingworth, R.S. "The Development of the Infant and Young Child: Normal and Abnormal", Churchill Livingstone, 8th Edition, London, 1983.

[4] McPhillips, M. Hepper, P.G. & Mulhern G., "Effects of replicating primary-reflex movements on specific reading difficulties in children; a randomised double-blind, controlled trial", Lancet 2000: 355: 537-41.

[5] Albiol, A., Torres, L., Delp, E.J. "Optimum color spaces for skin detection", Image Processing, 2001. Proceedings. 2001 International Conference on ,Volume: 1 , 7-10 Oct. 2001 Pages:122 - 124 vol.1

[6] Chai, D., Ngan, K.N. "Face segmentation using skin-color map in videophone applications", Circuits and Systems for Video Technology, IEEE Transactions on ,Volume: 9 , Issue: 4 , June 1999, Pages:551 - 564

[7] Felder, M.D., Mason, J.C., Evans, B.L., "Efficient dual-tone multifrequency detection using the nonuniform discrete Fourier transform" Signal Processing Letters, IEEE , Volume: 5 , Issue: 7 , July 1998 Pages:160 - 163

[8] H. Denman, N. Rea, A. Kokaram, "Content Based Analysis for Video from Snooker Broadcasts, " Journal of Computer Vision and Image Undrestanding - Special Issue on Video Retrieval and Summarization, Vol 92, Issues 2-3, Pages:176–195

[9] Gomez G., Morales E., " Automatic feature construction and a simple rule induction algorithm for skin detection" Proc. of the ICML Workshop on Machine Learning in Computer Vision, 31-38. 2002

[10] Black M.J., "Robust Incremental Optic Flow" PhD thesis

[11] Anil Kokaram, "Motion Picture Restoration", Springer Verlag, ISBN 3-540-76040-7 1998

Automatic Blackjack Monitoring

W. Cooper
Department of Computer Science
Trinity College
Dublin 2
wesley@ireland.com

K. Dawson-Howe
Department of Computer Science
Trinity College
Dublin 2
Kenneth.Dawson-Howe@cs.tcd.ie

Abstract
This paper describes a system for monitoring blackjack play based on the video feed from a single low resolution overhead camera system. The system successfully monitors play in limited tests. Given the resolution of the imagery new techniques for extracting and identifying the cards were required and are presented in this paper.

Keywords: Image processing, Calibration, Playing Card location and recognition.

1 Introduction

1.1 Background

Surveillance costs in casinos are significant and most game tables are monitored by many cameras, which, in general, are continually observed by security personnel. For example the Greektown Casino in Detroit has over 1100 cameras which are monitored by trained security staff on a wall of colour video monitors. There are products which attempt to automate some of this monitoring activity, such as the *MP21* system [1].

The *MP21* system provides a complete table (rather than working with existing tables) incorporating a magnetic card stripe reader, optical reading and accounting for all chips in the dealers chip tray, optical imaging system for reading cards as they leave the card shoe and optical monitoring of the position of every card and chip on the table. The system requires the use of special cards using patented WinMark™ technology.

Although these commercial systems exist there appears to be a serious lack of published work in the areas of card recognition and automated surveillance of card games. A search of computer vision papers was unable to locate any published work in either field.

1.2 Overview

This paper presents a system we have developed specifically for monitoring blackjack from a single relatively low resolution overhead camera. The problem was addressed in a series of stages which are detailed in the paper.

1. The video image of the table was rectified so that the view appeared as though it were from directly overhead. The important parts of the table (the bet squares and chip trays) were also automatically identified (See Section 2).
2. The cards are located as they are placed on the table (See Section 3).
3. The cards are recognized based on the picture or the number of dots (See Section 4).

One major advantage of the system described in this paper over existing commercial alternatives is the fact that it will be extremely low cost as the system work with the existing table (and security cameras).

2 Calibration

2.1 Geometric Correction of the table image

A geometric correction was required so that the image of the table to be used for processing appeared with the chip tray aligned with the horizontal image axis as though the image were taken from directly above the table. See Figure 1.

(a) (b) (c)

Figure 1. Image of a blackjack table taken from an overhead camera (a), with the calibration square (b) and geometrically corrected for processing (c). In normal casino situations it is usual for a surveillance camera to be virtually directly overhead lessening the need for this correction.

2.2 Identification of the important table components

From the perspective of monitoring the game, the chip tray and the bet squares (See Figure 2(a)) are the most important features on the table. These are located automatically as follows. All colours except the cloth colour (See Figure 2(b)) were filtered out. An opening is then applied (to a binary version of the resultant image) in order to ensure a continuous border around the bet squares. The chip tray is easily identified as it is appears a large hole in the cloth colour which has a very high value for rectangularity as defined in [2]. The bet squares are located by performing a statistical analysis of the areas of cloth colour which are completely encircled within markings on the tables (See Figure 2(c)). Values for area, width, height, rectangularity and elongatedness (as defined in [2]) are computed, and the bet squares are identified simply through similarity. The successfully located chip tray and bet squares are shown in Figure 2(d).

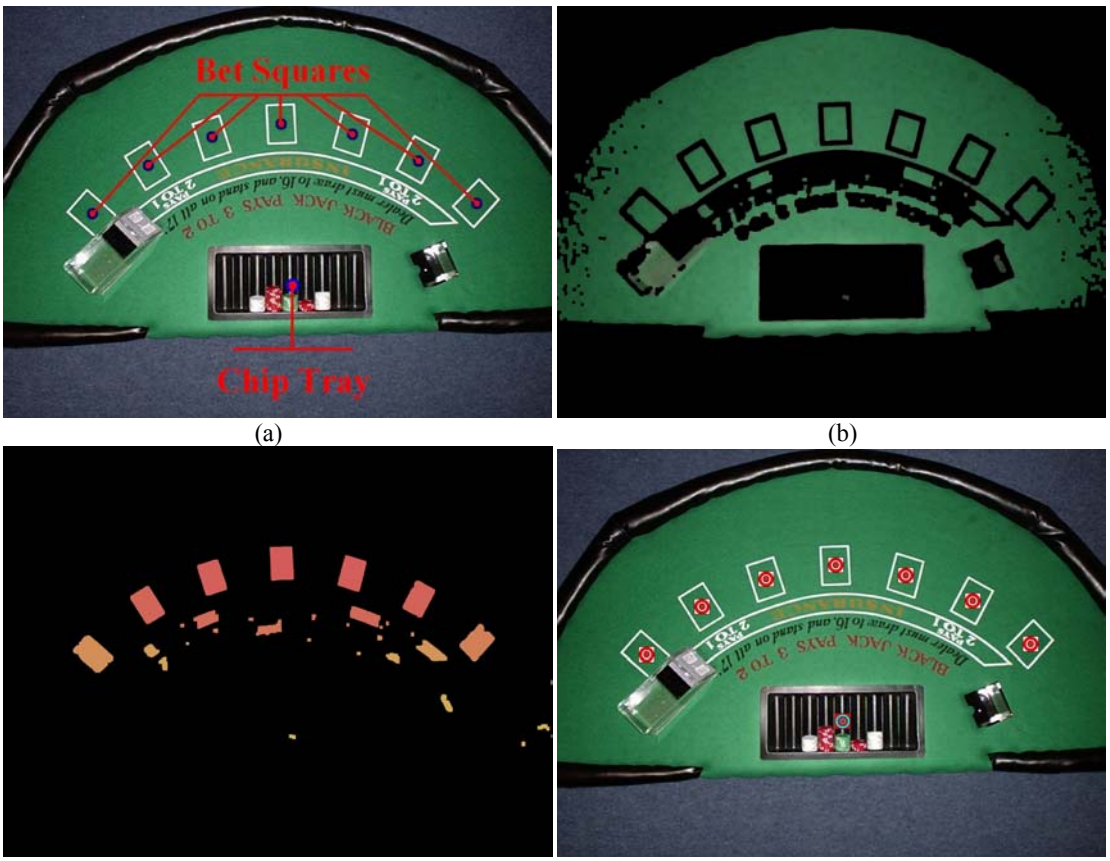


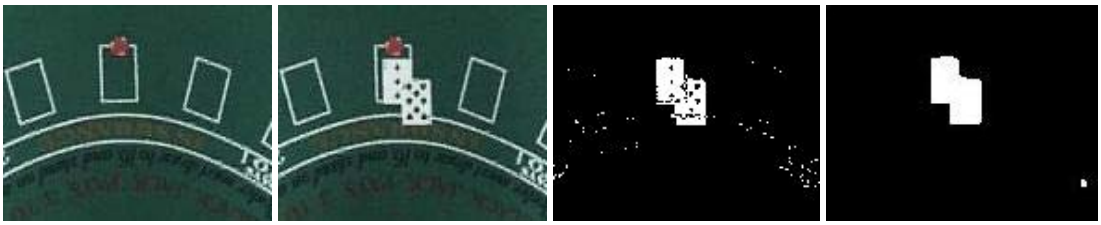
Figure 2. The chip tray and bet squares are shown (a), the cloth image after the opening operation (b), the regions of cloth colour which are completely encircled by markings on the table

3 Card Location

Having established the positions of the bet squares and the chip tray on the table, it is possible to identify regions of the image which can be associated with particular players or the croupier (See Figure 3). Each of these regions is monitored in order to locate *stable* frames (i.e. those in which there is no motion, such as a hand being present). These frames are then analyzed to determine any changes from the previous *stable* frame for that region.

Figure 3. The areas which are associated with each player and with the croupier/dealer. Note that the positions in which cards are placed by the croupier are well defined with respect to the bet squares (for the players) and with respect to the chip tray (for the croupier).

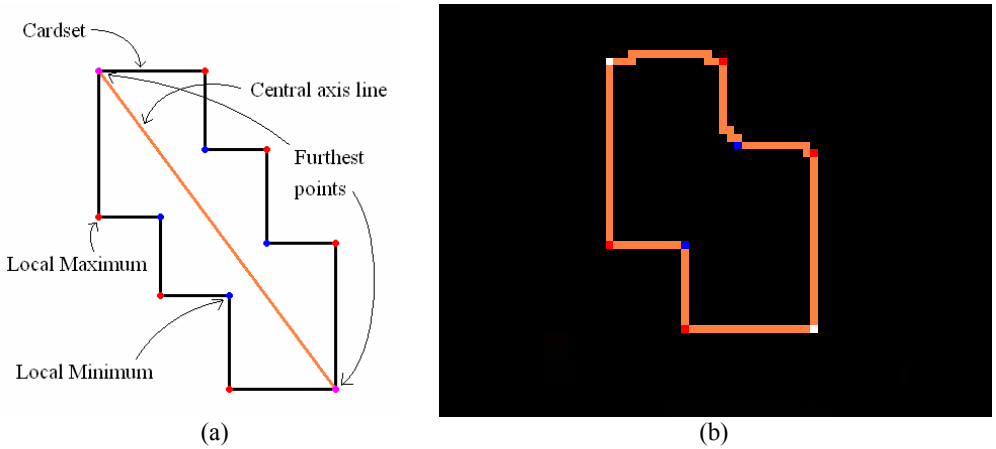
In a typical hand a number of events occur. Firstly the players place some chips on the edge of their bet squares. Once placed the system should be able to detect a new *stable* image (See Figure 4(a)) for each player region which is used as the *background* image when locating cards. Then, once all players have placed their chips, the croupier begins to deal cards and as each card is placed yet another *stable* image is determined for each region (e.g. See Figure 4(d)). Each new stable image is analyzed to see if any new cards are present. This is done by taking a difference between the latest *stable* image and the *background* image, thresholding the result (See Figure 4(c)), performing a closing followed by an opening (See Figure 4(d)) and finally locating all of the cards present. It should be born in mind that the placement of a new card can result in some minor movement of the previously placed cards and hence all cards are located during this processing (rather than just the most recently placed card).



(a) (b) (c) (d)

Figure 4. Cards and chips placed relative to the bet square of one player (a), a background image prior to the cards being placed on the table (b), a thresholded version of the difference between these images (c), and the result of applying a closing followed by an opening to the thresholded image.

Locating the individual cards is done by analyzing the outline of the binary region determined previously. This outline is determined using a Roberts cross operators and stored as a boundary chain code. A central axis line is determined (See Figure 5 (a)) between the top left-most corner of the first card in a set and the bottom right-most corner of the last card in a set. These points are the two edge points in the boundary chain code which are furthest apart (due to the way in which cards are placed in Blackjack). Local maxima and minima are then determined relative to the central axis line and from these the coordinates of the final can be calculated (one corner of the card is the bottom right-most corner, two of the other corners are the nearest local maxima, and the final corner can be calculated by intersecting the lines formed by those two local maxima and the next two local minima.

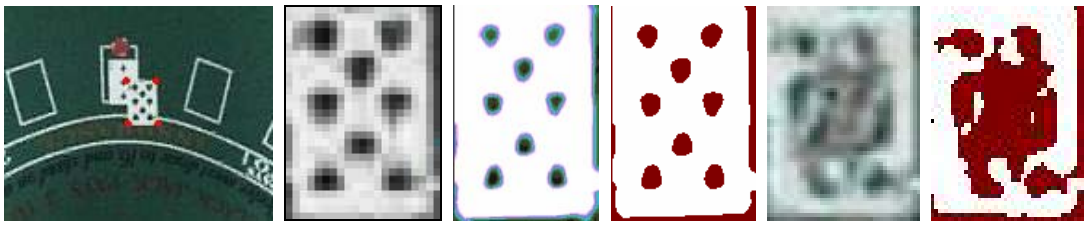


(a) (b)

Figure 5. The outline of three cards showing the central axis line between the furthest points (in pink), the local maxima (in red), and the local minima (in blue) (a), and an example of these points extracted from a sample card outline of two cards (b).

4 Card Recognition

Once the corners of the last card played are known, they are used to generate the co-efficient values needed to normalize the image of the card (so that the long side of the card is aligned to the vertical axis, and the short side is aligned with the horizontal axis (See Figure 6(a) and (b)).



(a) (b) (c) (d) (e) (f)

Figure 6. Cards in play. The four corners of the last card played are highlighted (a), the card is shown (b), an image with multiple thresholds (c), the selected threshold (d), a picture card (e) and finally the thresholded picture card (f).

4.1 Adaptive Thresholding

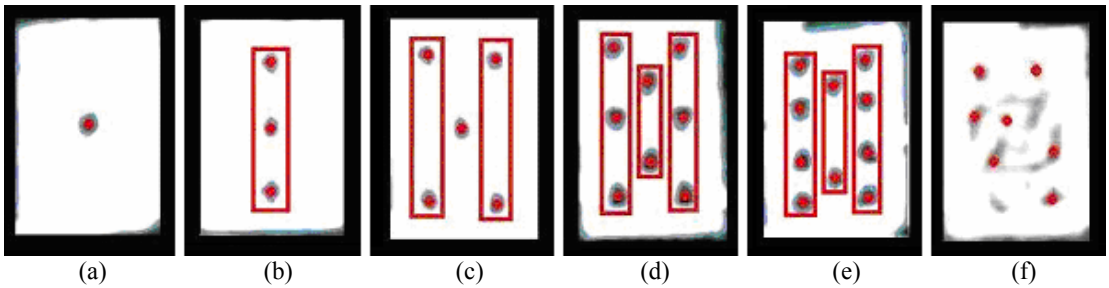
The images of the cards shown in Figure 6 (b) and (e) are typical of the resolution with which the system worked. In blackjack the suit of a card is unimportant and hence the system needs only to be able to distinguish the various number cards (1-10), and the picture cards (Jack, Queen, and King all have value 10).

In order to distinguish the various number cards the *dots* on the card had to be counted. This was done by first applying adaptive thresholding where a number of thresholds are applied to the image (See Figure 6(c)), each of these is processed using connected components analysis to determine the number of possible *dot* regions and the threshold with the largest number of possible *dots* is selected.

There is one exception to this: If any of the thresholded images has a region which is more than a third of the size of the card then the card is immediately classified as a picture card.

4.2 Identifying Number Cards

The orientation of the card is generally not perfect so a little flexibility is needed when attempting value of the card must be derived from the *dots* on its surface. All cards other than aces have multiple *dots* which are aligned vertically (See Figure 7 for examples). The orientation of the line between each possible pair of *dots* is determined and if that orientation is close to vertical, the relationship is noted for further processing. A number of sets of these *dots* may be determined (See Figure 7 (b)-(e)).



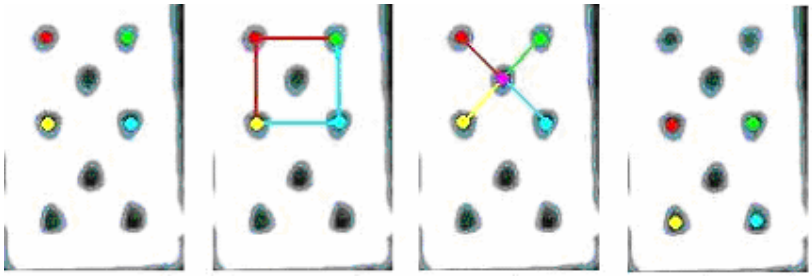
(a) (b) (c) (d) (e) (f)

Figure 7. The dots and vertical sets found for an ace (a), a three (b), a five (c), an eight (d), a ten (e) and a picture card (f).

If no sets are found but there is a single *dot*, which is located in the centre of the card, the card is recognized as an ace (See Figure 7(a)). If one set of *dots* is found the *dots* are evaluated to see if they are positioned in a manner consistent with a card of value two or three. If no set exists

but there are multiple (four or more) *dots* the lack of structure of the *dots* implies the card is a picture card, and has a value of ten (See Figure 7(f)).

In the case of multiple sets of vertical *dots*, the two largest sets are considered first. The first two *dots* from each of these sets (See Figure 8(a)) are evaluated to ensure that they approximately form a square (See Figure 8(b)). If they do then the centre of the square is checked to see if there is another *dot* present (See Figure 8(c)). This process is repeated with each adjacent pair of *dots* from the two largest sets (See Figure 8(d)).



(a) (b) (c) (d)

Figure 8. The four dots considered first (from the two largest vertical sets) (a) form a square (b) in the centre of which is another dot (c). Finally the next set of four dots to be considered are shown (d).

If cards cannot be recognized by one of the above rules then either some of the *dots* must be discarded (e.g. by using a different threshold) or else the card cannot be identified.

5 Results & Conclusions

To date the system has been successfully tested on two pre-recorded video sequences:

1. A sequence containing four hands of play for one player (1 minute, 3 seconds in length, resolution of 360x288pixels and a frame rate of 8 frames per second).
2. A sequence containing three hands of play two one player (1 minute, 35 seconds in length, resolution of 360x288pixels and a frame rate of 8 frames per second).

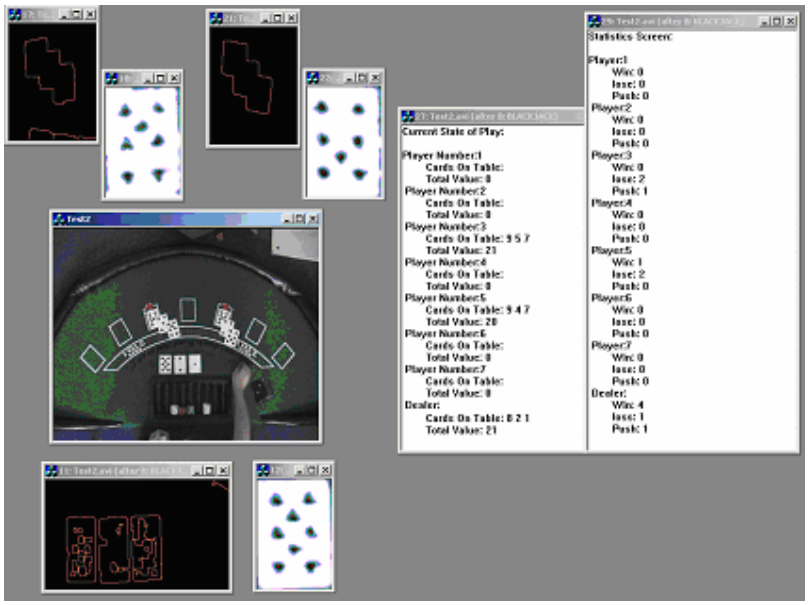


Figure 9. Sample screen shot from the system at the end of a hand of play. On the left hand side the full view of the table is shown along with outlines of card regions are shown for the two players and the dealer/croupier along with an image of the last card processed for each player. On the right hand side the cards registered for each player and the dealer are listed and statistics for each player and the dealer are given.

The system (when run under Windows on a 2.4GHz processor) executed in real-time on the sequences tested. This is despite a substantial processing overhead due to its implementation within a prototyping environment which was not developed as a time-critical application (for example every stage in the processing results in a rendering overhead).

To develop the system further there are many other aspects of blackjack which will need to be supported such as "doubling down", "splitting pairs", "insurance", and the monitoring of chips and their values. However, based on our results we believe that real time monitoring of blackjack from a single low-resolution overhead camera is feasible.

References

[1] Mindplay (2004). MP21. Mindplay LLC, 11225 SE 6th St., Building C, Suite 110, Bellevue, WA 98004, USA.

[2] Sonka, M., Hlavac, V. & Boyle, R. (1999) *Image Processing, Analysis, and Machine Vision*, PWS publishing.

A COMPLETE VISION SYSTEM FOR DEBRIS FLOW MODELLING

Alberto Biancardi*
Dip. di Informatica e Sistemistica
Massimiliano Barbolini Paolo Ghilardi
Dipartimento di Ingegneria Idraulica
University of Pavia
Italy
email: alberto.biancardi@unipv.it

Abstract

Debris flows can be highly destructive: they can denude vegetation, clog drainage ways, damage structures, and endanger humans. In some real world cases debris flow can be triggered by phenomena that are very close to a dam break and, up to now, it was not known how far clear-water models could be applied to debris and granular flows. An experimental set-up has been created to validate and tune a mathematical model of dam break flows: the collection of experimental data requires high-speed video acquisition and automatic processing of recorded sequences. This work presents the complete system that has been integrated and developed, using only off-the-shelf parts and opens source software, the processing steps necessary to extract flow profiles from lateral flume views, and a comparison between experimental and simulated data.
Keywords*: High-speed video acquisition, Sequence analysis, Debris flow, Connected filtering*

1 Introduction

The large attention paid by the scientific community to the understanding of debris flows comes from the high destructive power that such flows have exhibited in many dreadful occasions: debris flows can exert great impulsive loads on objects they encounter and are fluid enough to travel long distances or to inundate vast areas, they can exceed $10^9 m^3$ in volume and release more than $10^{16} J$ of potential energy; even commonplace flows of about $10^3 m^3$ can denude vegetation, clog drainage ways, damage structures, and endanger humans [14].

In some real world cases debris flow can be triggered by phenomena that are very close to a dam break. Water flows generated by a dam break have been widely studied and mathematical models for water dam-break waves are available on many textbooks [13]. Compared to water dam-break waves, debris flow waves display a wider variability; moreover, up to now, it was not known how far clear-water models could be applied to debris and granular flows and, from an engineering point of view, if the common practice to predict dam break peak discharge with the classical water formulas can lead to acceptable results when applied to granular flows.

As is the case for other mathematical models, the validation of debris-flow models relies on the ability of measuring crucial quantities in laboratory or in other monitored experiments. A key source for experiment observation and data collection is video acquisition because it is non-intrusive and is able to supply the measurements of many physical quantities from a single experiment execution.

It should be noted, however, that, if inter-frame information is used to derive such quantities, the pixel size and the frame rate are not independent variables, because the displacements inside images depends

*This work was partially granted by Italian Ministry of Instruction, University and Research under contract PRIN 2003

both on the speed of objects and on the zooming factor. When a high spatial resolution is required the maximum speeds that can be handled using standard 25 fps cameras limit severely the range of examinable phenomena, making those cameras almost totally useless.

In the laboratory equipment that has been set up to study dam-break-arisen debris flows, all the experiments are carried out in rectangular flumes placed at different degrees of inclination. Debris flows are triggered by the quick opening of a gate, allowing the material accumulated on one side of the gate to free flow to the other side. Different runs are made using simple water, a mixture of water and fine gravel, and dry granular media.

All the experiments require a high-speed recording with shots taken from a lateral point of view to be able to perform computation on the profile and other flow measurements. Even tough high-speed video-capture systems exist from some time now, they are characterized by really high costs of the hardware and software parts used to set-up the system. Our main aim was to build a system with the following characteristics:

- it is a complete system supplying all the stages from acquisition to data analysis and it is based on open source software;
- it provides researchers with recordings at high frame rates;
- even though the frame rates are remarkably higher than usual analog cameras the system has comparable costs so that a single laboratory may afford several of them.

In the following sections, after an overview of the programming programming environment and of the video acquisition sub-system, the processing of debris flows is presented. A comparison between experimental and simulated data concludes the paper.

2 The processing environment

Pacco is an extension of TCL, a general purpose command language, and its graphical toolkit Tk [5]. Its design is characterized by a *two-language* approach to object-oriented programming where flexible data-structuring and run-time extensibility let the programmer easily code both highly interactive programs and batch processing scripts.

2.1 Data Structuring and Processing

A major innovation of Pacco deals with the scalability of data structures. Instead of hiding whole data structures within the low level side (the C-language functions), the design of Pacco tries to bridge, from the data-structuring point of view, the TCL side and the C side by reckoning the existence of micro-structures and macro-structures and by favouring the use of composition of micro-structures at the TCL level. For instance, an image micro-structure is the two-dimensional array of its pixels, while a macro-structure of images may be a temporal sequence of frames or a multi-band image or an image with its iconic attributes (edges, regions,...). It is worth noting that all the macro-structures are ordered collections of the basic, single banded, image.

Pacco introduces the concept of *container* (or *composite*) object that means an object that stores other public objects, i.e. objects that are accessible from the TCL interpreter. These contained objects are named *components*. Each main container is named *box* and stores a number of components which may be data objects (both simple or composite) or other container components (`Cboxes`), thus allowing the creation of component trees. This macro-structure can be used to fold together heterogeneous data-clusters and to carry additional hints about existing relationships within the data.

Boxes may be private resources of the process, temporary or distributable. Private boxes are also dynamic: they can change, to a certain degree, their structure. In this way it is possible to handle highly

dynamic structures without too much degradation or to embed within a single box or `Cbox` different types of representation of the same data-cluster as soon as they become available (e.g. to keep a logical unity of items derived from a single source).

Actions are C-language or Tcl procedures which access data and produce results, either by modifying the invoking object or by returning a result-string to the Tcl interpreter.

Data-driven applications can be easily coded thanks to the bind action. This kernel service allows the Tcl programmer to add a list of commands to be executed whenever the data of a component change.

2.2 Available Classes And Widgets

The current distribution supports eight (non-kernel) classes: numerical vectors, point arrays, strings, templates (used to store invariant convolution kernels[15] and morphological structuring-elements[11]), and four types of images (single-banded, colour, complex rectangular and complex polar).

A graphic library extends Tk, the tcl-based X11[10] toolkit, with a set of new widgets and canvas items that can be linked directly to Pacco components. A number of utility procedure further enhance the environment with a polymorphic display command, region of interest handling, colormap animation support, and so on. Bindings allow the automatic update of visualized components.

2.3 Cooperative Development

Another service of the kernel is the loading of unknown classes and actions on demand; this means that each user can freely develop new extensions, without modifying the kernel of Pacco itself. In other words, any programmer can define new classes or write new C-language actions without interfering with the main sources. Thus there is no need to keep several copies of the base sources floating around or to trouble with concurrent patches. All and any new feature can be tested independently from other features, and can be installed in the main source tree at any time or kept as a separate library for ever.

3 The video acquisition sub-system

The video sub-system is designed to maximize the price/performance ratio. It is based on a standard GNU/linux compatible PC and it uses only off-the-shelf parts (boards and cameras) and open source software.

The heart of the system is a high-speed progressive camera with digital interface. It is capable of a sustained rate of 36 million pixels per second that may be arranged into different frame-rates (up to 350 fps). This camera is the most expensive piece of the whole sub-system and the one that had to give the best price/performance ratio, it is easily interfaced to a wide series of digital frame grabbers, and it has valuable features as far as image processing is concerned (square pixels, full-frame shutter, and progressive transfers).

The camera is connected to the PC via a digital frame-grabber: a board that handles the transfer of frames from the camera to the computer. Any board may be used as long as it meets the following two requirements: it is capable of operating as bus-master for DMA (direct memory access) transfers and it has an open-source driver, which has to be modified according to the guidelines exposed hereafter.

In order to reduce costs, acquired sequences are not recorded directly to disk; they are stored in RAM at first and only when the grabbing is completed they are transferred to disk. This approach is mandatory because designing a direct-to-disk system capable of storing 36MB/s is possible, but surely it is not cheap at all; on the other hand memory can easily sustain the necessary throughput, is inexpensive and, given current PC specs, can hold more seconds than each sequence lasts on average.

3.1 *One-shot* grabbing

The storage of a full sequence inside the computer memory is achieved by modifying the device driver[7] of the digital frame-grabber, i.e. the module that takes care of handling the frame-grabber on behalf of the operating system. Like every driver, it works at the lowest levels hardware-wise therefore it gets special access to hardware resources and special handling by the operating system scheduler (e.g. it has its interrupt handling routines served in real-time). Modern frame grabbers, in particular, transfer data using direct memory access, DMA for short; therefore the only processing takes place whenever a DMA block transfer is completed. This may be used to grant that the actual processing spent in the interrupt-service routines is minimal, that the data flow between the grabber and the main memory can be accurately controlled and that the time stamps marking the acquisition time of each frame are reliable.

What we did is to make sure that all the frames of a sequence could be saved in memory by building a large enough DMA buffer: during the acquisition of a sequence data is transferred by the board directly into RAM memory while the only processing required is to keep track of the amount of data transferred so that the time-stamp of each frame can be computed exactly.

It is important to notice that device drivers must deal with mechanisms and not policies, as they ignore who is using them (just like objects in OOP) — they must supply models of usage and should not limit how they are used. Hence a new mechanism, a new model of use, had to be defined: beside the single frame acquisition and the continuous frame acquisition (with buffer reuse), we added *one-shot* acquisition. One-shot acquisition works by allocating a really huge buffer, using most of the physical RAM, and by avoiding buffer re-use: it always starts at the first frame within the buffer and it stops automatically as soon as the last frame that fits in the allocated memory has been grabbed. One-shot acquisition gives a continuous sequence of frames starting from a user-defined time and stopping after a pre-selected number of frames.

Once the sequence is grabbed, its frames are available using either memory mapping or plain reads and may be easily saved into files. Owing to its size and its complex structuring, it is not completely painless processing the acquired data. The most straightforward way to bridge the sequence file into a processing environment is by means of memory mapping because the benefits of this solution are many fold: all the sequence frames are available at once, the task of managing the transfers of sequence data between disk and memory is handled efficiently by the operating system, the developed tools become truly independent from the type and length of acquisition.

The PACCO processing environment, being designed to supply flexible data-structuring and easy access to foreign data, makes the sequences accessible thanks to an extension module, loaded on-demand, that maps sequence frames as image objects and whole sequences as custom-build boxes, leaving the programmer free from the acquisition details as said before.

4 Automatic computation of flow profiles

A number of experiments was performed and recorded. In the following discussion special attention will be given to two sets of them as depicted in figures 1and 2. One set of runs used a mixture of water and fine gravel: the grain size is almost constant and its mean diameter value is $5mm$; the grain density is $\rho_s = 2610kg/m^3$, concentration at rest is $c^* = 0.59$, while the static friction angle was experimentally estimated as $\varphi = 33^o$. The other set of experiments was made using dry PET cylinders: the grain size is $2.5mm$, the grain density is $\rho_s = 1285kg/m^3$, and the thickness of the granular material at rest is 145mm.

The two sets of sequences show a different degree of difficulty as far as the determination of the flow profile is concerned: typically, the simple application of the Canny edge detector [1] to the dry cylinder sequences is able to highlight the sought-after profile. The water-sediment sequences, on the other hand, required a pre-processing step. In fact, even though the sediment solution appears markedly darker than the background a simple threshold operation is not able to segment correctly the foreground and the

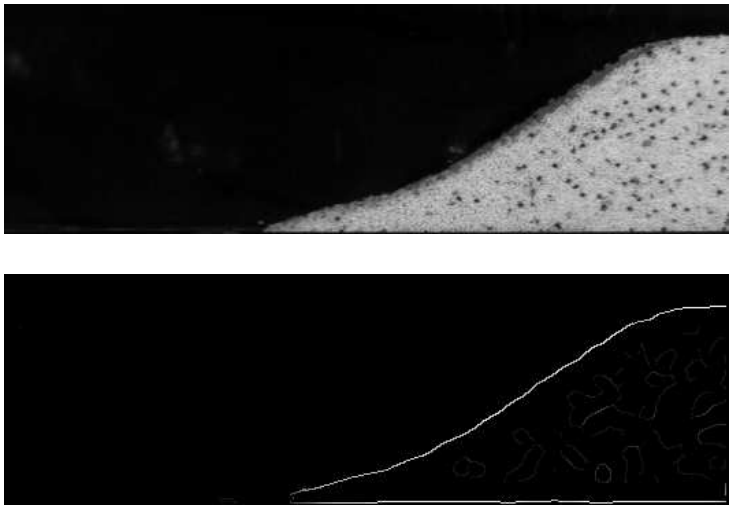


Figure 1: A frame of a dry flow experiment and its resulting edges

direct application of edge detectors does not result in a single, continuous and reliable profile.

4.1 Filters by reconstruction

Like any other filtering stage, the goal of our pre-processing is to smooth and eliminate spurious detail that exists in input images. In this case, *spurious* means anything that prevents the correct evaluation of the flow profile. While adaptive smoothing [8] provides an attractive multi-resolution scheme to filter input signals by implementing an anisotropic diffusion process [6], it is also true that the convergence of adaptive smoothing is only asymptotic and, when applied to simplify two-dimensional signals such as images, results are less satisfactory because of the unequal strength of image edges (which, indeed, is the motivation that lead us to the filtering stage).

Mathematical morphology, on the other hand, provides a family of filters that preserve the significant edges of an input image, similarly in some sense to adaptive smoothing. Those morphological filters, called filters by reconstruction [9], have the property of simplifying image contents while preserving contours. Filters by reconstructions collect openings by reconstruction and closings by reconstruction and work on connected components

In particular we define *opening by reconstruction* any operation that is the composition of any pixel-removing operation composed with a connected opening (which actually reconstructs any connected component that has not been completely removed); on the other hand *closing by reconstruction* is the dual operation in that it is the composition of a pixel-adding operation composed with a connected closing (which completely removes any component which is not entirely preserved). Connected openings and connected closings are also known under the names of geodesic dilations and geodesic erosions [3] or propagations [2] depending on the different points of view they were first introduced. Filters by reconstruction for grey-scale images are computed by stacking (i.e. adding) the result of their binary counterparts applied to each of the (grey-scale) image cross sections [12].

Aiming at preserving the integrity of the profile geometry, especially in the first frames of the sequences when we have the evolution of the debris-flow front, area filters were chosen among the filters by reconstruction because of their shape-preserving ability; at the same time these filters reduce variation among pixel values, which again plays favourably in limiting the bad effects textures may cause. Area filters belong to the class of filters by reconstruction; in particular area openings and area closings use a size criterion for the pixel-removing or pixel-adding operations: any component whose size is less then the required amount is removed.

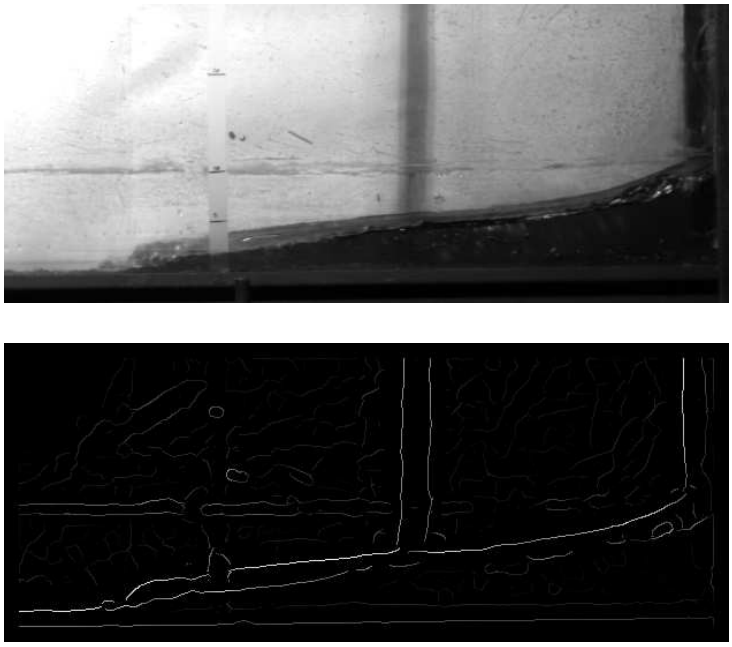


Figure 2: A frame of a mixture flow experiment and its resulting edges

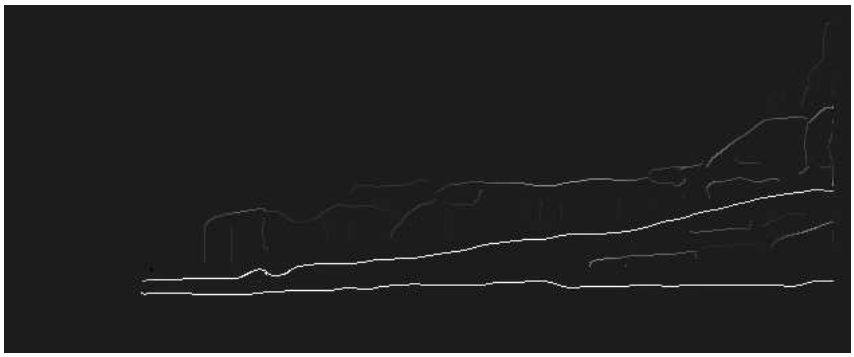


Figure 3: The resulting edges after the pre-processing steps

4.2 The full pre-processing pipeline

The very first steps in the proposed per-processing are the computation of brightness normalization and difference from a reference frame so that the static content of sequence frames is minimized and the effect of uneven illumination and of artefacts on the flume walls is made negligible.

The second pre-processing step is the application of area filtering. A sequence of increasing sizes are used to regularize the main image regions. The result is a quite homogeneous image that has not modified the border between foreground and background.

The final pre-processing step takes into account the gradient of decreasing luminosity when moving from right to left; this gradient can be removed by computing a normalization factor vertical line by vertical line (i.e. based on the maximum value within each vertical line of the image). The normalization factor is, however, limited to a factor of 1.5 in order to prevent a counterproductive amplification of noise in the darkest part of the image. The main benefit of this normalization is possibility of extracting the profile border without imposing any a-priori regularity and leaving the opportunity to choose the regularization function to the post-processing of border coordinates.

Figure 3 shows the final edges after all the pre-processing steps. The differences in magnitude among the main profile and the other edges make the profile selection really straightforward.

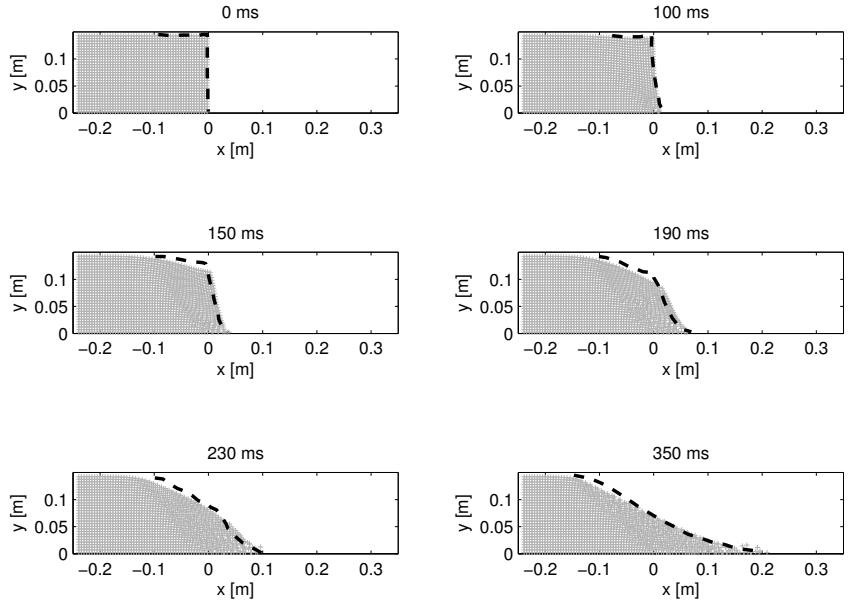


Figure 4: Comparison between a simulated flow and the flow profile extracted from the experiment recordings

5 Comparison between experimental data and numerical simulation

The numerical model was developed on the assumption that the rheology can be reproduced according to the Mohr-Coulomb yield criterion and is based on Smoothed Particles Hydrodynamics method [4] Preliminary results show a good agreement. Figure 4 displays the comparison between experimental and computed flow.

Flow discharges were evaluated by means of numerical integration of the extracted experimental profiles; they were computed using the mathematical model as well. The comparison between experimental and simulated ones is shown in figure 5. Even though there is slight shift in time, both observed and computed flow discharges exhibit the same peak at about $q_b = 3100ml/s$, which is markedly less than the theoretical value ($q_{Th} = 5120ml/s$) coming from the classical dam break formulas developed for clear water [13].

6 Conclusions

A complete system for the handling of debris flows experiments was presented. The system provides high-speed video acquisition and sequence processing; it is low cost and open-source based. The system is being used to model debris flows arising from dam breaks and the preliminary results show good agreement between experimental and simulated data.

Acknowledgments

The authors wish to tank Matteo Pagliardi for his work in testing and validating the system

References

[1] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6), November 1986.

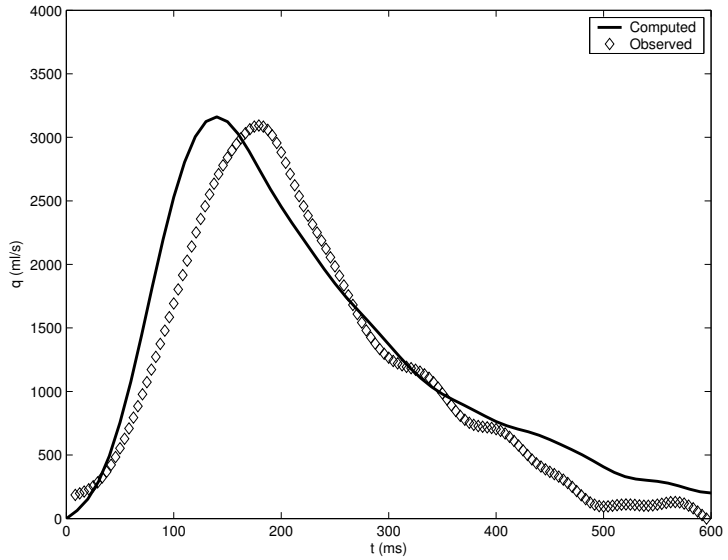


Figure 5: Comparison between the experimental flow discharge and the simulated one

[2] M. J. B. Duff. Propagation in cellular logic arrays. In *Proc. Workshop on Picture Data Description and Management*, pages 259–262, 1980.

[3] C. Lantuéjoul. Geodesic segmentation. In K. Preston Jr. and L. Uhr, editors, *Multicomputers and Image Processing*. Academic Press, New York, 1982.

[4] J. J. Monaghan. Smoothed particle hydrodynamics. *Annual Review of Astronomy and Astrophysics*, 30:543–574, 1992.

[5] J. K. Ousterhout. *Tcl and the Tk Toolkit*. Addison-Wesley, Reading, MA, 1994.

[6] P. Perona and J. Malik. Scale space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12:629–639, July 1989.

[7] A. Rubini. *Linux Device Drivers*. O'Reilly, Sebastopol, CA, 1998.

[8] P. Saint-Marc, J. Chen, and G. Medioni. Adaptive smoothing: a general tool for early vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13:514–529, June 1991.

[9] P. Salembier and J. Serra. Flat zones filtering, connected operators, and filters by reconstruction. *IEEE Transactions on Image Processing*, 4:1153–1160, 1995.

[10] R. Schifler and J. Gettys. *X Window System*. Digital Press, Reading, MA, 1990.

[11] J. Serra. *Image Analysis and Mathematical Morphology*. Academic Press, New York, 1982.

[12] P. Soille. *Morphological Image Analysis*. Springer-Verlag, Berlin, 1999.

[13] J. J. Stoker. *Water Waves*. Interscience Publishers, New York, 1957.

[14] T. Takahashi. *Debris Flow*. Balkema, Rotterdam, 1991.

[15] J. N. Wilson and G. X. Ritter. *Handbook of Computer Vision Algorithms in Image Algebra*. CRC Press, Boca Raton, FL, 2000.

Author Index