Real–time Facial Expression Analysis

A thesis presented in partial fulfillment of the requirements for the degree of

Doctor of Philosophy (Ph.D.)

in

Computer Science

at Massey University, Auckland,

New Zealand.

Chao Fan

2008

Abstract

As computers have become more and more advanced, with even the most basic computer capable of tasks almost unimaginable only a decade ago, researchers and developers are focusing on improving the way that computers interact with people in their everyday lives. A core goal, therefore, is to develop a computer system which can understand and react appropriately to natural human behavior.

A key requirement for such a system is the ability to automatically, and in real time, recognises human facial expressions. In addition, this must be successfully achieved regardless of the inherent differences in human faces or variations in lighting and other external conditions.

The focus of this research was to develop such a system by evaluating and then utilizing the most appropriate of the many image processing techniques currently available, and, where appropriate, developing new methodologies and algorithms.

The first key step in the system is to recognise a human face with acceptable levels of misses and false positives. This research analysed and evaluated a number of different face detection techniques, before developing a novel algorithm which combined phase congruency and template matching techniques. This novel algorithm provides key advantages over existing techniques because it can detect faces rotated to any angle, and it works in real time. Existing techniques could only recognise faces which were rotated less than 10 degrees (in either direction) and most could not work in real time due to excessive computational power requirements.

The next step for the system is to enhance and extract the facial features. To successfully achieve the stated goal, the enhancement and extraction of the facial features must reduce the number of facial dimensions to ensure the system can operate in real time, as well as providing sufficient clear and detailed features to allow the facial expressions to be accurately recognised. This part of the system was successfully completed by developing a novel algorithm based on the existing Contrast Limited Adaptive Histogram Equalization technique which quickly and accurately represents facial features, and developing another novel algorithm which reduces the number of feature dimensions by combining radon transformation and fast Fourier transformation techniques, ensuring real time operation is possible.

The final step for the system is to use the information provided by the first two steps to accurately recognise facial expressions. This is achieved using an SVM trained using a database including both real and computer generated facial images with various facial expressions.

The system developed during this research can be utilised in a number of ways, and, most significantly, has the potential to revolutionise future interactions between

humans and computers by assisting these reactions to become natural and intuitive. In addition, individual components of the system also have significant potential, with, for example, the algorithms which allow the recognition of an object regardless of its rotation under consideration as part of a project aiming to achieve non-invasive detection of early stage cancer cells.

Acknowledgement

In 1999, I left my beautiful motherland China to move to a strange country called New Zealand to improve my English and further my studies. I was an international student fully supported by my parents (Prof. Dr. ZhanGuo Fan, QiuLing Gong). It was impossible for me to continue my study without their help and it has been their continuing support and encouragement which has made my study in New Zealand possible, special thank you Mum and Dad.

In 2001, I met my main supervisor Dr. Abdolhossein Sarrafzadeh, who changed my life. He helps me thoughtfully and he gave valuable advice for my study and in 2003, he encouraged me to undertake my PHD study. His good suggestions and comments always kept my study and my research on the right track. I learned not only knowledge, but also spirit from him. This spirit is never give up no matter what the problem and this spirit will stay with me forever.

I would like to thank for Dr Hamid Gholamhosseini and Dr Martin Johnson, my co-supervisors for their kind help and advice with my research, and the way they generously shared their knowledge during the project.

I would like to thank for Zhuo Li, my wife, for her continuous support and help at all times.

I would like to thank for Ben Powles, who kindly and carefully read and corrected my thesis. Without his help this thesis would not read so smoothly.

I would like to thank Sam Alexander, who always helps me correct my English, as well as Farhad Dadgostar and Andre Barczak, my classmates and my friends. We helped each other, and we had wonderful time in IIMS, Massey University.

Finally, thank you for all New Zealanders who helped and supported me, your kind help made me consider New Zealand as another home.

Table of contents

List of Figures

List of Tables

Chapter 1. Introduction

Dave Bowman: Hello, HAL do you read me, HAL?
HAL: Affirmative, Dave, I read you.
Dave Bowman: Open the pod bay doors, HAL.
HAL: I'm sorry Dave, I'm afraid I can't do that.
Dave Bowman: What's the problem?
HAL: I think you know what the problem is just as well as I do.
Dave Bowman: What are you talking about, HAL?
HAL: This mission is too important for me to allow you to jeopardize it.
Dave Bowman: I don't know what you're talking about, HAL?
HAL: I know you and Frank were planning to disconnect me, and I'm afraid that's something I cannot allow to happen.
Dave Bowman: Where the hell'd you get that idea, HAL?
HAL: Dave, although you took thorough precautions in the pod against my hearing you, I could see your lips move.
2001: A Space Odyssey

2001: A Space Odyssey was produced in 1968 and shows the vision of what human-computer interactions were expected to be in the foreseeable future. While we have computers in our homes capable of millions of calculations per second, as well as computers controlling many other aspects of our lives, 50 years on from this movie, we are still nowhere near having this level of interaction. With so many advances in technology and science, why are we still forced to interact with computers using precise, discrete commands rather than natural human actions?

The aim of this research is to focus on a key element for improving human-computer interactions and to answer the question of whether computers can recognise and understand human emotions, and, based on that information, take appropriate actions.

1.1. Background

In last two decades, computers have become commonplace in our everyday lives, to the extent that they are now an integral and expected part of our society. One of the key reasons which have allowed computers to become an increased part of our lives is the advances in hardware technology which has meant that the cost of computers has decreased significantly while at the same time their capabilities have increased exponentially. This has meant that almost every part of our lives is touched by computers, from our cars to our TVs to our phones to our fridges and of course the home computer, with at least one (and sometimes more) present in almost every household in our society.

However, while these advances in hardware technology have been significant and dramatic, the way that humans and computers interact has not changed nearly as significantly. The very first computers required people to send commands to the computers using paper marked with holes. There is no doubt that this process has improved somewhat, in that we can now communicate with computers by keyboard and mouse – something accessible and easily understandable by everybody. However, the basic principle of computer-human interaction has not changed at all – computers still require precise and discrete commands as input – any commands that are outside those deemed acceptable simply result in error messages or system crashes – even when those commands seem perfectly normal and intuitive to the human users. And computers certainly cannot deal with large amounts of random or changing data from

which they are required to choose the most relevant information on which to base their actions.

The question of whether computers can learn, watch and think like human beings has been around almost as long as computers themselves. In 1950, in a paper titled “Can machine think?” Alan Turing (1950) gave the concept of the “Turing test”. The Turing Test is a proposal for a test of a machine's capability to perform human-like conversation. Turing came up with an elegant solution. He constructed the simple proposition that if human beings are intelligent, and if a machine can imitate a human, then the machine, too, would have to be considered intelligent. The Turing test required that a human “talk” to a computer and another human, and not know which one was the computer.

More recently, in 1997, chess world champion Gary Kasparov was defeated by Deep Blue, a super computer from IBM. However, this did not prove that Deep Blue was ‘intelligent’ in the sense that Turing described – only that it was able to compete with Kasporov in a specific task due solely to its huge computational power. A computer passing the Turing test would be a significant step forward from Deep Blue. However, despite recent developments in artificial neural networks, data mining, and support vector machines, which prove the concept of computer’s ability to learn, we still appear to be some distance away from a computer passing the Turing test.

What then is the next stage for human-computer interaction? Many researchers believe that the next significant advances are unlikely to come through advances in hardware, but through the way that computers can be made to understand and better react to natural human actions. This step may be the creation of real and effective artificial intelligence, where computers do not simply rely on input from users to make decisions, but can use visual and other stimuli to take appropriate actions themselves. In this way, our interactions with computers should develop so that, as closely as possible, they resemble our interactions with other people. This means that we should be able to communicate with computers by talking, using facial expressions and gestures with the computer understanding what the user means, and what the user wants the computer to do. In effect the computer must be intelligent.

However, first researchers must determine what exactly is meant by ‘intelligence’. The human brain obviously has the ability to generate and store intelligence, and there are now many accepted methods for measuring this intelligence, with such measurement often undertaken in real life. For example, people often receive a kind of aptitude test when they apply for a job and those people with high marks will have more opportunity than those people with low marks. The IQ (Intelligence Quotient) test is often used in schools and indicates a person's mental abilities relative to others of approximately the same age. However, the human brain has hundreds of specific mental abilities and only some can be measured accurately and may be reliable predictors of academic and financial success.

Some functions of the brain are well understood, and have been replicated in basic forms of artificial intelligence. However many of the exact workings of the brain have not yet been discovered by modern science, including how brain signals are delivered, how these signals are processed, and how images are automatically selected, sorted and stored. The question now is whether we can build a machine with artificial intelligence which can complete some of the human brain's more complicated functions – particularly those around detection, recognition and understanding of visual signals. Once we can successfully recognise those images, the next step is to build a system which can react appropriately and naturally to these signals. How though can this be achieved?

1.2. Summary of research goals

The goal of this research is to develop a system which can automatically detect and recognise human facial emotions. The system must work in real time, and on a computer similar to those found in most homes in today's society. In addition, the system must be able to provide accurate results despite the large variations in human faces due to gender, age and ethnicity, and also variations in lighting conditions.

This goal has been chosen because facial emotions are a key part of the way humans communicate with each other, both in verbal and non-verbal ways. In fact the display

(even inadvertently) of human emotions can often show true feelings, regardless of the words being said. This means that recognition and understanding of emotions must be an integral part of any system which allows humans and computers to interact with each other in a natural and instinctive way.

In order to achieve this goal, the following subtasks will need to be completed:

- Detect faces rotated by any number of degrees.
- Enhance and extract facial features regardless of differences in face shape due to gender, age and ethnicity and under a variety of different lighting conditions.
- Reduce the number of feature dimensions used in the facial expression classifiers so that accuracy is increased and computational time is decreased.

1.3. Statement of the problem

Automatic and real time facial expression analysis is a complicated process and it involves many related and sophisticated tasks, including object detection (specifically face and eye detection), feature extraction, and feature representation.

The wider field of machine vision and machine learning has been, and continues to be, the focus of a significant amount of research as it has many and varied current and

potential applications. For example in outer space or underwater exploration a robust and accurate machine vision system will allow machines to find and recognize objects within these dangerous environments where human exploration would be very dangerous and difficult. In addition, machine vision is also important in modern military defense with many countries are trying to develop fast and reliable defense system based on visual signals. Finally, machine vision systems have a variety of applications in our daily life, both now and in the future such as automatic pilot systems in vehicles which could, one day, allow vehicles to drive themselves. Machine vision will also offer significant advances in medical research. For example, cancer, which is very difficult to treat with drugs and radiation therapy alone, could be tackled with automatic cancer cell recognition systems which could save lives by detecting these cancer cells in their early stages. However, despite some significant advances in some areas, in general, today's machine vision systems are still very immature.

A significant amount of resource has also been spent on developing a system which will allow machines to learn and understand in a similar way to the human brain. Recently, researchers have focused on developing and simulating a neuron like structure which can complete particular, but limited tasks. These are known as artificial neural networks. These artificial neural networks have shown some ability to learn and make decisions and in some areas artificial neural network already play important roles. However, these abilities are still not strong enough, and significant progress still needs to be made in the field of artificial intelligence.

The first step for this project is to review and compare the significant amount of research already conducted and currently underway in the fields of machine vision and machine learning to determine which of these techniques may be suitable for use in this research. The next step is to then develop techniques which can fill the gaps left by existing methods, in order to achieve the stated goals.

1.4. Motivation and scope of this research

It is acknowledged that the stated goals of this research are broad, with large and varied (if not limitless) possible applications. For this reason, it was necessary to narrow the scope of the research so that it applies to a specific application, and it was decided to focus on an application which would provide immediate, useful, benefits. With this in mind, it was decided to create a human-like tutoring system, called the Next Generation Intelligent Tutoring System (NGITS) (Sarrafzadeh et al. 2003; Sarrafzadeh et al. 2004). The goal of the NGITS is to use a friendly, intuitive and natural interface to improve communications and interactions between humans and computers – in effect to act like a real human tutor would act. For example, if the student was frustrated or angry, the NGITS would provide encouragement, and make the questions easier, while if the student was happy as they were answering the questions correctly, the NGITS could give congratulations and make the questions harder.

One of the key requirements necessary for the NGITS to achieve its goals is the ability to correctly detect, recognise and understand human facial expression. This facial expression recognition is the subject of this thesis.

The NGITS also required several other capabilities including the recognition of human gestures, the collection of these gestures and the facial expression information, and then the ability to make assumptions about the correct action based on this information. These requirements are being undertaken by Farhad Dagostar (Dadgostar, Fan, Sarrafzadeh 2005; Dadgostar et al. 2006) and Samuel Alexander (Alexander, Sarrafzadeh, Fan 2003; Alexander, Sarrafzadeh 2004; Alexander, Hill, Sarrafzadeh 2005) respectively. If these requirements could all be successfully implemented, the NGITS system will be able to 'think' and give reasonable responses as expected from a human, based on natural interactions with users. The architecture of a complete NGITS is shown below in Figure 1.1, which shows the extent of the current research and how it will be extended by the NGITS team.

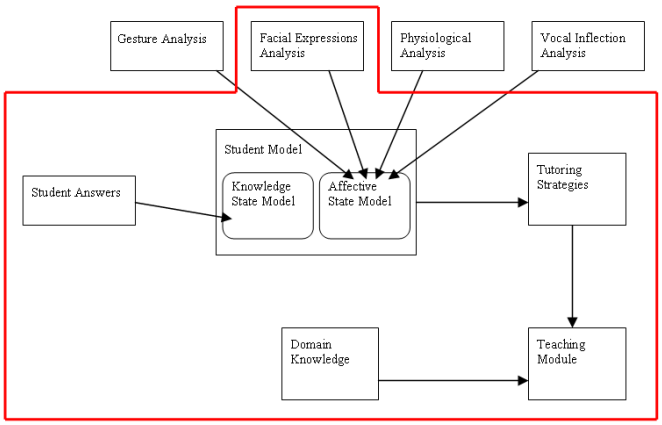


Figure 4.2.4.1.1 Structure of NGITS

1.5. Contribution of this research

This research has contributed to the knowledge in the area of computer vision and computer learning by both conducting a thorough review, analysis and comparison of existing methodologies and by developing novel techniques to allow the stated goals to be achieved.

Overall this research has developed a system which can accurately recognise human facial expressions regardless of differences in age, gender, ethnicity and lighting conditions in real time and using a standard household computer. In addition, the system has been taken out of the laboratory and implemented into a tutoring system which is showing immediate benefits during use in the real world.

Specific contributions are described in more detail below.

- A novel technique has been developed which allows faces to be detected regardless of the degree of rotation of the image. This is believed to be the first time this capability has been implemented, and can be applied and used for objects other than faces. The technique also allows faces to be recognised regardless of their size.

- The task of quick and accurate face detection has been significantly improved with a novel method that combines the phase congruency algorithm and

similarity measurements using FFT. The technique developed provides accurate results regardless of differences in face shape due to age, ethnicity and gender, and regardless of changes in lighting.

- The extraction of facial features has been improved by combining a novel algorithm using face images mean values with the existing CLAHE technique. This improvement has meant that key facial features are emphasized, but not distorted, meaning that the accuracy of expression detection is improved.

- A key bottleneck in existing recognition methods has also been significantly reduced. This bottleneck is the large number of feature dimensions which are created during the process and require an excessive amount of computational power and therefore time to process. Using a combination of radon transformations and discrete cosine transformations this research has introduced a new and efficient method for reducing the number of feature dimensions to a degree that the required calculations can now be undertaken in real time, with no significant loss in accuracy.

1.6. Thesis Overview

This thesis will introduce a new face detection algorithm and new algorithm for object classification which is applied for facial expression analysis. The outline of the thesis is described below.

1.6.1. Thesis outline

- Chapter 1: Introduction to machine intelligence and current human-computer interaction and summary of thesis.
- Chapter 2: Literature review of current research in relation to face detection, feature extraction, feature representation, and facial expression analysis.
- Chapter 3: Overview of the novel facial expression analysis system.
- Chapter 4: Description of facial detection algorithm which will use the novel algorithm for rotated unrelated feature extraction.
- Chapter 5: Describes feature enhancement algorithm, including a comparison of all available algorithms for image enhancement.
- Chapter 6: Description of novel algorithm for facial expression analysis.
- Chapter 7: Conclusion and suggested future work

Chapter 2. Literature review

A real time facial expression analysis system involves technology relating to image processing, feature extraction, and machine learning. This research combines and integrates all of these techniques. Over the past decade a large amount of research has been conducted in the field of object recognition, and many significant breakthroughs have been made, particularly involving the detection of inanimate, objects with known, fixed parameters. However, detection of the human face remains a difficult task to consistently and successfully achieve. This is primarily due to the large number of features in the human face and the wide range of variety each of those features can have – based on sex, age, and ethnicity. In addition, any face detection algorithm must be able to succeed in a wide variety of lighting conditions. Furthermore, the detection and successful recognition of facial expressions must surmount all of these difficulties, as well as being able to recognise a variety of expressions, which may vary significantly on different faces when made by different people. It is a very challenging task to achieve.

This research consists of three main areas - image processing, feature extraction, and the learning ability of machines. The optimum image processing algorithm combined with object classifier trained by Support Vector Machine or Artificial Neural Network will allow the system to accurately extract correct features. Feature extraction is an

important link between machine learning and machine vision, and a high performing feature extraction algorithm will improve accuracy and therefore system performance.

A variety of different algorithms have been developed in each of these areas, and a number of researchers have attempted to combine these techniques. This previous work has been reviewed and analysed and is summarized below.

2.1.Face detection

2.1.1. Face detection based on Skin Color

Face detection by skin color is the simplest and fastest method to detect the human face. The skin color is considered a feature and the system attempts to find regions containing skin by searching for pixels within the skin color range. It has the advantage that it is able to detect faces with a wide range of rotation and poses, but it has a significant disadvantage in that it is very inaccurate and results in a high number of false detections. Its accuracy also suffers significantly when there is a wide variety of skin colors and lighting conditions, and this technique does not work at all on grey level images.

Kovac, Peer and Solina (2003) attempted to reduce some of these problems relating to variations in lighting through the introduction of their algorithm Illumination Independent Color-Based Face Detection. Singh, Vatsa and Singh (2003) introduced a

method which combines RGB, YCbCr and HSI color space for face detection which reduces some of the issues in relation to the variety in skin colors. Other researchers have also realised the importance of accurate recognition of color for this technique. In Sobottka and Pittas's (1996) paper, they segment face image in Hue-Saturation-Value (HSV) color space, and apply Active Contour Modeling for face outline detection. However, despite these advances and improvements in accuracy, the applications of color based face detection techniques are limited. They can be used as a first step in face detection, but they are not suitable for high level features extraction and analysis.

2.1.2. Template matching

Template matching is another method of face detection. Under this technique an average face template image is obtained by averaging faces from a database. It is important to have a wide range of different face types – both genders and a variety of ages and races. This averaging is completed offline.

The system then scans the sample image using search windows (which can be automatically varied in size) and attempts to match sample images with the template image, by finding the distance value between the search windows and the template. A low value indicates a likely match and therefore a probable face candidate. One of the earliest template matching algorithms for face detection was developed in 2000 by Chang and Robles (2000). In 2003, Ilhan and Meiyappan (2003) combined skin color

and template matching algorithm in an attempt to improve both systems.

The advantage of the template matching technique is that it allows the position of the face within the image to be located. This means that if further processing is required, the non-face regions of the image can be excluded. However, there are also a number of disadvantages. A large amount of computation power is required, and the algorithms accuracy decreases significantly when images are rotated or distorted – through the faces being in different perspectives or variations in lighting. A variety of methods have been created which attempt to improve template matching, but, in general, because of computational complexity and the inherent likelihood of facial features becoming distorted in real life situations due to a number of factors including variation in face positioning and lighting, they do not improve results significantly. This means that face detection using template matching alone is still not robust enough for use in real life situations.

However, this research uses the template matching technique as part of the face detection system. This is described in more detail in Chapter 4.

2.1.3. Frequency domain

It is possible to separate objects based on different viewpoints using FFT and Wavelet Transformation which allow the analysis of data across frequency domains. Frequency

domain analysis has been used for face detection, and more broadly in signal analysis and image processing for a number of years. FFT is widely used to analyse signals generated by medical devices (such as electrocardiograms and ultrasounds) (Starr 2005), as it is relatively simple to classify one dimension signals by mapping them to frequency domains.

However, mapping images (i.e. two dimensions) to a number of frequency domains using FFT is much more difficult as the accuracy depends on a range of factors, including image noise and quality, differences in skin color and lighting shadows. This problem means that, the number of inaccuracies is high and therefore FFT cannot be used alone for object detection. However, Ben-Yacoub (1997) has created a method which combines Multi-layer Perception (MLP) and FFT to achieve a algorithm capable of fast and accurate object detection.

This research also uses FFT to improve the template matching part of the face detection system. This is described in more detail in Chapter 4.

2.1.4. Geometrical Moment and Zernike Moment

Geometrical Moment

The geometrical moment is a successful technique for detection and recognition of object features. Geometrical moment is the calculation of the properties of connected

pixels or regions and can successfully operate with variations in translation, rotation and scale. This method was originally created by Teague (1979) and Teh and Chin (1988). In 1962, Hu (1962) improved this original work and introduced seven invariant functions for use with the geometrical moment technique and image classification. Hu's invariant moments are widely used in character detection, because characters have well defined and non-changing shapes. However these calculations are very time consuming, and accuracy can be adversely affected by image noise. Because of this problem, this technique cannot be used for real time object detection.

Complex Zernike moments

The Zernike polynomials were first introduced in 1934 by Zernike (1934). In comparison with the geometrical moment technique, Zernike polynomials can more precisely describe object characteristics, and also can operate with variations of translation, rotation and scale. Bhatia and Wolf (1954) further improved the Zernike technique by changing the Zernike transformed images to polar system and then normalised them (Khotanzad, Hong 1990). However, these techniques, while more accurate and less susceptible to noise than the geometrical moment technique, still do not work in real time, as the calculations used are computationally complex and time consuming.

In this research, Geometrical Moment, Hu's moment and Zernike Moment were tested and were found to be very good algorithms for representing object features, but they still cannot be considered as accurate image analysis tools. To improve performance,

beside calculating the whole image's moment, calculating these moments locally will improve performance a lot. Details of this will be discussed in future work section.

2.1.5. Viola and Jones's method

In 2001, Paul Viola and Michael Jones (2001; 2002) created a new method for fast and accurate object detection. Their method uses the integral image process to guide the image search and then extract the Haar features. Following this step, the AdaBoost learning algorithm (also known as OpenCV) is used to classify whether or not the image contains a face. Viola and Jones's algorithm can run in real time and can accurately and efficiently eliminate images without facial features before further processing. However, a significant problem with this method is that the algorithm will not recognise faces which are rotated more than 10 degrees (in either direction). This is a significant barrier to being able to deploy an accurate face recognition system for use in real life situations, where people's faces are often not held straight up.

2.1.6. ANN for object detection

ANN is the best approximation for computer systems to simulate the functionality of human brain. A basic form of ANN was first created in 1940 by McCulloch and Pitts (1943) who introduced the first neural network computing model. After ten years, a two-layer network was introduced by Rosenblatt (1962) which introduced the concept of perception. However while ANN's were able to complete certain tasks, they could

not solve more complicated problems, and researchers were unable to make any significant breakthroughs for several decades. The next improvement came in 1986, when a multilayer neural network with back-propagation learning algorithm was introduced by Rumelhart and McClelland (1986). This type of ANN is still the most popular in use today.

ANN has been used successfully in the field of image processing and in particular for character recognition for many years. Using ANN to detect faces was first attempted by Rowley, Baluja and Kanade (1998). Their system is highly accurate when faces are not rotated beyond 10 degrees and is not affected by different size images as they reduce the image size in steps and search each reduced image until a set minimum image size is met. However, this scale reduction technique means that their system cannot be used in real time, and again, it is unable to locate faces rotated beyond 10 degrees. In addition, while the training for the system is done offline, it is difficult and can take several days to complete each time.

2.1.7. Statistical K-mean

Statistical k-means clustering (Jain, Dubes 1988) is an algorithm to classify or to group objects based on features into K group. It has many different applications and has been used successfully in image processing, with some of the original work in this area completed by Byrd and Balaji (2006), and Su and Chou (2001) for face detection and

also skin color classification. In this algorithm, K is the positive integer number for a given group. The grouping is done by minimizing the sum of squares of distances between data to the corresponding cluster centroid. K-means clustering are constructed with voting, weighted voting, selective voting, and selective weighted voting.

In comparison with other methods, Statistical k-means clustering has better performance in detection, and better results at locating skin color. However it has limited pattern analysis ability which is necessary for feature detection. In addition, the speed of this algorithm is affected by a number of factors including the image size and number of groups, so is unsuitable to operate in real time.

2.1.8. Support Vector Machines

Aware of the limitations of ANN, researchers focused on developing another method for machine learning, known as a Support Vector Machines (SVM). SVM were originally introduced by Vladimir Vapnik (1974; 1979; 1995; 1999; 2006). SVMs map data (input vectors) to a higher dimensional space where a maximal separating hyper-plane is constructed. Two parallel hyper-planes are constructed on each side of the hyper-plane that separates the data. An assumption is made that the larger the distance between these parallel hyper-planes the better the generalization error of the classifier will be. Schölkopf and others (Schölkopf 1997; Schölkopf, Smola, Williamson and Bartlett

2000; Schölkopf, Platt, Shawe-Taylor, Smola and Williamson 2001), Cristianini (2000) and Shawe-Taylor and Cristianini (2004) introduced kernel methods to classify more complex input data.

In comparison with ANN, SVMs have improved learning abilities and have better general performance. In addition, the training required for SVMs is much simpler and significantly less time consuming. The training is completed offline, which means that SVMs can be used for real time applications. In this research, SVM is trained as a classifier for face detection, eye detection and facial expression analysis, and is described in more detail in the following chapters.

2.2. Image processing for facial feature extraction

Feature extraction is a crucial part for object recognition and objects with clear, fixed features can be classified easily. For this reason, the methods use to process images are important, as they help to improve the clarity of the features within the image This research included the review of many different methods for image processing as there are many algorithms, each of which is suitable for different purposes.

2.2.1. Image noise removal

All images contain noise to some degree, which degrades picture quality and edge and

feature definition. However it is possible to reduce noise to tolerable limits before further analysis is undertaken on the image. There are many noise removal methods available, e.g. image averaging, median filtering, sigma filtering, wiener filter, FFT and Wavelet filtering. As part of this research, these methods have been reviewed and compared.

Average Filter and Median filter

Image averaging is the simplest and fast way of removing noise and is widely used in digital photograph processing. In this algorithm, the value of each output pixel is determined from averaging the pixel value of its neighborhoods. Median filtering uses a similar algorithm, where value of each output pixel is determined from the median pixel value of its neighborhoods. However, both of these techniques results in smoothing of the image, which has the disadvantage that it also blurs the images and degrades edge information, which, as described above, is crucial for feature detection.

Sigma filter

In 1983, Lee (1983) created his Sigma filter which was designed to reduce the problems of the median and averaging filter. The idea of the sigma-filter consists of averaging only those grey values in a window which differ from the grey value of the central pixel by no more than a fixed parameter – known as the “Sigma” value. This filter has the advantage of smoothing the image, without any significant blurring or edge degradation. It also has the added advantage that it is very fast as the computational

requirements are reduced. For these reasons, the Sigma filter was considered the most appropriate for use in this research.

Wiener filter

A classical approach to spatial noise filtering is the noise-adaptive Wiener Filter (Wiener 1949; Brown 1996). Wiener filter uses a pixel-wise adaptive method based on information gathered from a local neighborhood surrounding each pixel. It uses this information to estimate the local mean and variance around each pixel. However, a problem with the method is that it is relatively successful for images with a lot of noise, but does not perform well for clean images. In addition, it also introduces some blurring and is computationally complex meaning that it cannot be used in real time applications.

FFT and wavelet transformation

FFT analysis is a very important technology in signal processing and image processing and operates by separating an image into its various spatial frequency domains. By separating to high frequency and low frequency, the noise is removed easily. An algorithm developed by Kovesi (1993; 1999) provides a good example of image noise removal by using FFT filtering. This algorithm was used in this research for the extraction of facial features. Kovesi also successfully uses phase congruency based on FFT. This is described in more detail in Chapter 4. A similar process is known as wavelet transformation (Arivazhagan et al. 2007).

Edge preserving smoothing process

As discussed above, averaging filtering and median filtering will blur images and degrade edge information and it is difficult to find the optimum sigma value for use in Sigma filtering. Nagao (1979), Horowitz and Pavlidis (1994), and others attempted to create an algorithm which would smooth an image without blurring or edge degradation. Their approach was to output the value of each individual pixel using calculations based on the surrounding pixels. This method was improved, firstly by Nagao (1979) and Tomita (1997) who proposed using a number of rectangular masks containing a set number of pixels as the basis for the calculations. The average grey level of the most homogenous mask will be assigned to each pixel as the output value. This means that the image is smooth, the noise reduced, but the image is not blurred or the edges degraded significantly. This algorithm was used in this research to assist with the generation of plotlines to represent facial features.

Overall, the various algorithms described above have different applications and each has their advantages and disadvantages. For large size images average filtering and median filtering is an efficient technique. FFT and wavelet transformation is a good algorithm to enhance special edge features, while sigma filters are popular in photo processing. A comparison of each technique is summarized below.

Table 2.1 Comparison of different noise removal algorithms

Methods	**Noise removal**	**Blurring of the image**	**Performance**	**Parameter adjustment**	**Universal**
Average Filter	Normal	Yes	Fast	Convenient	Large size image.
Median filter	Normal	Yes	Fast	Convenient	Large size image.
Sigma filter	Good	No	Average	Try	Average
Edge preserving smoothing process	Good	No	Average	Convenient	Average
Wiener filter	Normal	Yes	Average	No	Yes
FFT and Wavelet filtering	Good	Parameter control	Slow	No	Yes

2.2.2. Lighting correction and facial feature enhancement

As mentioned above, differences in skin color and lighting conditions can make facial expression analysis more difficult. Darker skined people look lighter in a bright lighting environment, while lighter skinned people look darker in a poor lighting environment. These problems cannot be overcome by using only techniques which involve RGB or HSV color space analysis. As one objective of this research is to make features more clear to extract regardless of the lighting conditions and skin color, it was decided that grey level images should be used instead of color images. As part of this research, existing feature enhancement algorithms have been reviewed and are summarised in the following paragraphs.

Image gamma adjustment

This technique enhances images by adjusting the gamma value in a simple way- the same technique is available in many consumer image processing software (e.g.

Photoshop). The gamma adjustment changes the relationship between black and white. The default value for gamma is 1, which means linear between black and white. If gamma is less than 1, the mapping is weighted toward higher output values (white) and if gamma is greater than 1, the mapping is weighted toward lower output values (Lee 1983). This means that gamma adjustment is an effective way to adjust brightness. Both Martinkauppi (2002) and Sterring, Anderson and Granum (1999) showed that gamma correction can be applied to adjust an image histogram which will enhance facial features. However, the significant disadvantage of this process is that the gamma adjustment required to optimize the image is different for every image. This makes it unsuitable for this research.

Histogram equalization

Histogram equalization is the term which describes the process by which an algorithm averages the histogram of an image. This algorithm can adjust the histogram in a variety of ways. For example as well as averaging, the image contrast may be enhanced by stretching histogram or it can be adjusted to approximately match a specified histogram.

The histogram gives information about the exposure of an image. If the histogram indicates that there are a large number of dark pixels then the image is probably underexposed. If there are a large number of light pixels then it is probably overexposed. In Rowley, Baluja, and Kanade's (1998) face detection algorithm, a

lighting correction algorithm is applied, and then histogram equalization is applied to enhance these facial features.

Enhancing by Morph logic operation

The morph logic operation is designed to distinguish objects in the foreground from the background, by measuring differences in light exposure and contrast across the image (van den Boomgard, van Balen 1992) (Adams 1993) (Jones, Soille 1996). This algorithm considers the current pixel value and its surrounding area (known as the structuring element). Again, similar to the gamma adjustment technique, optimization of this method is required for each individual image.

2D Plane fitting

The linear lighting correction (Rowley, Baluja and Kanade (1998) is a method for adjusting strength of lighting which allows skin color to be recognised in most lighting conditions. Briefly, this method uses statistical regression analysis to estimate approximate lighting across an image.

Contrast-limited adaptive histogram equalization (CLAHE)

Instead of enhancing features according to the histogram of an entire image Pizer et al. (1987) introduced adaptive histogram equalization in 1987 which enhanced features within an image based on smaller regions. This was further improved in 1994 when

CLAHE was created by Zuiderveld (1994). CLAHE is a similar algorithm with histogram equalization algorithm, but it gives better enhanced features. The basic idea is to divide the original image into many tiles or regions. Instead of calculating the global histogram of an entire image, CLAHE only calculates the histogram for each region. It gives much better contrast and more accurate results. However, it takes a significant amount of testing to find the region size which produces the most optimal results. This process is described in more detail in Chapter 6.

In summary, the basic idea of image contrast enhancement is the estimation of image background and the subtraction of the background image from the foreground (and main) image. Histogram equalization and plane fitting provide global estimations of the background of an image. They are easily implemented and fast, but only suitable for small images (approximately 50 x 50 pixels), as it is relatively simple to estimate and recognise the background is small images. But for images larger than 50 x 50 pixels, histogram equalization and plane fitting are not a useful solution. For such situations, morph logic and CLAHE perform well as they are analyse the image background in small sections, although it can be difficult to find the optimum region size.

2.3. Facial expression analysis

Real time facial expression analysis combines a number of tasks. It involves accurate face detection, facial features extraction, facial features representation, and training a

robust classifier. The most challenging task is detecting facial features and recognising these features. There are a large number of expressions which can be generated by the human face. To simplify the task, this research has been limited to six basic facial expressions – happy, sad, disgust, surprise, anger and natural (i.e. no expression). Previous research in the field of facial feature detection is reviewed below.

2.3.1. Basic facial expressions

A facial expression results from one or more motions or positions of the muscles of the face, and according to Darwin's theory of evolution (Darwin 1872), facial expressions are inherent, rather than learned. These movements convey the emotional state of the individual to observers and are a form of nonverbal communication. They are a significant means of conveying social information among humans, but also occur in most other mammals and some others animal species.

Researchers tried many different methods to recognize and categorise facial expressions (Fasel, Luettin 2002). Ekman and Friesen's (1971) study showed that the six basic expressions have unified properties. Further Ekman and Friesen in 1976 (1976) developed Facial Action Coding System (FACS), to taxonomize every conceivable human facial expression, and in 1998 Essa and Pentland (1998) tried to classify these facial action units. It is the most popular standard currently used to systematically categorize the physical expression of emotions, and it has proven useful both to psychologists and

to animators.

There are a number of existing image processing techniques with which researchers have attempted to use or modify to automatically analyse human facial expressions, including Hidden Marko Models, ANN, and SVMs. For example, Darrel and Pentland (1994), Avent, Ng and Neal (1994), Lisetti and Rumelhart (1998), Yoshitomi, Kim Kawano and Kitazoe (2000), and Lin and Chcn (1999) attempted to recognize the facial expressions using ANN (ANN). Kumar and Poggio (2002) tried using SVMs (SVMs) to classify facial features, and Oliver, Pentland and Berard 1997) , Hu, de Silva, and Sengupta (2002), tried to recognize facial expressions by Hidden Marko Models (HMMs). The results of all of these experiments and testing indicated that, while these existing techniques can accurately detect and recognise a variety of objects, they were not suited to the detection of facial expressions. The failures of these generic image processing systems led researchers to attempt to create techniques specifically for the detection and recognition of facial expressions. For example, Min and Bin (2006) tried analysis of facial expressions based on Graph Spectral Decomposition, and Moses, Reynard and Blake (1995) and Essa and Pentland (1995), developed a system to track facial features using optical flow.

One of the major problems with attempting to recognise facial expressions using Hidden Marko Models, ANN, or SVMs is the large number of feature dimensions that are generated during the image processing, which in turn require a significant amount

of computational power. Therefore, many research groups have been trying in different ways to reduce the number of features dimensions generated by this task.

These facial features extraction algorithms can be broadly classified into the following groups: facial features enhancement, facial features geometry information, facial features color information, hyper-plane mapping method and statistical appearance models.

Yang and Huang introduced facial features extraction by using a method known as mosaic image. In their research, the image is separated via different lattices and the grey value for each lattice are calculated. It then uses known characteristics of the face, eyes and nose to locate these within the image. This method requires a significant amount of computation.

Brunelli (1990) introduced horizontal and vertical projection of facial features, which separates facial features in the projection plan. This was improved by Feng and Yuen (1998) with their Variance Projection Function. This algorithm does not require much computational power, but it fails completely when eye detection fails. In addition, its accuracy drops significantly when the targets are in a complex background or lighting is unbalanced.

Reisfeld, Wolfson and Yeshurun (1995) introduced generalized symmetry transform

(GST) method based on facial features symmetry and shape of facial features. This method can be used for different expressions and lighting conditions but requires huge computational power. This method also does not work when eye detection fails as it uses the eye location for centre symmetry.

A general image processing technique called active contour modeling (also known as snake) was introduced in 1987 by Kass, Witkin and Terzopoulos (1987). This was applied specifically to facial recognition by Liu and Sclaroff (1997) who introduced parameter models to locate the eyes and mouth within images. This method has a limitation similar with active contour modeling, accuracy depends on initial control points.

A further improvement was a technique based on active contour modeling and known as Active Appearance Models (AAM) introduced in 1998 by Cootes and others (Cootes et al. 1995; Cootes, Edwards and Taylor 1998), which focused further on the extraction of facial features. This technique has been used in a number of recent facial expression analysis papers. The AAM algorithm was improved further in 2007 by Torre, Campoy, Cohn and Kanade (2007). Cohn (1996) introduced Temporal Segmentation of Facial Behavior. Wang, Lucey, and Cohn (2007) introduced Non-Rigid Object Alignment with a Mismatch Template Based on Exhaustive Local Search. Datcu and Rothkrantz (2007) introduced facial expression analysis for still images and video in 2007. Wang, et al. (2008) further improved the training of AAM facial feature system by using an SVM

in 2008.

However, despite, all these significant improvements, the AAM algorithm still has its limitations. Although AAM is a good tool to detect and locate people's facial features, it is generally not accurate enough to detect facial expressions. However training an AAM system is difficult and time consuming as it requires various points (such as eyes and mouth) to be manually marked on the training images. Furthermore, researchers still need to improve the step which emphasizes the facial features which will help the SVM to classify the facial expression.

This research introduces a novel facial expression detection algorithm using FFT and a novel facial features enhancement algorithm based on CLAHE. These facial features are further extracted and enhanced using a discrete cosine transform. The final step uses a trained SVM facial expression classifier (Fan et al. 2005a; Fan et al. 2005b; Fan et al. 2005c). The entire system works in real time and the experimental result shows high performance and accuracy. The system is described in more detail in the following chapters.

Chapter 3. Real-time facial expression analysis system overview

Designing an automatic, real time facial expression analysis system is a challenging task. The system involves many advanced technologies in a variety of areas including artificial intelligence, machine learning, machine vision, statistical regression analysis, statistical learning theory and computer graphics.

This work predominately focuses on two key areas - improving machine learning abilities and improving facial feature extraction and facial feature representation. These capabilities are crucial for making accurate facial expression recognition possible. In the previous chapter, two main methods for facial expression analysis were discussed and reviewed - video sequence processing and frame based image processing. Video sequence processing compares changes of adjacent frames, and although it is simple and straightforward, a significant weakness is that it cannot analyze static, single images. This means that frame based image processing techniques are more appropriate for this research.

A real time facial expression system includes three separate but inter-related main tasks. These are real time face detection, facial region image processing and facial expression analysis. The processing in each of these three parts is time consuming and

the way the tasks are organized and interact also affects performance of the entire system. In this chapter, an overview of entire real time facial expression analysis system and the integration of the three different components of the system are discussed. In addition, the setting in which the experiments were conducted is described.

3.1. Background

A requirement for a real time facial expressions analysis system is the correct recognition of facial expressions within reasonable time. In theory, 50 frames per second are practically considered flawless in computer animation, and 12 frames per second are considered as real time. It is also necessary to consider the size of the image - large size images may demand large memory and also CPU time to process, and on the other hand a small size image may not provide all the information needed to correctly identify the expression.

3.2. Equipment and environment

Poor lighting conditions make the process of facial feature extraction difficult - a problem also experienced by the human vision system. Also, skin

Figure 4.2.4.1.1 The set up for the normal environment including a computer & web camera

color, and noise within the image are other factors which reduce system accuracy. However it is important that any automatic facial expression analysis system is developed to be accurate in as wide a range of situations and conditions as possible. While it is impossible to design a perfect system suitable for any environment, it is possible to design systems to run within set tolerance levels. The development and testing in this research assumes normal lighting environment, with no single strong lighting coming from any direction. These conditions reflect what would be expected in indoor controlled lighting environments, such as schools, offices, retail shops and homes. Figure 3.1 depicts the testing environment.

In all experiments, people sit in front of a computer with a normal web camera operating. The web camera used is a Logitech QuickCam 4000 which connects to the computer through a USB port (shown in Figure 3.2). The computer used is an Intel Pentium 4 CPU 2.0 GHz. Using this camera and computer combination, 160x120 and 320x240 video can be captured at a rate of 30 frames per second. However for 640x480 video, the frame rate drops to 15 frames per second and also a little blurring occurs. For testing, it is assumed that the people are sitting in front of the computer and within 2 metres from the camera. With these criteria, the image size of 640x480 pixels was chosen as this is

Figure 4.2.4.1.1 Logitech QuickCam 4000 used for the experiments in the research

high enough quality for facial expression analysis. Because people sit in front of the camera, within 2 metres, the image size is then reduced by half for face detection (i.e. 320x240).

3.3. System structure

From a functional point of view, as discussed above, there are three main components in this system, face detection, facial feature extraction, and facial expression analysis. The diagram in Figure 3.3 shows the structure of the system.

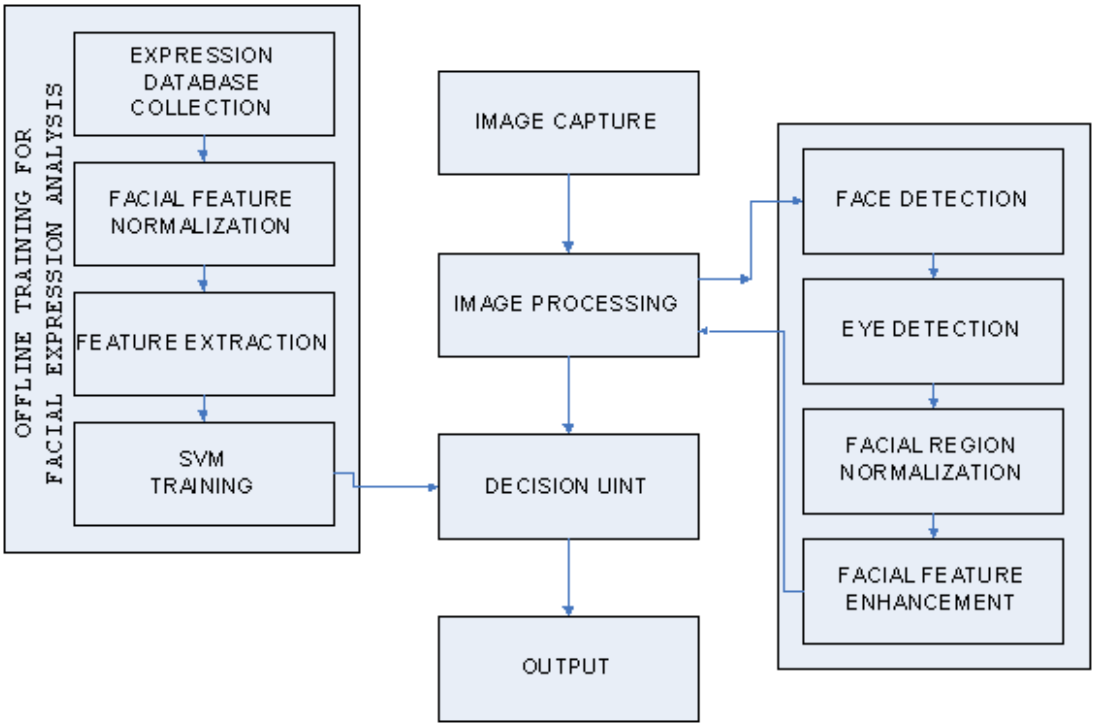


Figure 4.2.4.1.1 Real time facial expression analysis system structure

3.3.1. Face detection

The first stage is face detection. Real time face detection requires the face to be

located in the image captured from the camera as fast as possible. Researchers have had success designing algorithms suitable for detection of various objects, but detecting faces is particularly difficult as the face is a special kind of object that does not have a defined shape or uniform set of features.

In general, face detection occurs using the following process. Firstly, an image is captured by a digital camera. These images are then sent to the face detection system, which determines whether the image contains a face. The face detection system in this research uses two different filters to determine whether an image contains a face. First a fast template matching algorithm makes the initial decision about whether a face is present. Images which are considered to contain a face are then sent to the second, more accurate filter - a trained support vector machine (the training is conducted offline). Support vector machines are used because they are accurate at detecting faces as they have a strong learning ability and converge easily. While the support vector machines are accurate to a certain degree, they are not yet very good at detecting faces which are even slightly rotated – beyond approximately 10 degrees in either direction. In order to solve this problem, which would clearly have an adverse affect on any face detection system designed to operate in a real world environment, this research has developed a real time face detection algorithm which is able to detect a face with wide varieties of angle rotations. This will be described in more detail in the following chapter. This technique can be applied to any type of object, and ability to detect fully rotated objects has a variety of useful applications, such as the detection of

cancer cells.

Following the face detection, the next step is to normalize the face region. The size of the face image is not always the same, so it is necessary to modify it to the same size before continuing with further processing. In order to do this correctly without deforming or degrading the image, the eyes must be detected using an eye classifier. In previous systems, eyes were detected by using a known parameter that the eye region is a darker region within the face. Using this knowledge, a histogram equalization algorithm can be applied and the approximate position of the eyes can be found by assuming that they are the darkest region within the face. Obviously this can lead to inaccurate results as sometimes other parts of the face are in fact the darkest region. However, this problem has been solved as this research has introduced a novel iris detection algorithm which detects and locates the actual eye, rather than just the approximate region of the eye. This is described in more detail in the following chapters.

3.3.2. Facial feature extraction

The second stage is facial feature extraction. Human face images are inherently different due to variations of age, sex, skin color, and external factors such as lighting conditions. In order to have an accurate facial expression recognition system, it is important that these features are normalized as much as possible.

After conducting testing of a variety of different methods, a number of techniques have been combined to achieve this normalization. These include effective lighting correction, sigma filter and contrast-limited adaptive histogram equalization, and edge preventing smoothing algorithms. When these techniques are organised and implemented in a coherent, effective way, the system is able to enhance and extract facial features from a wide variety of human faces and still achieve the goal of real time operation.

To remove the effects of skin color and variations in lighting conditions, the facial region of the image is converted to gray level and the lighting is corrected using a linear lighting correction algorithm. This algorithm averages lighting in all directions which nullifies the effect of differences in lighting and skin color. Image noise can also affect feature extraction quality, and there are many existing algorithms which eliminate noise from images. Lee's (1983) sigma filter was chosen as the face region noise removal algorithm because the sigma filter does not destroy edge information which is important for the recognition of facial features. An edge smoothing algorithm is applied to make the facial features more consistent. The facial features are then also further enhanced by contrast-limited adaptive histogram equalization (CLAHE). The facial features enhancement and CLAHE algorithms are described in more detail in Chapter 5.

3.3.3. Facial expression analysis

The last stage is the detection, analysis and recognition of the facial expressions themselves. In this part of the system, a trained support vector machine is used to classify different facial expressions. The support vector machine is trained offline using a database containing a variety of facial expressions. To produce a large facial expression database for better training results the faces in the facial expression database for training were collected from many different sources, and FaceGen software was used to generate additional facial expressions. This means that the training database is a combination of real and generated facial expression. The facial expression database also includes faces of individuals of different age, sex, and race. Furthermore, to achieve the best performance for the classifier, an exhaustive parameter searching algorithm is applied. Finding the most effective parameters for the support vector machine is a difficult task and the steps taken during this research are described in more detail in Chapter 6.

Accurate representation of facial features and reducing feature dimensions is also important in improving the machine learning ability. The novel algorithm developed in this research which reduces features dimensions by radon transformation and discrete cosine transformation is also described in detail in Chapter 6.

Chapter 4. Scale and Rotation Invariant Face Detection using Template Matching and SVM

4.1. Introduction

Accurately locating the face is a very important step for facial expression analysis. Face detection has become a specialised task in machine learning and machine vision as, unlike many other objects which are detected, faces are inherently varied without uniform features. Many researchers have developed different face detection algorithms based on different theories and methodologies.

In this research, it was important to test and compare a large variety of face detection techniques to determine which was most suitable to assist with achieving the stated goals. To evaluate an object detection algorithm, the robustness and performance of the algorithm are tested in a number of ways. In general, there are four criteria used to evaluate face detection algorithms: scale invariance, rotation invariance, deformation intolerance and speed.

- **Scale invariance**

  Rowley, Baluja and Kanade (1998) propose a method in which, like in many

detection algorithms, face searching is performed in an image pyramid to be scale invariant. However the calculations originally involved in this process are not efficient, which meant that this technique was not suitable for real time applications. Viola and Jones's (2001; 2002) method improved the speed significantly using an integral image for fast feature extraction which allows them to accurately and efficiently filter out images that do not contain faces.

- **Rotation invariance**

  Images which contain faces which are rotated beyond approximately 10 degrees (in either direction) are even more difficult to detect successfully. In fact, existing algorithms and face detection methods are unable to successfully detect images containing faces which are rotated more than 10 degrees. Both Rowley's and also Viola and Jones' methods suffer from this problem. This is because the Artificial Neural Network or the Adaboosting algorithm they have used only gives the ability to detect face rotation within 10 degrees of centre.

- **Deformation intolerance**

  Similar to rotation, inadvertent deformation of face images makes successful face detection more difficult. In 1999, the Scale Invariant Features Transformation (SIFT) approach was introduced by Lowe (1999), and updated in 2004 (Lowe 2004). This technique provides efficient functions to compute the difference of the Gaussian (DOG) pyramid (Burt, Adelson, 1983).

  SIFT matches objects by detecting invariant key-points in levels in the

Difference of Gaussian (DOG) pyramid (see Figure 4.1). The key-points descriptor is shown in Figure 4.2.

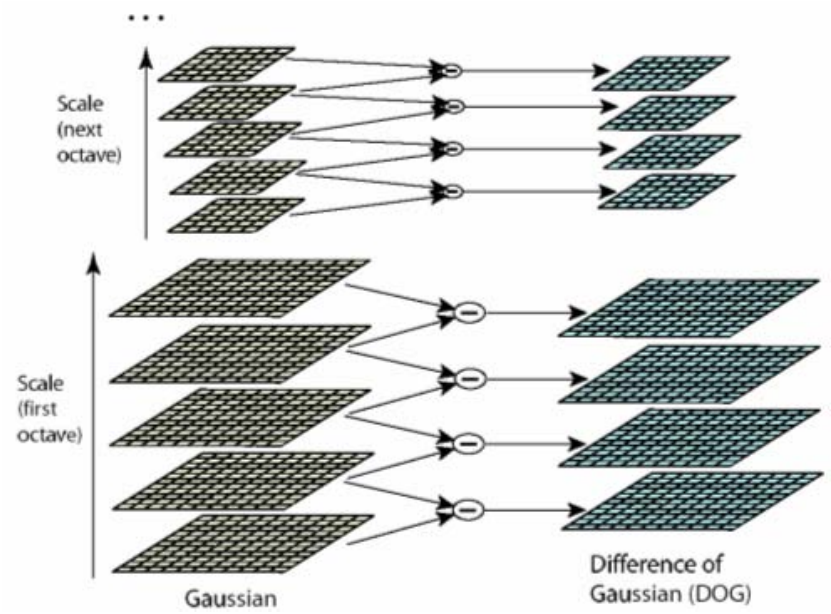


Figure 4.2.4.1.1 Difference of Gaussian (DOG) pyramid

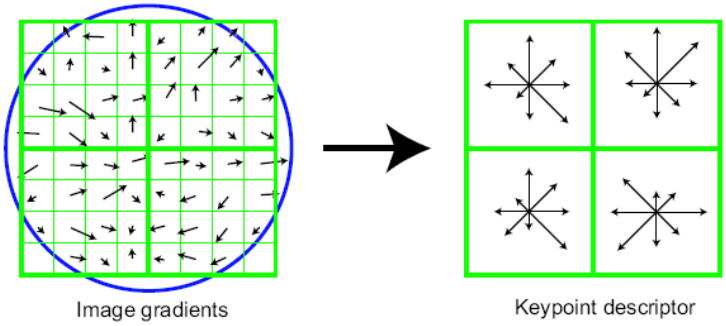


Figure 4.2.4.1.2 Key-point descriptor

The results of testing of Lowe's algorithm as shown in Figure 4.3, verified that the algorithm works very well for image matching by finding key points from the difference of the Gaussian pyramid. The properties of these key points are not significantly affected by different viewpoints. Figure 4.3 shows matched points for the same person at different view angles.

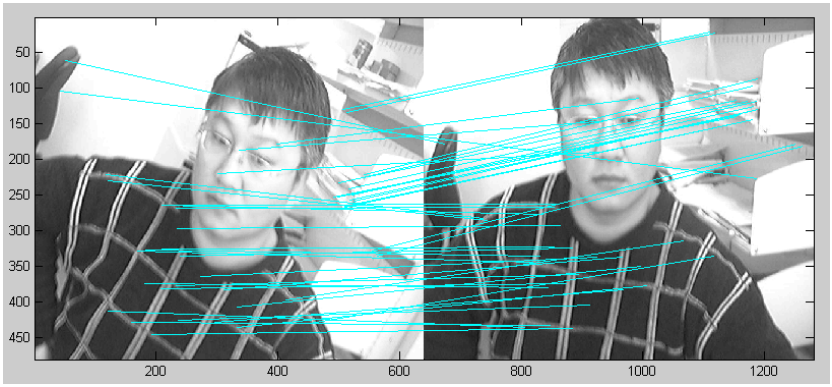


Figure 4.2.4.1.3 Matched image for the same person - 51 key points found.

However, despite the fact that these face images have similar properties (and are of the same person), there are only 51 matched key points. Figure 4.4 shows the result of matched key points for different people using Lowe's algorithm. This test shows that there are even less matched points found (33 matched key points). Furthermore, there are in fact no key-points found on the face region.

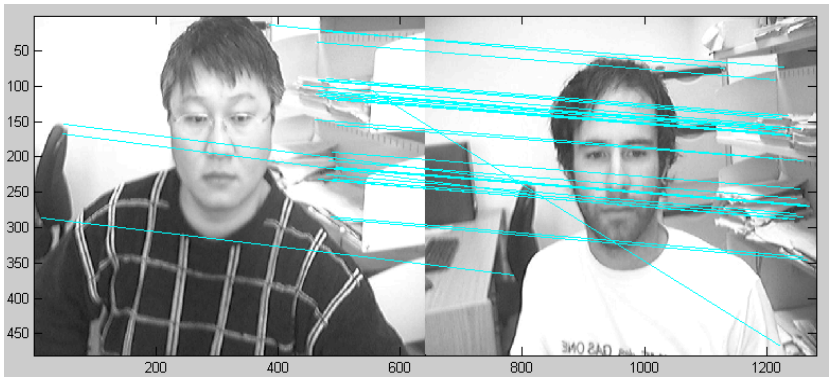


Figure 4.2.4.1.4 Matched image for different people - 33 key points found.

This confirms that face detection should be based on facial feature similarity and that therefore, SIFT is not suitable for face detection. Despite this, large size key-point descriptors combined with facial similarity measurements can improve the performance of the facial detection algorithms. This is discussed in future work.

- **Speed**

  Without question, speed is another important factor for evaluating real time facial detection algorithms, as some applications require detection to occur in real time. The NGITS system the subject of this research is one such application.

  All the different tasks that are required to be undertaken to ensure that a face is successfully recognised take a significant amount of computational power. This means that is it crucial that each task is finely tuned and efficient as possible and that the tasks interact coherently and concisely to effect the detection.

4.2. Optimized Candidates using Similarity Measurement (OCSM)

The proposed scale and rotation invariant algorithm developed as part of this research has been tested and compared with existing algorithms, based on the four criteria of scale invariance, rotation invariance, deformation intolerance, and speed. The details of the algorithm and comparison of results with existing techniques are discussed below.

Firstly, the average of a face image is calculated. The averaged face image is considered as a face sample for template matching. The averaged face image is generated as the mean face from a database of faces. An averaged face sample pyramid is also created by scaling down the number of pixels in each averaged sample face image until the size of each averaged face image is less than 24 pixels in width and 24 pixels in height. The processing of averaging and creating the face sample pyramid is pre-calculated offline.

Next, the system captures images which are blurred by the Gaussian filter to remove noise and then facial features are extracted by phase congruency. Phase congruency is used as it is one of the best algorithms to eliminate the effects of skin color and lighting condition.

Finally, face candidates are selected by fast convolution between the input image and the sample image based on Fourier Transformation. Fast convolution by Fourier Transformation is applied to find the positions in the input image which are similar to the average face image. The sample image is then tested against the averaged face pyramid, which (as described above) contains averaged face images of varying sizes, with a minimum size of 24x24 pixels. This means that a match between the sample face and the averaged face can be found, regardless of the size of the face in the input image. This process is, shown in Figure 4.5.

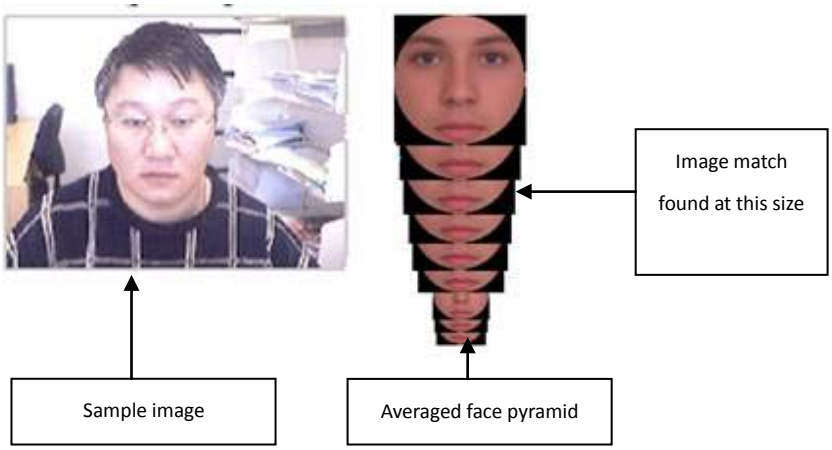


Figure 4.2.4.1.1 Face detection using an averaged face pyramid.

As previously described, successful detection of faces rotated beyond 10 degrees is particularly difficult. The way this research has approached this problem is to rotate the averaged faces by 10 degrees and to then reduce the scale of these images, thereby, creating a *rotated* averaged face pyramid. This process is repeated by rotating the face an additional 10 degrees and creating another rotated face pyramid as many times as desired. Again, this entire process can be pre-calculated offline and as only relatively simple multiplication is involved there is no significant increase in time required. Using this method faces rotated up to 180 degrees can be detected.

The final step in this process is to send the images which have been detected as having a face to a trained SVM for further classification and confirmation that a face is present. This algorithm combines organized fast template matching with the SVM. This technique advances the existing methodologies as it allows faces rotated by any number of degrees to be detected accurately in real time. The flowchart of the face detection algorithm is shown in Figure 4.6.

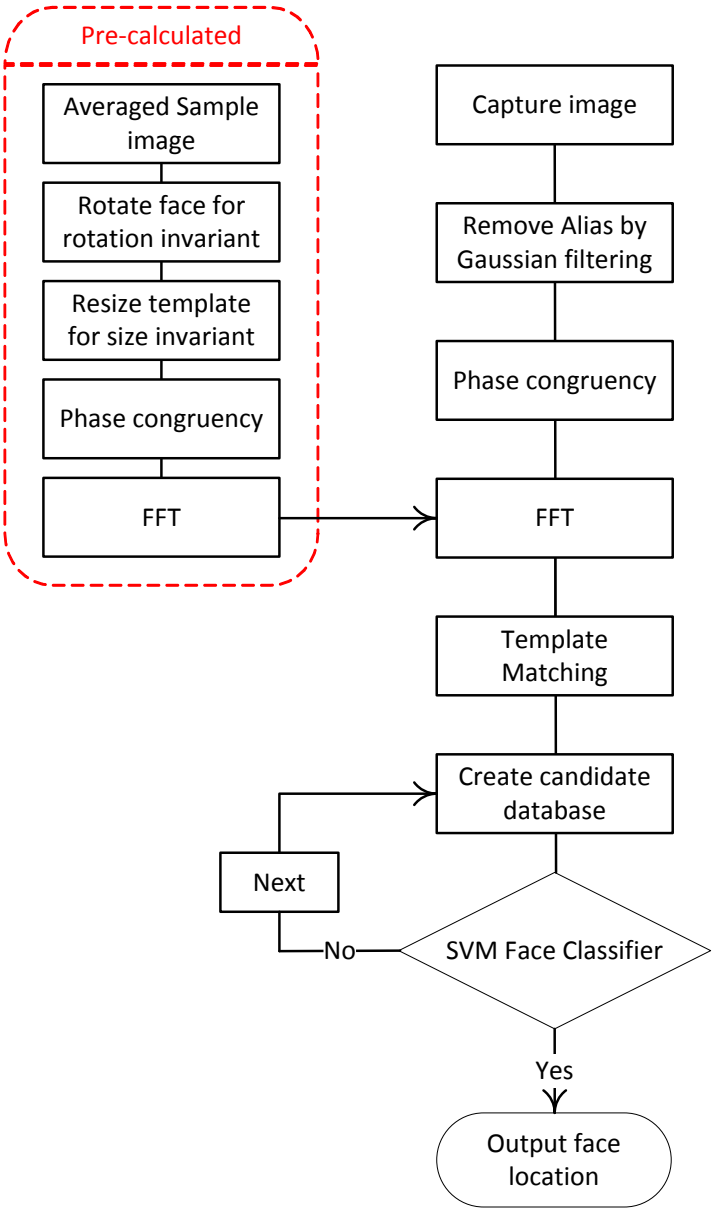


Figure 4.2.4.1.2 Flowchart of the scale and rotation invariant face detection

4.2.1. Image feature extraction by phase congruency algorithm

There are many feature extraction methods, such as color detection and edge detection which can be used for both faces and other objects. For example, it is relatively simple to separate a predominant foreground color from background colors,

but color information alone is not sufficient to recognize an object accurately – this is particularly true for faces which have a wide variation in color. Obviously, edges also provide important information about the features of an object. Sobel and Feldman (1968), Canny (1986) and others, have developed a variety of gradient based edge detection algorithms. Sobel and Feldman use a first order derivative:

$$\nabla I = \frac{\partial I}{\partial x} + \frac{\partial I}{\partial y}$$

And Canny use second order derivative:

$$\nabla^2 I = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2}$$

However using first and second order derivatives for template matching does not produce accurate results because, although they will show strong differences where there are clear differences in pixel values, these techniques are much less effective when the differences between the pixel values are much smaller, such as within a facial region of an image. Also first and second order derivatives can sometimes result in the edges within an image becoming thickened. While this is not a significant issue for the detection of some objects, when trying to detect a face, thickening of the edges will often distort the face features, which again adversely affects accuracy.

The research by Lim (1990) shows that phase congruency information can provide more accurate and simplified information about an image. According to Venkatesh and Owens' (1989) research, maximum phase congruency can be searched using peaks of

local energy function. In one dimension the local energy function is defined as:

$$E(x) = \sqrt{F^2(x) + H^2(x)}$$

Where F(x) is the signal I(x) with its DC component removed and H(x) is the Hilbert transform of F(x). Venkatesh and Owens' research also shows that the energy is equal to phase congruency multiplied by the sum of Fourier amplitudes.

$$E(x) = PC(x)\sum_n A_n$$

Morrone and Owens (1987) further define one dimensional phase congruency as:

$$pc(x) = \max_{\Phi(x)\in[0,2\pi]} \frac{\sum_n A_n \cos(\phi_n(x) - \Phi(x))}{\sum_n A_n}$$

Where, A_n is the amplitude of the n[th] Flourier component, $\phi_n(x)$ is the local phase of the Fourier component, and $\Phi(x)$ is the mean local phase angle. Figure 4.7 illustrates the relationship between phase congruency, local energy and the sum of Fourier amplitudes. The Polar diagram shows the Fourier components at a location in the signal plotted head to tail. The weighted mean phase angle is given by $\bar{\phi}(x)$. The noise circle represents the level of *E(x)* which can be expected just from the noise in the signal.

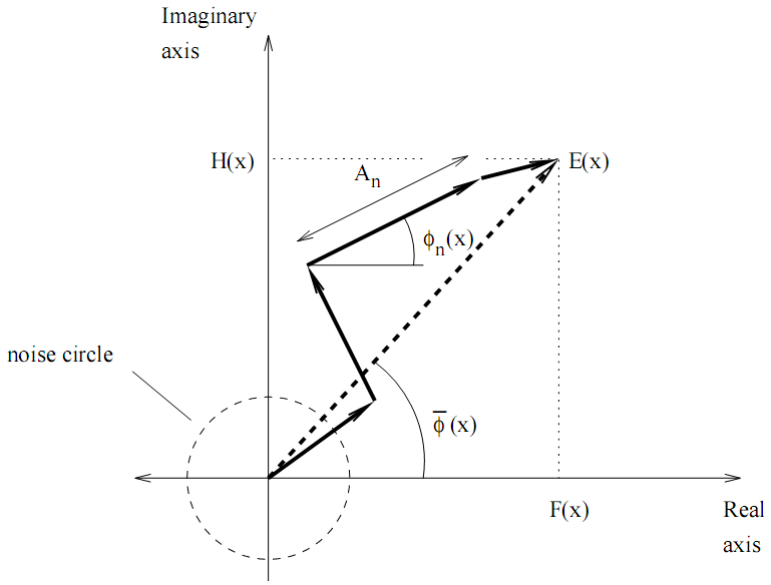


Figure 4.2.4.1.1 The relationship between phase congruency, Local energy and the sum of the Fourier amplitudes

For two dimensional image processing, phase congruency can be calculated as proposed by Kovesi (1993; 1999) using:

$$PC(x,y) = \frac{\sum_o \sum_s W(x) \lfloor A_{so}(x,y) \Delta\Phi_{so}(x,y) - T \rfloor}{\sum_o \sum_s A_{so}(x,y) + \varepsilon}$$

Where,

$$W(x) = \frac{1}{1 + e^{\gamma(c - s(x))}}$$

$$A_{so} = \sqrt{e_{so}(x,y)^2 + o_{so}(x,y)^2}$$

$$\Delta\Phi_{so}(x,y) = \cos(\phi_{so}(x,y) - \bar{\phi}_o(x,y)) - \left|\sin(\phi_{so}(x,y) - \bar{\phi}_o(x,y))\right|$$

$$T = \mu_R + k\sigma_R$$

and ε is a small constant to avoid division by zero.

In comparison with gradient based edge detection, feature extraction by phase congruency is more even and balanced. This research shows that phase congruency also produces more defined edge detection than gradient based edge detection. This is

primarily because gradient based edge detection doubles the thickness of an edge which often results in blurred or distorted edges, making feature detection and recognition more difficult. In contrast, however, edge detection using phase congruency does not, increase the thickness of edges. Figure 4.8 shows, in a sample of the current research, the comparative results of gradient based edge detection and phase congruency edge detection. The result of edge detection using the Sobel gradient based filter is shown in Figure 4.8 (b) and it is clear that the thickness of the feature edges has been increased, meaning they are actually less clear and more difficult to recognise. In comparison the features detected using phase congruency are clean and clear and can easily be used for the similarity measurements.

(a) Original image (b) Sobel filtered image (c) Phase congruency

Figure 4.2.4.1.2 Comparison of gradient based edge detection and phase congruency edge detection

4.2.2. Template matching for facial feature extraction

The fastest existing face detection is based on Viola and Jones' algorithm (Viola, Jones 2001) in which only face-like regions are fed to the classifier. This intelligent searching process is guided by an integral image as features can be extracted easily using the

integral image process. The process is also fast as the calculation used only involves relatively simple addition and subtraction. Face objects have many known similar features which, if used correctly, can significantly improve the performance of face detection systems. An example is eye regions which are generally darker than their surrounding regions.

However, there are some significant disadvantages of using this method. One problem is that it results in false detections, become some of the features used are found in other objects, as well as faces. For example, the contrasting light and dark regions formed by the eyes and their surrounding areas mentioned above is also formed by a variety of other objects in the real world which are not faces. In addition, this method may correctly detect that an image contains a face, but not detect the correct position of the face. This means that the accuracy of any subsequent processing of that particular image will be adversely affected. Finally, this method cannot detect faces rotated more than 10 degrees. This research has developed a more effective algorithm which uses a face template, Fast Fourier Transformation and classification by an SVM, and is described in more detail in the following sections.

The basic concept of template matching is to locate the face position by finding the minimum distance value between the search window and the template. Template matching uses Euler's measurement of distance between two objects.

$$TM = \min(\sum (S - T)^2)$$

where *S* is image of Search window and *T* is image of the template

Based on the formula above, if S is close to T, the output is close to zero which indicates a successful match. But this method is a rough estimate only, and it does not give an accurate determination of whether a face is present. Normally, in two dimensional image matching, the correlation coefficient can be used to measure similarity of two images to a higher degree of accuracy. Below is the formula for correlation coefficient between an image A and image B.

$$r = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{\left(\sum_m \sum_n (A_{mn} - \bar{A})^2\right)\left(\sum_m \sum_n (B_{mn} - \bar{B})^2\right)}}$$

where $\bar{A}$ = mean2(A), and $\bar{B}$ = mean2(B).

Correlation coefficient indicates the strength and direction of a linear relationship between two random variables. In image processing, correlation coefficients can be used to locate features within an image. This method employs a correlation to define the matching template and is sometimes known as template matching. A face template is created by averaging face images, including male and female faces from a range of different ages. Figure 4.9 shows a collection of different faces which have been selected from our database and used in this study and Figure 4.10 shows the averaged face.

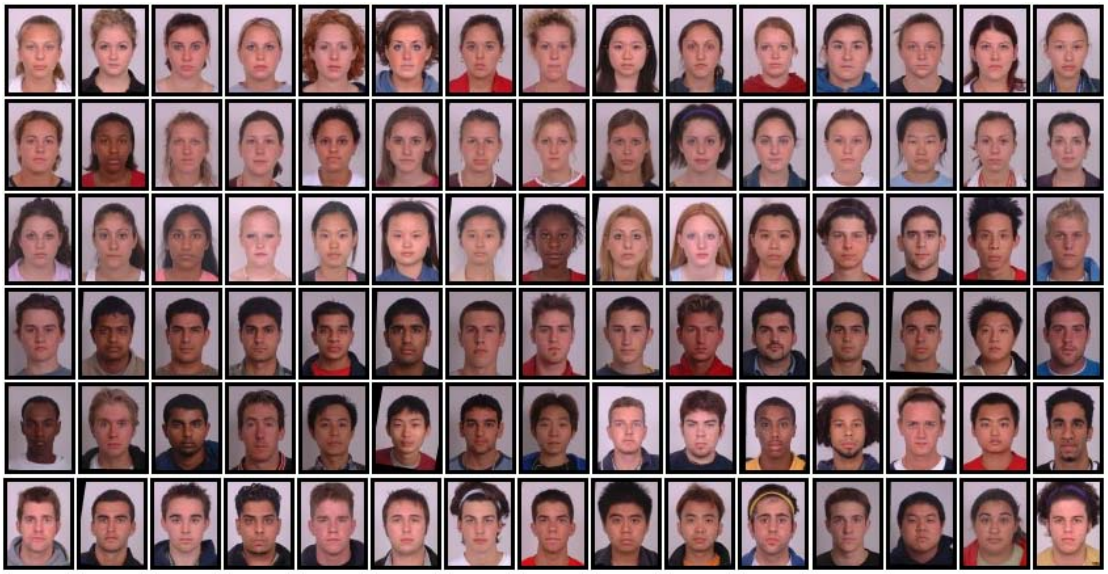


Figure 4.2.4.1.1 Selected images from the Face database.

(a) (b)

Figure 4.2.4.1.2 (a) Averaged face – color and (b) averaged face - grey level

Using the averaged face in the figure above, clear facial features are set and used in the template. Averaged face is also called mean face. This was created by collecting 400 face samples, normalizing by eye's location, summing together and dividing by the total number of face samples. The searching process finds all areas within each image which

also contain these facial features. The formula for template matching by correlation is explained below.

Given two images *I* and *T*, *I* is input image and *T* is template image (also known as the kernel image). The size of image *T* is *a* in width and *b* in height. The mean value of I is I_m, and the mean value of *T* is T_m. Correlation coefficients for two dimensional images are calculated using the in following formula:

$$R(x,y)=\frac{\sum_b\sum_a (I(x+a,y+b)-I_m)\times(T(a,b)-T_m)}{\sqrt{\sum_b\sum_a (I(x+a,y+b)-I_m)^2\times\sum_b\sum_a (T(a,b)-T_m)^2}}$$

However template matching algorithms are not applied widely for object detection. This is because gradient based algorithms cannot represent features accurately, and also traditional techniques for template matching are time consuming.

Convolutions can be applied for template matching – a higher output means a higher similarity with a kernel. In a template matching algorithm, an averaged face image is set as the kernel. The highest output of the convolution image indicates the most similar region with the kernel image (the averaged face). In comparison with other methods, template matching is an efficient algorithm for locating facial features and it does not require a significant amount of further image processing. However, there are still some weaknesses in these techniques:

- The highest value may not be a face, because it only gives the most similar region compared with the kernel.
- The false detection rate is increased due to variations in skin color or lighting condition changes. Some researchers have applied edge filters prior to template matching, but the rate of false detection still remains very high. The reason for this is that gradient based edge detection cannot represent features accurately and uneven edge detection will affect the result of template matching.
- Traditional calculation of convolution is computationally expensive.

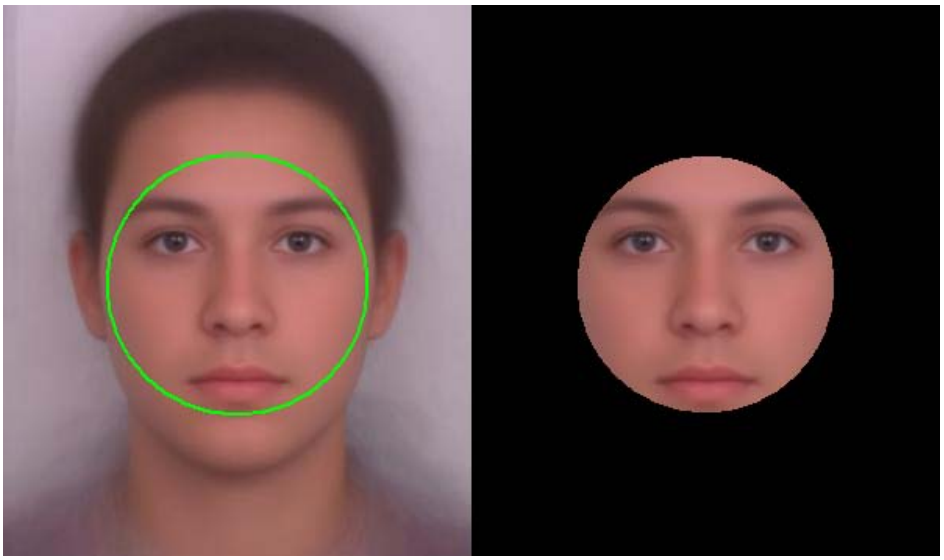
Because of these shortcomings, template matching has not played an important role for object detection for a time and researchers have focused on other techniques. This research aimed to reduce or eliminate these problems so that template matching could be used effectively.

Firstly, phase congruency, rather than gradient based edge detection is applied for feature extraction, because phase congruency represents features more accurately. Secondly, Fast Fourier Transform is used to improve the performance of template matching. Only images with locations similar with the averaged face template are considered as likely to contain a face. This greatly reduces the number of images which move to the next stage of processing and improves performance – both accuracy and

speed significantly. Finally, a trained SVM makes the final determination of whether or not a face is present.

4.2.3. Template for face matching

This research uses a circular shape for template matching. There are two reasons for this decision. The first is that a circular shape increases the accuracy of face detection as it ensures that unwanted features are excluded from the analysis. For face detection, only the eyes, nose and mouth are considered important features and a circular shape allows us to include those but exclude everything else. The other reason is that a circle is the best shape to assist with and facilitate rotated face detection. Figure 4.11 shows averaged face (a) and face templates (b).



(a) Averaged face (b) Face template

Figure 4.2.4.1.1 Training samples for face detection

4.2.4. Fast Fourier Transform

The Fourier series is a mathematical method for signal processing. It can also be applied to two dimensional images. The Fourier series was introduced by Baptiste Joseph Fourier, French mathematician and physicist. The methods of the Fourier series are to break down image signals to sinusoidal components. There are several common conventions for defining the continuous Fourier transformation algorithm in signal processing. It is defined using the following function:

$$X(f) = \int_{-\infty}^{+\infty} x(t) e^{-i 2\pi f t} dt$$

For Discrete Fourier transformation (DFT), Finite Fourier Transformation is defined using the following function:

$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N} kn} \qquad k = 0, \ldots, N-1$$

And Inverse Discrete Fourier Transform (IDFT) is defined as:

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} x_k e^{\frac{2\pi i}{N} kn} \qquad n = 0, \ldots, N-1$$

The original Discrete Fourier Transformation is a time consuming calculation. However the performance of Discrete Fourier Transform has been improved by Cooley and Tukey (1965), and others and is known as Fast Fourier Transformation. Evaluating these sums directly would take $O(N^2)$ arithmetic operations. Fast Fourier Transformation is an algorithm designed to compute the same result in only O(N log N) operations. The most well-known use of the Cooley-Tukey algorithm is to divide the transform into two

pieces of size N / 2 at each step, and is therefore limited to power-of-two sizes, but, in general, any factorization can be used.

4.2.4.1. Convolution and the Convolution theorem

Convolution is a method to help determine the effect of a given kernel image on an input image. Continuous convolution is defined as follows, where *'a'* is image, and '*k*' is a convolution mask also called a kernel:

$$(a \otimes k)(t) = \int_m^n a(\tau)k(t-\tau)d\tau$$

In discrete image processing, discrete continuous convolution can be defined as follows.

$$h(i, j) = \sum\sum a(i-m, j-n)k(m, n)$$

However, this is a time consuming operation when the kernel becomes large and this technique cannot be used in real time applications. However, the Convolution Theorem gives useful properties for fast convolution calculation. Again, let *'a'* denote an image, and *'k'* convolution mask, and $a \otimes k$ the convolution of *'a'* and *'k'*. *F* represents Fourier transformation. F^{-1} represents inverse Fourier transformation. The convolution theorem shows that the convolution of image *'a'* and *'k'* can be calculated by inverse Fourier transformation of *a'*s Fourier transformation multiplied by *k'*s Fourier transformation.

$$a \otimes k = F^{-1}[F[a] \times F\{k]]$$

because $a(t) = F_r^{-1}[F(v)](t) = \int_{-\infty}^{+\infty} F(v)e^{2\pi irt} d$ 、

and $k(t) = F_r^{-1}[K(v)](t) = \int_{-\infty}^{+\infty} K(v)e^{2\pi irt} d$ 、

$$a \otimes k = \int_{-\infty}^{+\infty} g(t')a(t-t')dt'$$

$$a \otimes k = \int_{-\infty}^{+\infty} k(t')\left[\int_{-\infty}^{+\infty} F(v)e^{2\pi ir(t-t')}dv\right]dt'$$

Interchanging the order of integration,

$$a \otimes k = \int_{-\infty}^{+\infty} F(v)\left[\int_{-\infty}^{+\infty} g(t')e^{-2\pi irt'}dt'\right]e^{2\pi irt}dv$$

$$a \otimes k = \int_{-\infty}^{+\infty} F(v)K(v)e^{2\pi irt}dv$$

$$a \otimes k = F_r^{-1}[F(v)K(v)](t)$$

So applying a Fourier transform to each side, we have

$$F\left[a \otimes k\right] = F[a] \times F[k]$$

$$a \otimes k = F^{-1}[F[a] \times F[k]]$$

With this property, image convolution can be calculated efficiently by FFT. FFT has various advantages including

- Analysis of data in frequency domain.
- Improving performance of image convolution.
- And other useful properties which are described in the following section.

4.2.5. Scale invariance and rotation invariance

The size of face images can vary depending on the distance from the face to the camera. Rowley, Baluja and Kanade (1998) detected faces using an image pyramid. The Gaussian pyramid was introduced by Burt and Adelson (1983), and improved in 1999 (Lowe 1999) and 2004 (Lowe 2004). Lowe applied the difference of Gaussian pyramid to find scale invariant key-points. All these methods are required to create a pyramid based on input images.

Searching for a face object or finding features on an image pyramid or difference of Gaussian pyramid requires a significant amount of computational power. Testing of the convolution theorem during this research produced some interesting results. Let *F* denote Fast Fourier Transformation, F^{-1} denote inverse Fast Fourier Transformation, *a* denote input image, *k* denote kernel or template and *g* denote a variable-scale Gaussian. The scale space of kernel image can be defined as a function $L(x, y, \sigma)$, and it can be calculated by:

$$L(x, y, \sigma) = g(x, y, \sigma) \otimes k(x, y, \sigma)\text{, or in short: } L = g \otimes k$$

If we calculate the fast convolution theorem with an input image, then we get:

$$F^{-1}(F[a] \times F[L]) = F^{-1}(F[a] \times F[g \otimes k])$$

Applying the convolution theorem again we then have:

$$F^{-1}(F[a] \times F[L]) = F^{-1}(F[a] \times F[F^{-1}[F[g] \times F[k]]])$$

Eliminate Fourier Transformation marked in red, and then we have:

$$F^{-1}(F[a]\times F[L]) = F^{-1}(F[a]\times F(g)\times F(k))$$

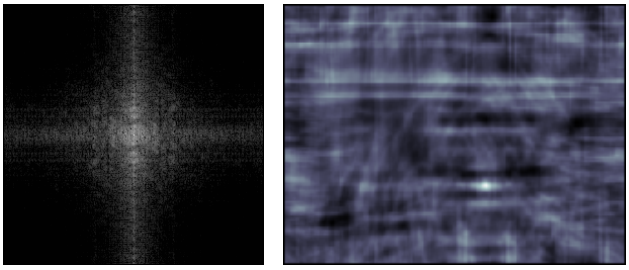
The above formula shows important properties:

- In a template matching algorithm, convolution of input image and different size face template can be calculated effectively.
- $F(g), F(k), L = g \otimes k$ can be calculated offline, which saves a significant amount of computational time.
- $F[a]\times F(g)\times F(k)$ is the only modification involved, by analysis in frequency domain. Only the highest value is considered as the face location.

As SVM and ANN currently have approximately 10 degrees tolerance for detecting rotated objects, we first rotated the template image for 35 times 10 degree each time to cover a full 360 degree rotation of the object (shown in Figure 4.12 below). The next step was to scale each rotated image and created a template database which allowed for variations in scale and in rotation. As explained above, all these calculations can be performed off line meaning that they require large amounts of memory rather than of consuming more computational time.

Figure 4.2.4.1.1 Rotated Face templates

From formula $F^{-1}(F[a]\times F[L]) = F^{-1}(F[a]\times F(g)\times F(k))$, the term $F[a]\times F(g)\times F(k)$ is shown in Figure 4.13(a), and $F^{-1}(F[a]\times F(g)\times F(k))$ is shown in Figure 4.13(b). Obviously, the centre value of Fast Fourier Transformation always corresponds with the maximum value of inverse Fourier Transformation, which is maximum value of the correlation coefficients.



(a) (b)

Figure 4.2.4.1.2 Fourier Transformation and Correlation coefficients

In Figure 4.14, the original image and the template image are shown in (a) and (b), with features extracted by the phase congruency algorithm. The correlation coefficients map of the two images is calculated and shown in (c). The peak values which are the best locations are shown with a green dot (d). These are the locations that are most similar to the face template.

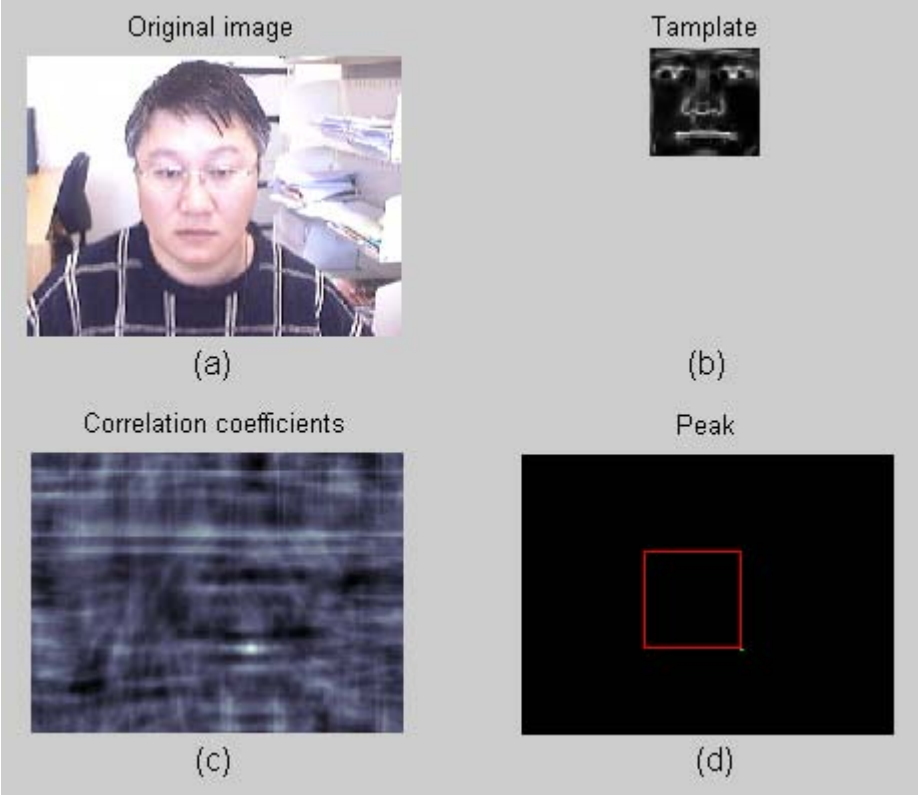


Figure 4.2.4.1.3 Image template matching by Correlation coefficients

The maximum output of the correlation coefficients gives the position within the sample image which most likely contains a face when compared with the template image. However, it does not mean that a face is detected. The images and corresponding locations are then sent to an SVM for final classification on whether a face is present or not. The training of the SVM is explained in detail in the next section. There are 35 (rotation variance) x 29 (scale variance) = 1015 templates. Some

templates may overlap which means that it is therefore unnecessary to send all candidates to the SVM. The values of $F[a]\times F(g)\times F(k)$ are stored and only the first five maximum outputs are considered candidates. These are sent to the SVM for classification.

4.2.6. Face Classifier training using an SVM

The template matching algorithm can achieve the best matching result for face similarity and therefore finds the most likely location for a face within an image. However, this does not mean that a face will actually be detected within this region. The template matching is applied only to increase the speed of the selection of possible face candidates. Following this step it is necessary to classify images as containing face or not using an SVM or ANN. SVMs have better learning abilities and converge easily, and were therefore considered the most appropriate technique for use in this research. As discussed in the next section, the template matching algorithm is used to extract a possible face location and then the SVM makes the final decision of whether the image contains a face.

4.2.6.1. Face database and non-face database

The training database for face detection contains 4000 jpeg images, including 2000 face and 2000 non-face images. These images are selected from the following databases

available on the web.

- Frontal Face images from CMU.

  http://vasc.ri.cmu.edu/idb/html/face/frontal_images/

- CBCL from MIT

  http://cbcl.mit.edu/software-datasets/heisele/facerecognition-database.html

- JAFFE Database

  http://www.kasrl.org/jaffe_download.html

- PAL Face Database

  https://pal.utdallas.edu/facedb/

- The CAS-PEAL Large-Scale Chinese Face Database and Baseline Evaluations"

  http://www.jdl.ac.cn/peal/index.html

These face images include different races, sex, ages, facial expressions and lighting conditions. The faces used are normalized prior to training. Each image is 24 pixels in widths by 24 pixels in height. Figure 4.15 shows two samples from the face training database.

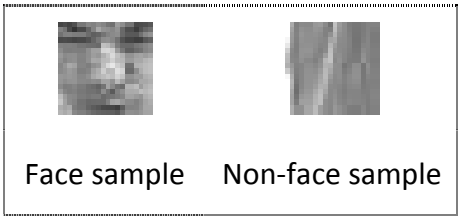


Figure 4.2.6.1.1 Training samples for face detection

4.2.6.2.Face classification

As discussed in the explanation of template matching by Fast Fourier Transformation, the computational complexity of classification will be decreased significantly. SVMs and ANNs are both effective tools to classify face patterns. During this research, both techniques were studied and compared.

For ANN, a feed-forward back-propagation network was trained to classify objects. The structure of ANN is shown in following figure (Figure 4.16), and the detail of the training of the ANN is presented in Chapter 6.

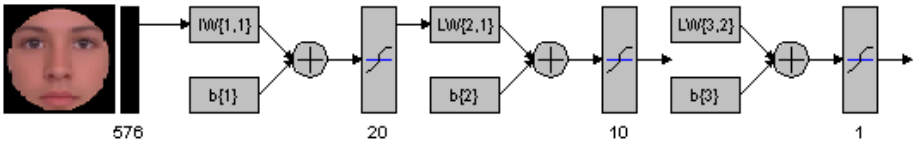


Figure 4.2.6.2.1 Artificial Neural Networks for Face Detection.

For SVM, the LIBSVM package was used as the training database to find support vectors. The kernel function used was a radial basis function as follows:

$$\phi = e^{(-\gamma|x_i - x_j|^2)}$$

Decision function used was as follows:

$$f_j(x) = sign(\sum_{i=1}^{j} \beta_i k(x, z_i) + b_j).$$

The details of training procedure for SVM are presented in Chapter 6.

4.2.7. Experimental result

Most face detection algorithms use a facial features extractor combined with a face classifier. In Viola and Jones' (2001) method, an integral image for feature extraction plus Adabooting classifiers has been applied. According to Viola and Jones's methods, the speed of the cascaded detector is directly related to the number of features evaluated per scanned sub-window. A C++ implementation of Viola and Jones's methods is freely available as a download from the Intel website and is known as the OpenCV detection method. The novel method developed in this research uses an effective combination of phase congruency and support vector machine (PC + SVM) and has a considerable advantage over OpenCV. It is more efficient for facial features extraction, which results in a significantly lower number of possible face candidate's locations being selected for further evaluation by the SVM. However, the performance of the novel method proposed in this research is slightly slower than OpenCV detection. The average performance comparison is shown in Figure 4.17.

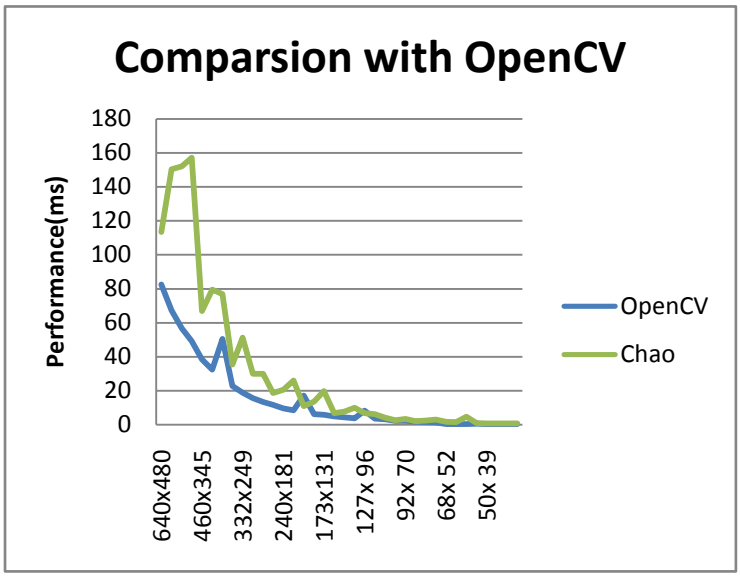


Figure 4.2.6.2.1 Comparison result with openCV

More importantly however, the novel technique, allows a rotated face to be detected – something not possible using existing methodologies. Figure 4.18 shows result of rotated face detection using the algorithm developed.

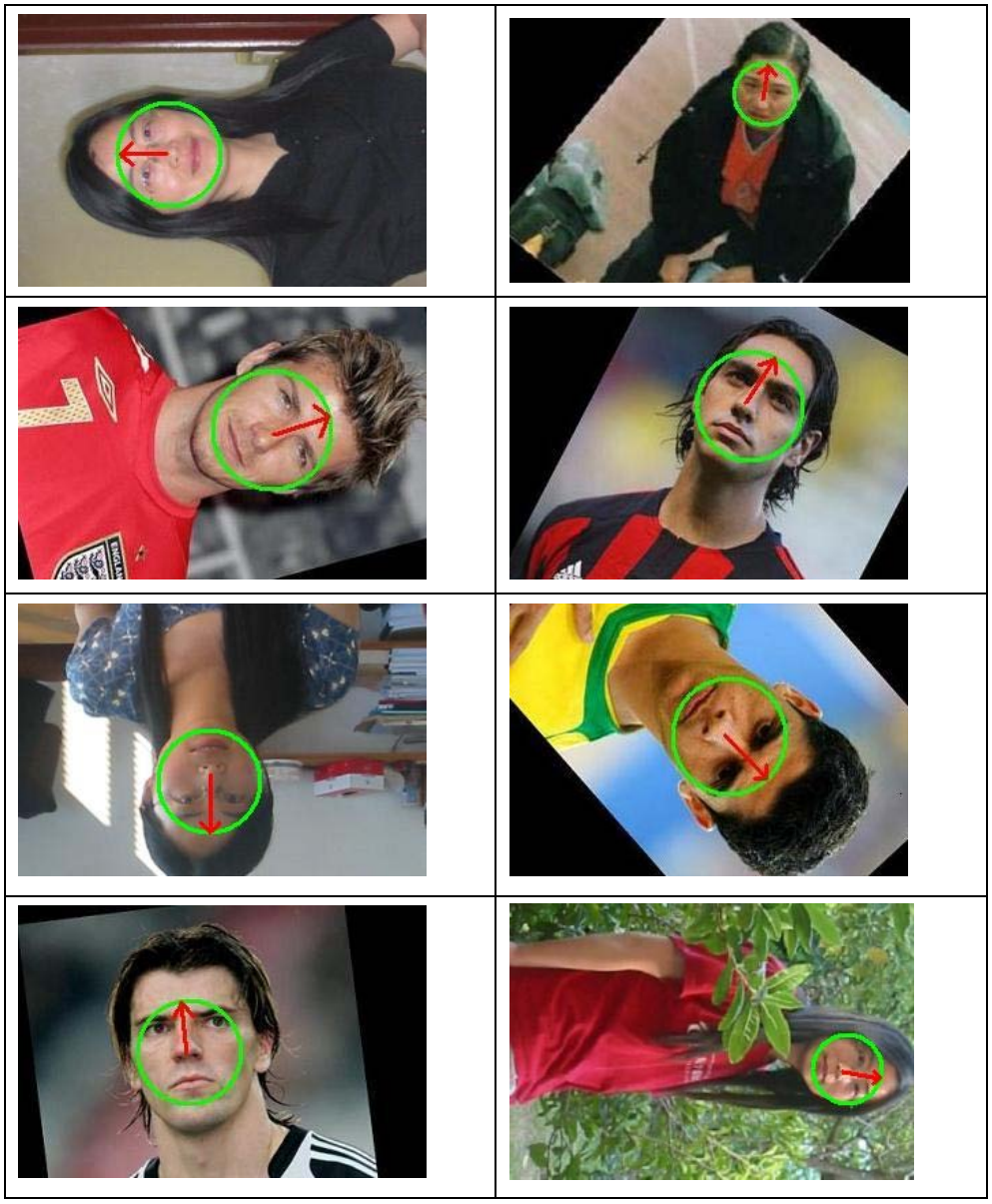


Figure 4.2.6.2.2 Result of rotate face detection

Images were rotated to a variety of angles and then tested using OpenCV and the proposed PC + SVM methodology. The test results show that our PC + SVM algorithm is able to detect faces at any angle. OpenCV, however, is limited to rotations of less than 10 degrees (in either direction). Figure 4.19 below shows the results of the testing.

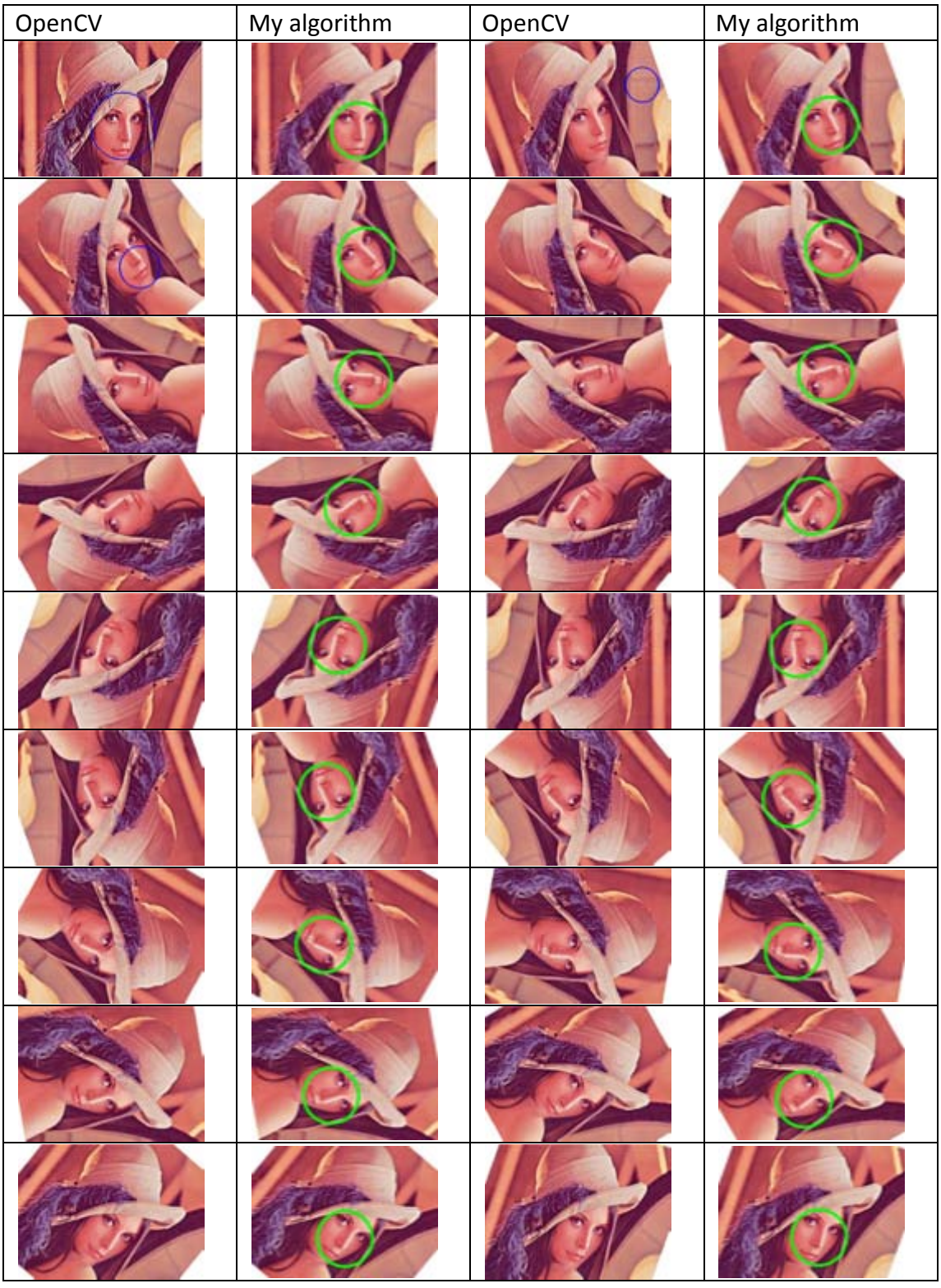


Figure 4.2.6.2.3 Comparison result with openCV

Results presented in Figure 4.19 and the performance and accuracy presented in Figure 4.20 and Figure 4.21 clearly show the higher rate of correct face detection using the proposed PC + SVM methodology.

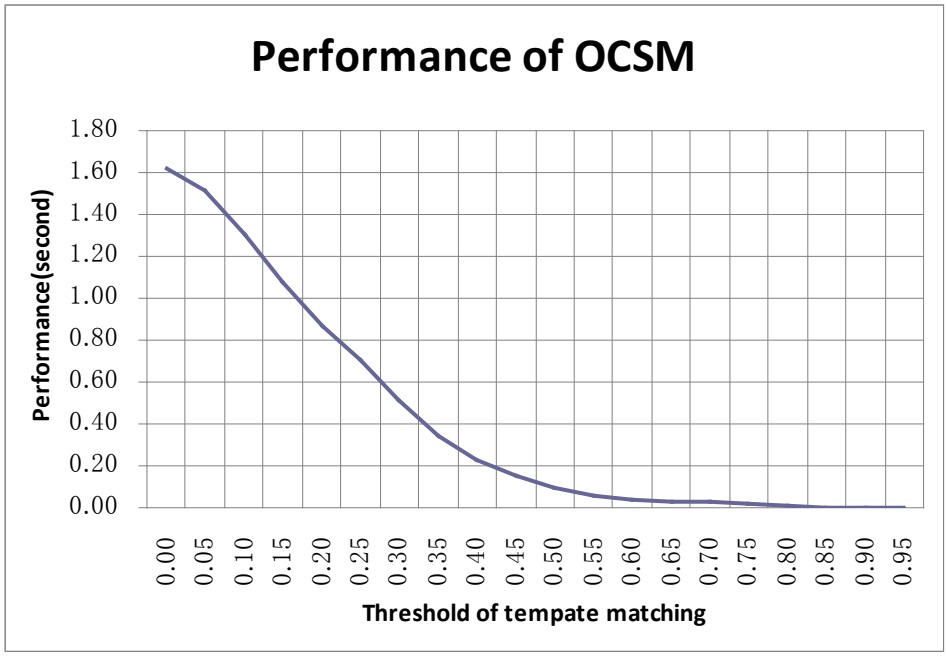


Figure 4.20 Performance of OCSM

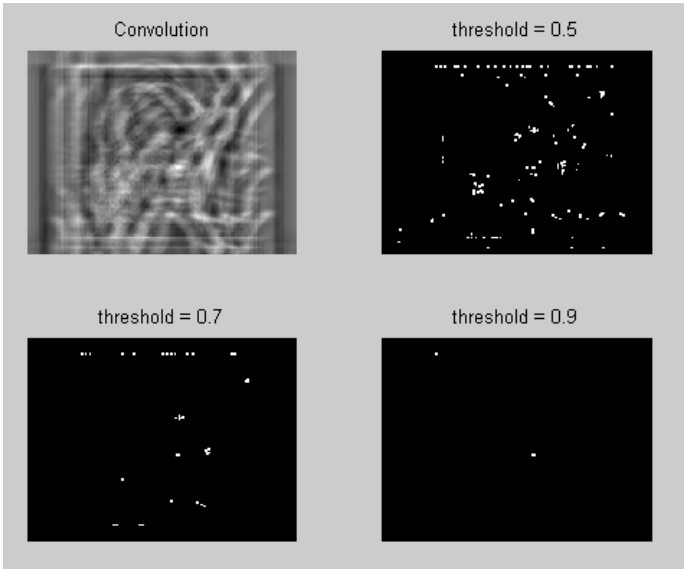


Figure 4.21 Accuracy of OCSM

Chapter 5. Facial image enhancement

5.1. Introduction

In this chapter, image processing techniques for removing noise and enhancing facial features are discussed. The first step, which was introduced in Chapter 4, is to detect the face using Optimized Candidates using Similarity Measurement (OCSM) object detection. For further processing, it is important to then extract facial features as accurately and clearly as possible. Facial feature detection is highly dependent on edge information, but simply applying existing edge detection or edge filtering algorithms will result in the loss of important information. In addition, the accuracy of results will be affected by noise within the image. This means that prior to feature extraction, further image processing to enhance the face image and to remove noise is essential.

In the technique developed by this research, localized face average skin color is considered as background and subtracted from the original image. This allows facial features to be enhanced without losing edge information containing facial features. The proposed facial image enhancement methods improve the performance of feature extraction algorithms and achieve better end results.

In this research, the facial image is captured as described in Chapter 4 – with the location of the facial region being the region of interest. In this chapter, different

histogram analysis algorithms are compared and tested, and a new algorithm for highlighting and analysing the facial region is presented. The results of the testing indicate that sigma filtering is the most effective method for the removal of noise within the image. The best results for facial feature enhancement are achieved using Contrast-Limited Adaptive Histogram Equalization (CLAHE) (Zuiderveld 1994), which is followed by edge preserving and smoothing. The whole process is well designed to avoid destroying the edge information which represents and distinguishes important facial features. This process for face image enhancement can be divided into 4 steps as shown in Figure 5.1. Each step will be described in detail in the following sections.

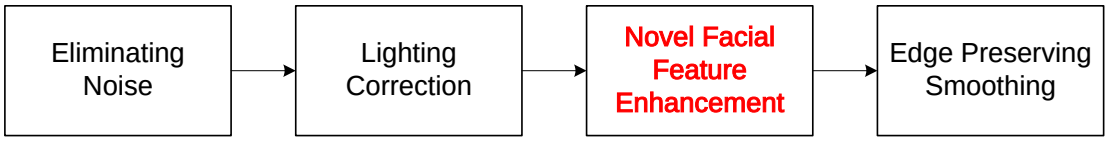


Figure 4.2.6.2.1 Steps required for facial image enhancement.

5.2. Eliminating noise

Noise within an image may be generated in a number of ways, including by the hardware used to capture the image. It is virtually impossible to eliminate but there are many existing algorithms which can be used to reduce noise to an acceptable level. Examples of these algorithms include image averaging, sigma filtering, K-means clustering, and mean-shift clustering, Fourier analysis and wavelet transform.

Each of the variety of different algorithms for noise elimination is suitable for different

and specific tasks. For example, the techniques which use pixel average filtering algorithms are suitable for large size images. However, these pixel averaging techniques destroy edge information, which is important for some applications, such as medical imaging. Other methods to remove noise are controlled by parameters. Examples of these are Gaussian and sigma filtering.

In order to assist with the evaluation of different image restoration methods, each method was placed into one of three broad categories - neighbourhood operation, spatial analysis, and statistical analysis, shown in Table 5.1.

Table 5.1 Methods for image noise remove

Neighbourhood operation:	Average filtering
	Median filtering
	Sigma filtering
	Gaussian filtering
	Wiener filtering
Spatial analysis:	Fast Fourier Transformation (FFT)
	Discrete Fourier Transformation (DFT)
	Wavelet Transformation
Statistical analysis:	Mean-Shift algorithm
	K-means algorithm

The performance of the image noise eliminating algorithms as applied to images of different size is summarized in Figure 5.2. The results show that the average filtering

algorithms (blue line) are the fastest, while Median and Wiener filtering have almost the same performance. Wavelet transform is the slowest technique.

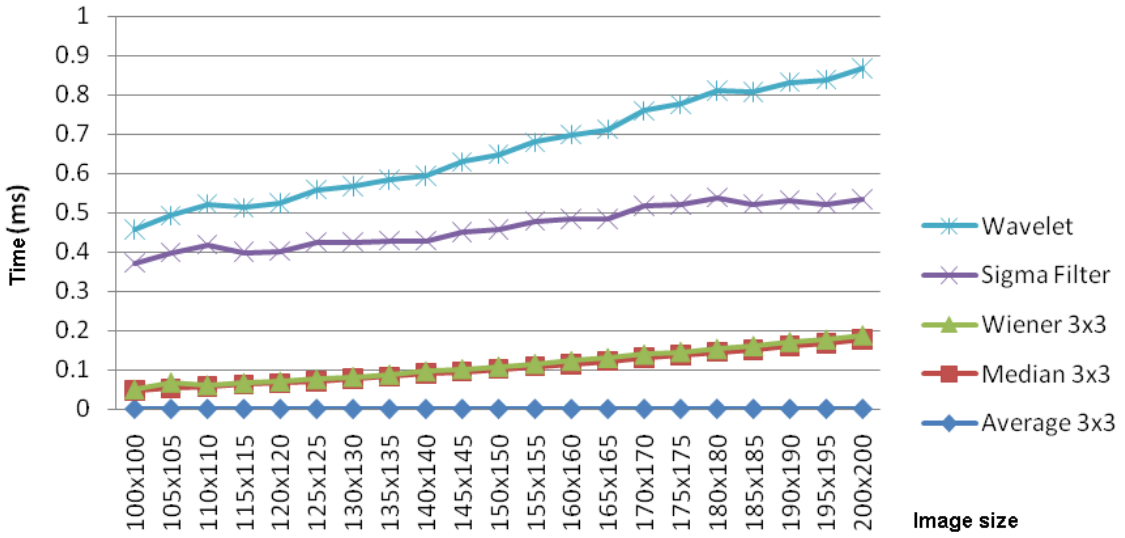


Figure 4.2.6.2.1 Performance comparsion on different size of images by different noise removal algorithms

It is also clear that the filter size plays an important role in the evaluation of the filter performance. Small size filter generates poor results, while large filter sizes tend to blur the image. Figures 5.3 and 5.4 show the results of applying different filtering techniques to facial information. Among the tested algorithms, the Wavelet transform is the best in removing noise from the face images but as shown in Figure 5.2, Wavelet requires the highest computation time. The result of sigma filtering is very close to that for wavelet noise removal. Moreover, if the parameter sigma is selected appropriately the sigma filtering method does not destroy the edge information (blue frames in Figures 5.3 and 5.4).

Figure 4.2.6.2.2 Comparison of noise removal algorithms as applied to face images without a moustache.

Figure 4.2.6.2.3 Comparison of noise removal algorithms as applied to face images with a moustache.

Research and testing showed that average filtering, Median filtering, Gaussian filtering, and Wiener filtering algorithms blurred the edge information equally and to an unacceptable degree. Moreover, Discrete Fourier Transformation (DFT) and Wavelet Transformations are computationally expensive and therefore are not suitable for real time applications. After consideration of the performance and test results of different filtering techniques, it was clear that, for noise elimination, the sigma filter technique was the most appropriate for use in this research. However, it was also clear that the basic sigma filter while accurate, was still slow, and further modifications were necessary. In the following section, the modified version of sigma filtering algorithm for real time facial analysis which is used in this research is described in more detail.

5.2.1. Sigma Filtering – modified for real time applications

The algorithm for sigma filtering was original introduced by Lee (1983) (also known as Lee's filter). The sigma filtering algorithm consists of averaging only those gray values in a window which are less than a set value – known as "sigma". Parameter sigma is a threshold value, and the gray value of the central pixel cannot exceed that sigma. The advantage of sigma filtering is the resulting smoothing of the image through the removal of noise, without smoothing edges within the image. The pseudo code of sigma filtering is presented in Figure 5.5:

```
(define sigma = σ)
sum = 0;
number = 0;
M = Input(X,Y);
for each pixel Input(x,y) in the window with the centre at (X,Y):
    if (abs(Input(x,y)-M)<sigma) {
        sum = sum + Input(x,y);
        number = number + 1;
    }
```

Figure 4.2.6.2.1 Pseudo code of sigma filtering

The original sigma filtering technique is a non-linear procedure and it is not efficient. In the worst case, it needs about $4 \times W^2$ operations per pixel, where W is window size. Kovalevsky (1989) improved the sigma filtering by using a local histogram. The histogram is an array, in which each element contains the number of occurrences of the corresponding gray value in the window. The sigma filter calculates the histogram for each location of the window by the means of the updating procedure that follows the histogram. Grey values in the vertical column at the right border of the window are used to increase the corresponding values of the histogram, while the values at the left border are used to decrease them. This reduces the number of operations per pixel to approximately $2 \times W + 2 \times (2 \times sigma + 1)$. The following figure shows the result of sigma filtering.

Figure 4.2.6.2.2 Sigma Filtering (Sigma = 10)

These modifications to the original sigma filter make it much more time efficient, without any significant loss of accuracy which means that this technique is the most appropriate noise elimination method for use in this research,

5.3. Lighting correction

Lighting conditions and differences in skin color also make facial expression analysis more difficult. There are two main methods for correcting lighting. One method is to analyze the image background color using statistical regression analysis. This method can be applied when images are complex. Another method is to apply a morph logical operation, which can be used most effectively when the shape of the object is well defined. The morph logical operation has a strong ability to enhance features when compared with other image enhancement algorithms. The background can be estimated and subtracted from the original image to enhance the features. The resulting light correction will make the lighting balanced in all directions.

After evaluating different methods, the morph logical approach was chosen as the most appropriate lighting correction algorithm for use in this research. Because the face has already been detected and normalized in the previous phase, the face shape is well defined, which means that the morph logical operation can be used effectively. Another significant advantage of morphological operation is fast performance, which makes it suitable for real time applications. Using the morph logic operation, the skin

color is considered background color and everything which is not the background color – i.e. all the facial features – are what are focused on and enhanced in the next stage of processing. The result of lighting correction on a sample image is shown in Figure 5.7.

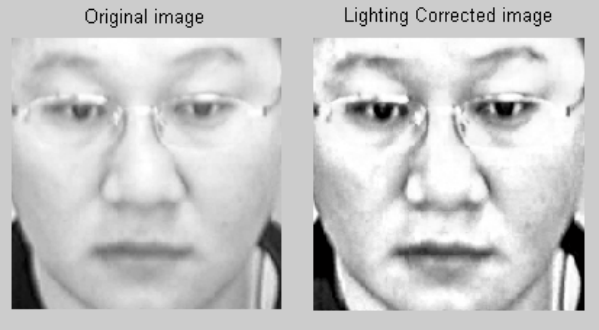


Figure 4.2.6.2.1 Result of Lighting Correction Using Morphological Operation

5.4. Facial features contrast enhancement

The procedure for removing noise is done prior to contrast enhancement to avoid enhancing the noise. An image which has been contrast enhanced is easier for the human eye to classify, and similarly, such images also provide more accurate results when analysed by computers.

As mentioned above, an efficient and accurate method to enhance facial features is to consider skin color as the background color. Facial features can then be enhanced by subtracting the background color from the face image. However, it is very difficult to find a threshold to distinguish between facial features and skin without using an additional method. Histogram, gamma and CLAHE techniques were reviewed during this research and are discussed below. Comparisons of images processed using each

technique is shown in Figure 5.8 below.

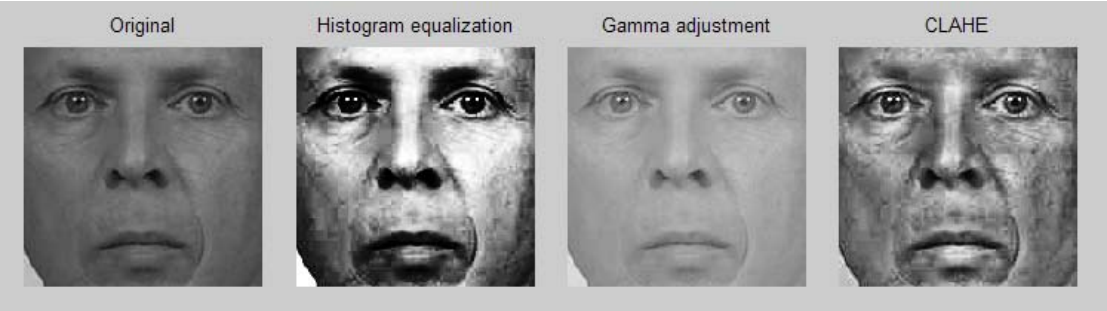


Figure 4.2.6.2.1 Different feature enhancement algorithms as applied to a face image.

Histogram analysis is a technique used in some image processing systems. For example, we can locate facial features by reviewing a face image's histogram. Figure 5.9 shows skin color estimation by analysis of histogram. The left image is the input image. The threshold to separate facial features is manually chosen as 0.6 (marked in red color). The right image is the threshold filtered image. The black region shows the approximate skin color. Facial region histogram analysis is important to identifying skin color and the histogram can be adjusted to obtain better contrast.

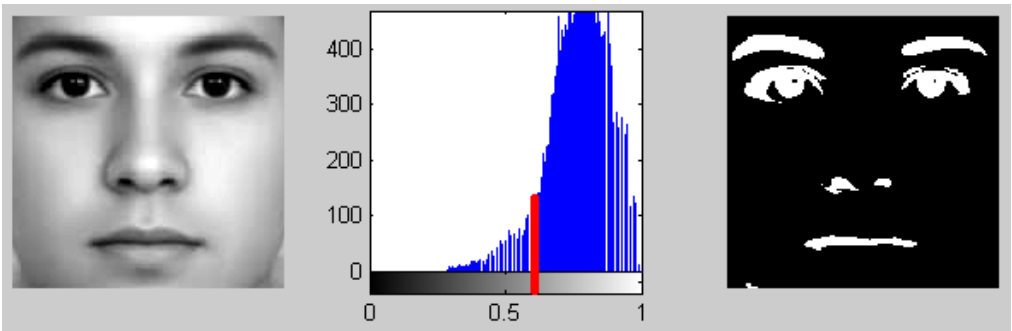


Figure 4.2.6.2.2 Skin color histogram analysis

However, histogram equalization enhances features, but the algorithm is not intelligent enough to separate these background pixels and pixels which denote important

features. This is a major disadvantage and means that this technique requires manual input for each image.

Gamma adjustment can also be used. With this technique, images are enhanced by adjusting gamma value manually, which the gamma value is moved upwards, to brighten, or downwards, to darken, the image. However, this method cannot be used automatically – it also requires a manual adjustment based on the lighting of each image.

Contrast-limited adaptive histogram equalization algorithm (CLAHE) was also reviewed. CLAHE is an improved method of histogram equalization. This technique analyses the histogram and enhances features *locally*, and gives better results for image enhancement. Traditionally, histogram equalization works by stretching an image's histogram to enhance the image - forcing an image's histogram to match with a specified histogram. Histogram equalization works well on small size images but poorly for larger size images. CLAHE algorithms (Zuiderveld 1994) (Umbaugh 1998) (Pizer et al. 1987) take this one step further and divide the image into small regions rather than analyzing the entire image all at once. These small regions are also called tiles. Each tile's histogram is analyzed and contrast enhanced. The neighbouring tiles are combined by using 'bilinear' interpolation to eliminate artificially induced boundaries. However, CLAHE has one major disadvantage - the result of facial feature extracted image by CLAHE cannot be further processed because these features are not clearly

distinguished from their background. If this problem could be solved, it was clear that the CLAHE technique would be the most appropriate method to enhance facial features. Therefore considerable effort was made during this research to solve this problem successfully, and the novel technique used is described below.

5.4.1. A novel facial feature enhancement method

As described above, each feature enhancement method reviewed had a variety of problems or limitations which mean that they are not suitable for the task proposed in this research. This has necessitated the creation of a new feature enhancement technique, based on CLAHE. In this research, the CLAHE image is known as the feature image.

In Figure 5.5 above, most pixels of a face image represent face skin color, so it is possible to consider skin color as background color. By subtracting the mean of skin color, an “averaged” face image can be obtained. Furthermore to more accurately extract skin color, we can mask out the face outline by using an oval mask. Facial features enhanced in the image are calculated by adding the average face image to a facial feature image, as described in the following steps and shown in Figure 5.10:

Step 1: Input image is masked by the oval face mask so that only facial region

is left.

Step 2: Calculate the mean skin color of the facial region and subtract it from the original image.

Step 3: Calculate facial features using CLAHE.

Step 4: Add facial features to the mean subtracted image created in Step 2.

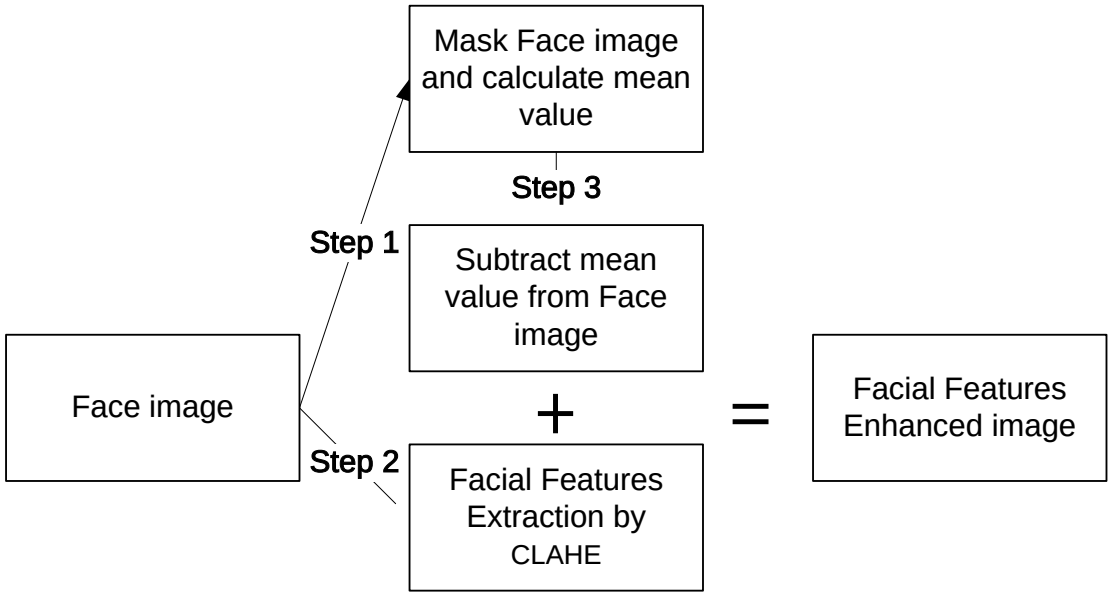


Figure 4.2.6.2.1 A novel method for facial feature enhancement

The following figure (Figure 5.11) shows this algorithm applied on faces, and Figure 5.12 shows more results on different people.

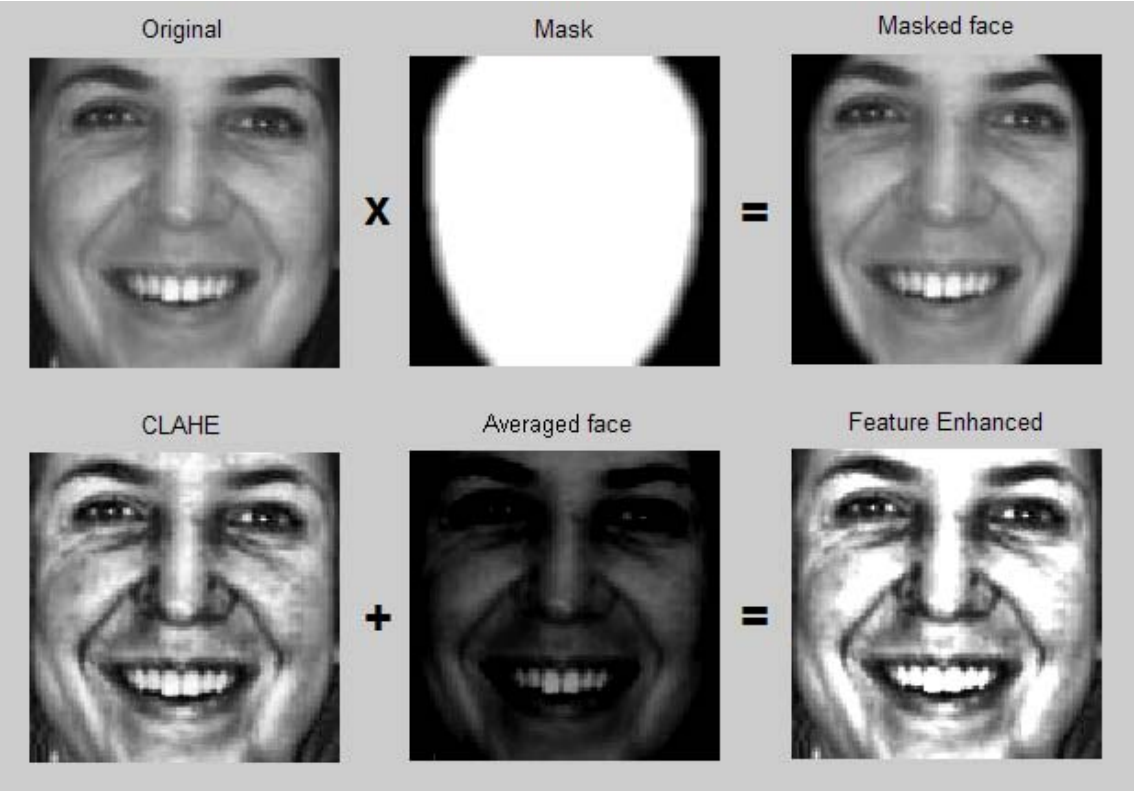


Figure 4.2.6.2.2 The results of applying the algorithms on faces

Figure 4.2.6.2.3 More test results of facial feature enhancement

5.5. Edge preserving smoothing process

After the noise has been removed from an image and the features have been enhanced, a further processing step can be undertaken to increase the accuracy of results. This is to group the pixels in order to reduce the number of duplicated features. Among the many algorithms available for grouping pixels, two were considered and compared in this study. The first method finds similarity between pixels by statistical analysis techniques such as K-means clustering and mean-shift clustering. The second method applies pixel neighbourhood operations which are known as efficient techniques in image processing. Examples of the second method which are based on filtering process include Tomita filter (Tomita, Tsuji 1997), Nagao filter (Nagao, Matsuyama 1979), and Kuwahara filter (Kuwahara, Eiho 1976).

Compared with image average filtering and Gaussian filtering, Kuwahara filtering is a non-linear filter. The purpose of Kuwahara filtering is to smooth images without blurring or sharpening the edges. In the Kuwahara filtering algorithm, for each pixel (known as the centre pixel), the mean and variance of its four sub-quadrant's (Region1, Region2, Region3, Region4 – shown in Figure 5.13) are calculated, and the mean value for the region with the smallest variance is chosen as the output value.

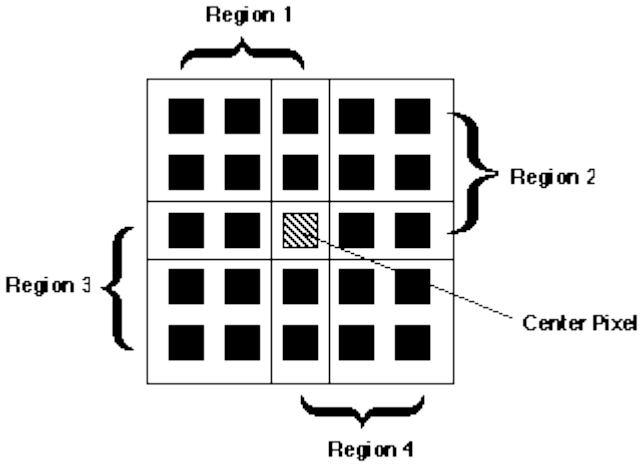


Figure 4.2.6.2.1 Kuwahara filtering of an image with 5x5 pixels.

The Kuwahara filtering can be implemented by a variety of different filter sizes. The number of operations required for Kuwahara filtering is O(K * K), where K is an integer value equal to the filter size. Young, Gerbrands, and van Vliet (1995) implemented and improved the performance of Kuwahara filtering by using column-wise calculations, which perform operations similar to that of distinct block processing for an image. It is about 38 times faster than the original Kuwahara filtering for a 480x640 image with the window size equal to 5.

The fast K-mean clustering algorithm (with k = 8) (Elkan 2003) and Kuwahara filtering were applied to an image. Figure 5.14 shows the performance of these two algorithms in terms of computational complexity and efficiency. After comparing these methods it was clear that the performance of improved Kuwahara filter is significantly faster and it does not destroy edge information. It was therefore considered the most suitable to use as an edge preserving smoothing algorithm in this research.

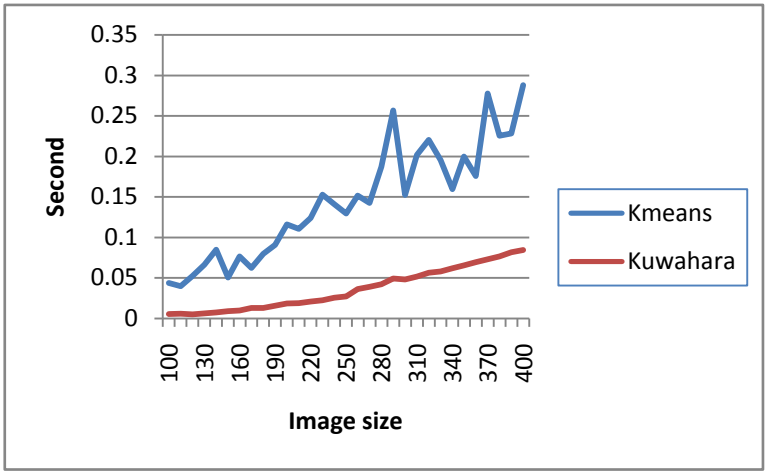


Figure 4.2.6.2.2 The performance comparison between K-means clustering and Kuwahara filtering

5.6. Summary of experimental results

There is not a single algorithm which can extract facial features accurately, but it is desirable to represent facial features as clearly as possible. Moreover, the computational complexity is another major factor in selecting a proper algorithm, particularly when, as with this research, it is a requirement that the application works in real time. A novel procedure for real-time facial feature extraction has been presented in this chapter. This procedure is well organized and results presented in Figure 5.15 show that it is suitable for most skin color, age, sex, and lighting conditions.

Figure 4.2.6.2.1 More results for facial features enhancement on different people using the novel algorithm developed during this research.

5.6.1. Image size

Another important consideration during this research was the most appropriate image size to use within the system. Large-size images are not ideal for representing facial features, because they consequently require a huge computer memory and computation time for processing, while images which are too small may result in some

information being missed during processing. A number of different size images were tested to choose the best image size for optimum performance. The relationship between image size and time required to process using the proposed facial feature enhancement algorithm is shown in Figure 5.16. Over all, it was concluded that an image size of 150 by 150 was the best image size as important feature information would not be lost and computational time was about 0.15 seconds - acceptable for a real time application.

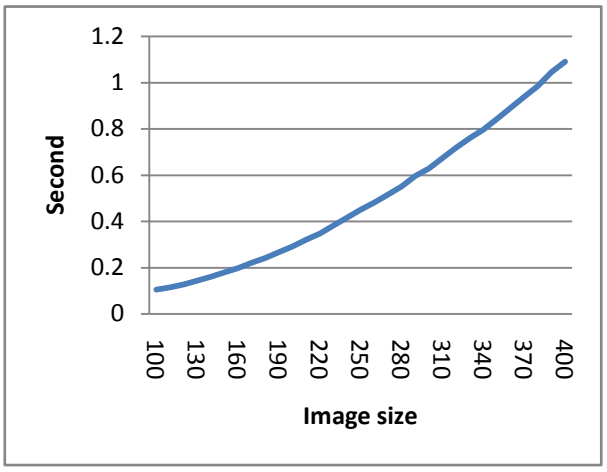


Figure 4.2.6.2.1 Time complexity of the Facial Feture Enhancement Algoritnm

5.7.Eye detection and face normalization

Within the images to be analysed by the system, the size of face regions will differ, which means that the image needs to be further normalized so that the regions are the same size for the next step of processing. This step is very important. It is also important that this normalization is done using a method which does not degrade or otherwise affect the image, by, for example, cutting off part of the facial region. To achieve this goal, this research introduced a novel method to quickly locate eyes within

the image. The eyes can then be used as a base location or marker from which the normalization can be based.

In order to locate the eyes an integral image is used to improve performance. The integral image allows fast computation of rectangular features. The integral image at location *x, y* contains the sum of the pixels above and to the left (Viola 2001; Viola 2002).

$$ii(x, y) = \sum_{x' \le x, y' \le y,} i(x', y')$$

The rectangular features calculation of integral image is shown in Figure 5.17.

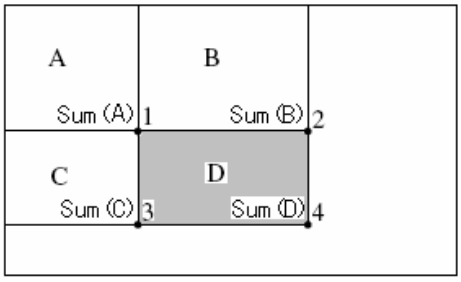


Figure 4.2.6.2.1 Integral image feature calculation

The sum value of all the grey pixels of D is Sum (D) + Sum (B) + Sum(C) – Sum (A).

For the eye region, the iris region is always the darkest region compared to the surrounding regions shown in Figure 5.18.

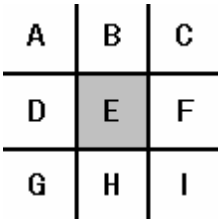


Figure 4.2.6.2.2 Property of eye region

Eye region has the following property:

Sum (A + B + C) > Sum (D + E +F)

Sum (G + H + I) > Sum (D + E +F)

Sum (A +D + G) > Sum (B + E +H)

Sum (C + F + I) > Sum (B + E +H)

Eye detection results are presented in Figure 5.19.

Figure 4.2.6.2.3 Result of eye detection and face detection

The approximate positions of the eyes are (37,112) and (112,112), as shown in Figure 5.20.

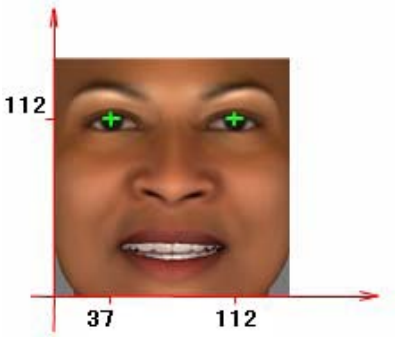


Figure 4.2.6.2.4 Positions of the eyes in the normalized image

Using this eye detection technique, the face normalisation can be undertaken accurately and without distorting the face itself.

5.8. Summary

In summary, various existing facial features enhancement algorithms were compared and tested. A novel facial feature enhancement technique which is fast and effective for real time image processing was also presented. In the next chapter, a novel facial expression analysis algorithm is introduced which employs a trained SVM to separate different expressions.

Chapter 6. Real time Facial expression analysis

There are two main issues that make automatic facial expression analysis difficult in comparison with the same process in humans. The first major issue is the relatively undeveloped ability of computers to learn. Although medical science has made a great deal of progress in uncovering the human brain structure and its amazing ability to learn, researchers have been unable to replicate this process in computers, meaning that the computer's learning ability is still very immature and simplistic – even compared with a two year old child's brain. A significant amount of research and experiments have been done to try to simulate human brain's behaviour in machines. ANNs and SVMs are two forms of artificial system which do show an amount of machine learning ability, and have been used with a degree of success in image processing.

The second major issue is the computation requirements for analysis - especially when the subject of the computation is an image or a video. Whether ANNs, SVMs or other computer learning algorithms are used, they are all computationally complex and require huge computer memory when processing images or video. This means that analysis of images or video can only be done in real time with a very expensive and sophisticated computer.

Feature extraction and representation play an important role in this context. Extracting facial features accurately and efficiently not only reduces computational complexity but also improves the classifier's learning ability over time. It is reasonable and acceptable to represent an image using different features (rather than the image itself). This makes the task of automatic facial expression analysis on a standard household computer in real time possible. Essa, Pentland (1998) and others, break facial features down to action units. Kanade, Tian, and Cohn (2002) tried to classify facial expressions by classifying these action units. As described in detail in Chapter 2, other researchers employed other facial expression classification methods including Optical Flow (Cohn et al. 1998) (Yacoob, Davis 1996), Wavelet-based Facial Feature Tracking (Wu 1997) (Wu et al. 1998), dynamic Bayesian networks (Essa, Pentland 1995), Muscle-Based Feature Models (Ohta, Saji, Nakatani 1998) and Hidden Markov Models (Ma et al. 2004) (Lien 1998). The purpose of all of these methods is to reduce the image dimensions while describing facial features as accurately and as concisely as possible for the classifier (i.e. SVM or ANN etc) to make the final decision without requiring an excessive amount of computational power.

ANNs and SVMs (as well as other methods) are widely and successfully applied in optical character recognition. But they are not competent with large, complex images which require more in depth and robust analysis, such as those used in this facial

expression analysis project. One of the reasons for this is that the image size for facial expression analysis is much larger than those used for optical characters recognition. In the context of character recognition for the English alphabet 32 by 32 pixels is enough and for more complex character sets such as that of Chinese 64 by 64 pixels is adequate. In addition to their small size, optical images which are input to character recognition systems also have prominent features and defined edges. As the image size and complexity increases the problems are more difficult to solve.

It is clear from the research conducted that a key issue to resolve in order to successfully achieve the stated goals is to determine whether computers can be enabled to recognise and process only features necessary for the classifier to make the correct decision, and at the same time ignore duplicate and otherwise useless information.

In this research, the facial region of the image is captured by algorithm introduced in Chapter 4, and facial features are enhanced by the algorithm described in Chapter 5. In this chapter, by analysing the advantages and disadvantages of existing method, an effective real time facial expressions analysis system is created. The components and processes used in the system are described below. Firstly, the novel facial features extraction algorithm is introduced and described. The efficient feature extraction

algorithm is an important link between image processing and machine learning. The algorithms represented in this research combine the radon transformation and Fast Fourier transformation and reduce the feature dimension significantly, while at the same time continuing to represent the image accurately. The algorithm can also easily be extended to classify objects other than faces.

- Secondly, comparison between SVMs and ANNs is made, and the reasons why SVMs are more efficient than ANNs in learning ability are discussed.

- Thirdly, experiments on optimal parameter selection and kernel selection for SVMs are presented.

- Finally the experimental results for the novel real time facial expression analysis system are presented. The results also confirm that this algorithm is able to analysis facial expression for video sequences.

6.1. Basic facial expressions

Humans often express their feelings through different facial expressions. These expressions are generated by facial muscles controlled by the brain. Although the human face can generate many different facial expressions, only a few facial expressions have significant meanings for communication in real life. This research focuses on these six basic facial expressions:

- Happy expression

Happy expressions are universally and easily recognized, and are interpreted as conveying messages related to enjoyment, pleasure, a positive disposition, and friendliness. Figure 6.1 shows some examples of facial expressions of happiness.

Figure 4.2.6.2.1 Examples of happy expressions

- Sad expression

Sad expressions are often conceived as opposite to happy ones, but this view is too simplistic. Depending on the situation, sad expressions convey a variety of messages related to loss, bereavement, discomfort, pain, helplessness, etc. Figure 6.2 shows some examples of facial expressions of sadness.

Figure 4.2.6.2.2 Examples expressions of Sadness

- Anger expression

Anger expressions are seen increasingly often in modern society, as daily stresses and frustrations underlying anger seem to increase, but the expectation of reprisals decrease with the higher sense of personal security. Figure 6.3 shows some examples of facial expressions of anger.

Figure 4.2.6.2.3 Examples of anger expressions

- Fear expression

Fear expressions are not often seen in societies which have good personal security, as the imminent possibility of personal harm, from interpersonal violence or impersonal dangers, is the primary elicitor of fear. Figure 6.4 shows some examples of facial expressions of fear.

Figure 4.2.6.2.4 Examples of fear expressions

- Disgust

Disgust expressions are often part of the body's responses to objects that are revolting and nauseating, such as rotting flesh, fecal matter, insects in food, or other offensive materials that are rejected as unsuitable to eat or touch. Figure 6.5 shows some examples of facial expressions of disgust.

Figure 4.2.6.2.5 Examples of disgust expressions

- Surprise expression

Genuine surprise expressions are usually fleeting, and difficult to detect or record in real time. They almost always occur in response to events that are unanticipated, and they convey messages about something being unexpected, sudden, novel, or amazing. The brief surprise expression is often followed by other expressions that reveal emotion in response to the surprise feeling or to the object of surprise, emotions such as happiness or fear. Figure 6.6 shows some examples of facial expressions of surprise.

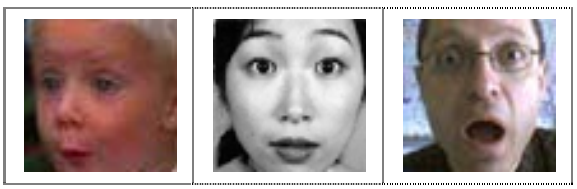
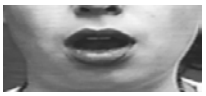


Figure 4.2.6.2.6 Examples of surprise expressions

These facial features for different expression are further summarised in Facial Action Coding System (FACS) (Ekman, Friesen 1978). The following table shows FACS.

Table 6.1 Facial Action Coding System

AU	Description	Facial muscle	**Example image**
1	Inner Brow Raiser	Frontalis pars medialis	
2	Outer Brow Raiser	Frontalis pars lateralis	
4	Brow Lowered	Corrugator supercilii Depressor supercilii	
5	Upper Lid Raiser	Levator palpebrae superioris	
6	Cheek Raiser	Orbicularis oculi pars orbitalis	
7	Lid Tightener	Orbicularis oculi pars palpebralis	
9	Nose Wrinkler	Levator labii superioris alaquae nasi	
10	Upper Lip Raiser	Levator labii superioris	
11	Nasolabial Deepener	Zygomaticus minor	
12	Lip Corner Puller	Zygomaticus major	
13	Cheek Puffer	Levator anguli oris (a.k.a. Caninus)	
14	Dimpler	Buccinator	
15	Lip Corner Depressor	Depressor anguli oris (a.k.a. Triangularis)	

16	Lower Lip Depressor	Depressor labii inferioris	
17	Chin Raiser	Mentalis	
18	Lip Puckerer	Incisivii labii superioris and Incisivii labii inferioris	
20	Lip stretcher	Risorius w/ platysma	
22	Lip Funneler	Orbicularis oris	
23	Lip Tightener	Orbicularis oris	
24	Lip Pressor	Orbicularis oris	
25	Lips part	Depressor labii inferioris or relaxation of Mentalis or Orbicularis oris	
26	Jaw Drop	Masseter relaxed Temporalis and internal Pterygoid	
27	Mouth Stretch	Pterygoids Digastric	
28	Lip Suck	Orbicularis oris	
41	Lid droop	Relaxation of Levator palpebrae superioris	
42	Slit	Orbicularis oculi	
43	Eyes Closed	elaxation of Levator palpebrae superioris; Orbicularis oculi pars palpebralis	
44	Squint	Orbicularis oculi pars palpebralis	

6.2. Facial feature extraction for facial expression analysis

In this research facial expression images were collected from the Internet and these images were separated into six groups as smile, laugh, surprise, sad, disgust, and normal expressions. For each image, facial features are enhanced and normalized to the same size (150 pixels in width and 150 pixels in height).

Many different experiments aimed at finding the best image size for training the SVMs or ANNs were undertaken. As discussed in the previous chapter, a large sized image will require huge memory as well as CPU time and may result in a reduction in accuracy as due to uncertain or ambiguous features, while a small sized image will result in the loss of some useful information. Experiments were necessary to ensure that the image size of 150 x 150 pixels which were used for the feature extraction part of the system would also be appropriate for SVMs and ANNs. The results of the testing showed that 150 x 150 pixel images provided the correct amount of information, and allowed the processing to be completed in real time, and therefore were appropriate.

Furthermore, a good feature extraction algorithm not only leads to better computer performance, but also improves the machine learning abilities. Choosing a feature

extraction algorithm is therefore a key factor in automatic object detection and many researchers are working on this area. Edge detection and gradient filters can enhance features, but they do not help with reducing the number of feature dimensions. Image histogram analysis is an effective feature extraction algorithm and this technique does not adversely suffer if an image is rotated. The following figure (Figure 6.7) shows that the histograms are almost the same for the image in the figure and its rotated image. These images are feature enhanced by Sobel filtering. However, image histogram analysis is a not an accurate method for comparing object similarity, and is therefore not useful for this research.

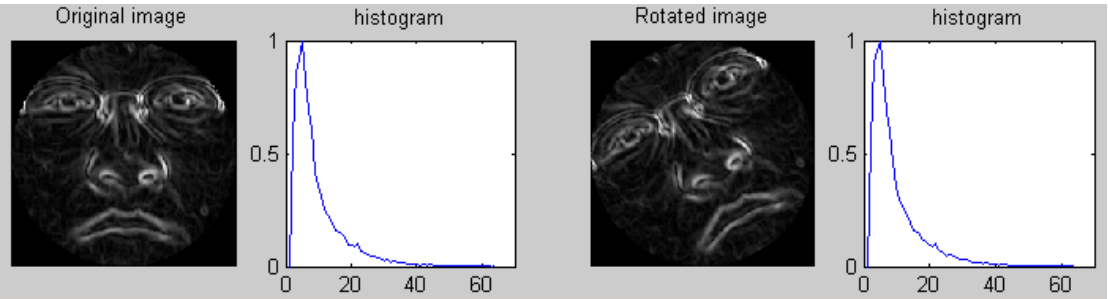


Figure 4.2.6.2.1 Comparison of an image's histogram before and after rotation

The idea naturally derived from optical characters recognition is that the simplest and most reasonable method is to separate a large image into regions of interest (ROI). The classification of the entire image can be done by classifying these small pieces. This method works for some tasks. Unfortunately it also increases the number of classifiers required, so it also demands more training time. Moreover, this method requires

transition rules between these classifiers and this method does not work when the small pieces are highly related. Results from this research indicated that this method is not suitable for training classifiers and does not work to an acceptable accuracy for very large sized images. The steps used in this process are shown in Figure 6.8. The face region is divided to three pieces and each region is associated with a classifier and output result of each classifier is fed into a final classifier. This final classifier makes the ultimate decision as to which facial expression is shown. However, a major problem is that for facial expression analysis, the height of each region is not always the same as it depends on the shape of each person's face. This requires the image size in each region to be adjusted before being input to the classifier. This operation degrades and distorts the image itself as well as the facial features within the image and means the accuracy is adversely effected.

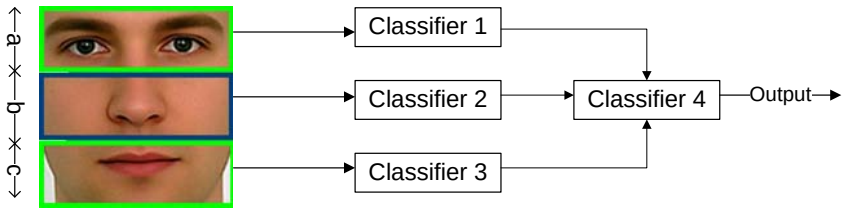


Figure 4.2.6.2.2 Multi-Classifier for Facial Feature Expression Analysis

There are also many effective feature extraction algorithms used in optical characters recognition (OCR). Most OCR systems operate on black and white images, which mean that they require perfect image processing before the characters can be recognized. One feature extraction algorithm used in optical character recognition is the character outline extraction algorithm. The outline extraction algorithm works perfectly for

simple characters like the English alphabet or Arabic numerals. However, it cannot classify characters in other languages which have a similar outline. This research showed a success rate of only approximately 60% on approximately 3000 Chinese characters. Figure 6.9 shows examples of ambiguous character outlines. In the following example it will get a similar outline features for two different Chinese characters.

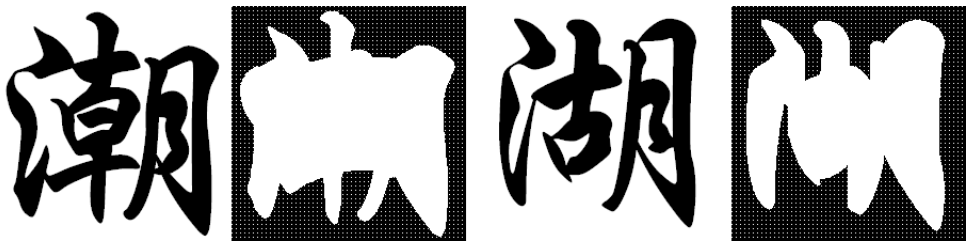
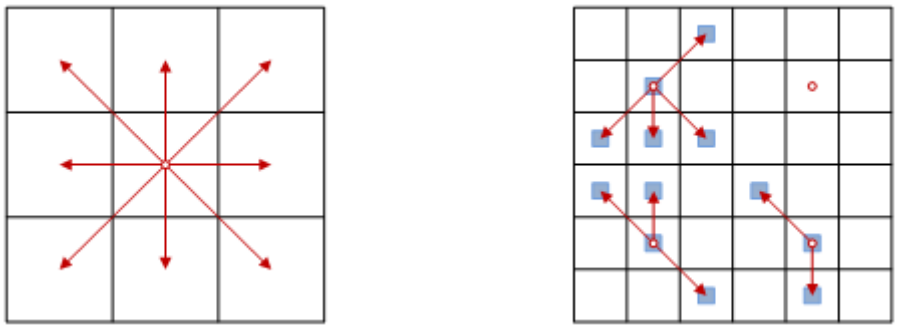


Figure 4.2.6.2.3 Outline extractions for Chinese characters

This problem was also illustrated when the algorithm was applied to facial expression analysis as it is also difficult to generate edge images from a face image that represents facial features.

Another feature extraction algorithm used in optical character recognition is pixel density feature analysis and pixel connectivity analysis. The pixel density feature analysis algorithm sums the value of each pixel and its neighbourhood pixels, and sets the sum as a feature. The algorithm calculates how many connections there are between each pixel and its neighbours, which are normally four or eight way

connections. Both the outline extraction and pixel density analysis feature extraction algorithms halve the image size and produce very similar results. The following figure (Figure 6.10) shows an example of eight-direction connections for an image.



(a) Eight-directions (b) Example of connections test

Figure 4.2.6.2.4 Eight-directions connections and an example

As stated above, the outline extraction algorithm resulted in an accuracy of about 60% on 3000 Chinese characters. The pixel density feature analysis and pixel connectivity analysis have an accuracy of approximately 80%. When these two techniques (outline feature extraction algorithm and pixel density feature analysis) are combined, the results have an accuracy of about 92% on the same characters. This accuracy is high enough to mean that these methods can be combined and applied successfully to very 'simple' facial features extraction tasks, for example cartoon faces or heavily made up (and therefore well defined) faces. These methods also require that the images are set in a good, consistent lighting environment.

Principal component analysis (PCA) is a way to reduce feature dimensions. PCA is a useful statistical technique in face recognition and image compression. However, this method does not allow the recovery of the original image and facial features extracted using PCA are not able to be enhanced. It is therefore not suitable for use in this research.

Geometry moments are a way to measure object features. Hu's (1962) simplified moments contain useful properties. Frits Zernike (1934), a Dutch mathematician and Nobel Prize winner in 1953, introduced complex moments. Since then a significant number of research and experiments (including experiments in the field of object detection and recognition) have been done to investigate the capabilities of Hu's moments and Zernike complex moments. These two techniques attempt to represent features regardless of rotation and scale. Although these techniques can successfully reduce the number of image dimensions, testing has confirmed that when dealing with large sized images, they are produce unclear results. Moreover they computationaly complex and are therefore not useful for achieving the stated goals.

This research focused on designing and tested a novel feature extraction algorithm for facial expression analysis which reduced or eliminated some of the limitations of the

existing processes described above. The algorithm combines the radon Transform and FFT. The radon transforms works by projecting every pixel's intensity value to a specific angle group. Normally the angle group is ranged from 0 degrees to 179 degrees, (with image projection for degrees from 180 to 359 being symmetrical to projection from 0 to 179 degrees). Figure 6.11 shows the radon projection for 45 and 135 degrees.

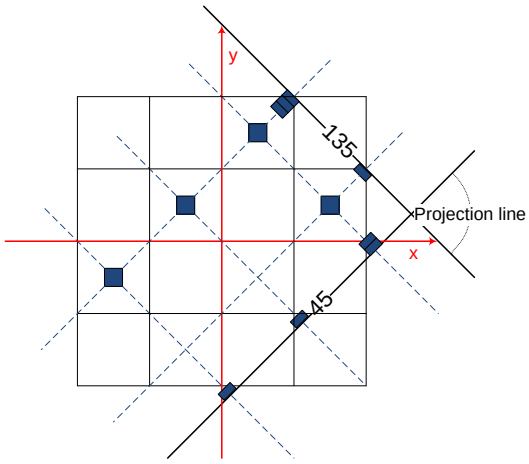


Figure 4.2.6.2.5 The radon projection for 45 and 135 degrees

The radon transform can successfully detect objects created by clear lines, because the line projection will be concentrated in a certain angle. In the radon projection map, every pixel will create a half sine signal when projected within 0 to 179 degrees. The projection of the entire image will therefore be the accumulation all these sine signals. Figure 6.12 shows the radon projection of the images of extract facial feature.

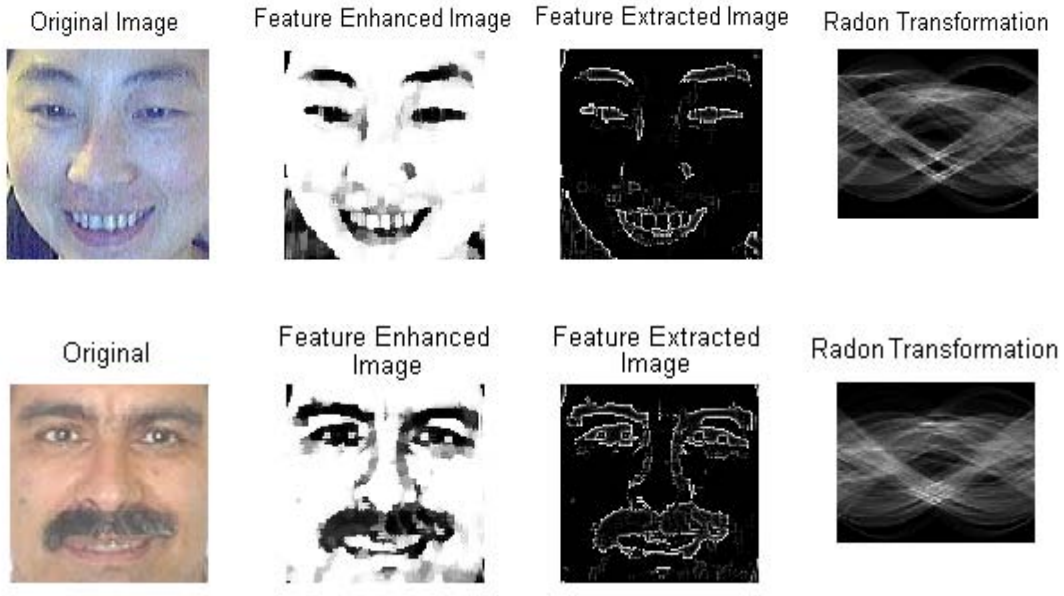


Figure 4.2.6.2.6 Feature Extraction and Radon Transformation

FFT is a convenient way to represent these sine signals which reduces the number of dimensions of image features without losing important information. FFT breaks down features and separates the real parts and the imaginary parts. It is important to undertake this separation as the real parts and imaginary parts double the size of important features. More simplified features can be extracted using a discrete cosine transform (DCT) which is closely related to the discrete Fourier transform. The formula for DCT is described below, where M and N are number of the rows and columns of image I. The DCT gives real output when applied it to an image.

$$O_{pq} = \alpha_p \alpha_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} I_{mn} \cos \frac{\pi(2m+1)p}{2M} \cos \frac{\pi(2n+1)q}{2N}$$

$$0 \le p \le M-1 \ , 0 \le q \le N-1,$$

$$\alpha_p = \begin{cases} 1/\sqrt{M}, p=0 \\ \sqrt{2/M}, 1 \le p \le M-1 \end{cases}, \text{ and } \alpha_q = \begin{cases} 1/\sqrt{N}, p=0 \\ \sqrt{2/N}, 1 \le q \le N-1 \end{cases}$$

The DCT is widely used in JPEG image compression. This algorithm can compress an image at a very high ratio - a 24 bit per pixel color bitmap image compressed by DCT in JPEG format reduces by approximately 90%. It is possible to apply DCT to an image of extracted facial features directly without applying radon transformation first, but this means that it is not possible to trace important facial features. Figure 6.13 shows a sample of experimental results of applying discrete cosine transform to facial feature extracted images directly.

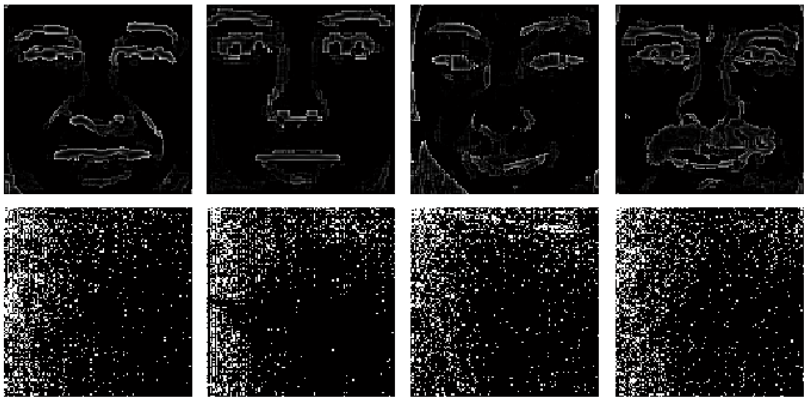


Figure 4.2.6.2.7 Images of extracted feature and their discrete cosine transform

These results show that important phases have been marked with white dots which denote important facial features. Those phases which are ignored (and are unimportant) are marked with black dots. The result shows that while the discrete cosine transform reduces image dimensions significantly, it also renders them indecipherable for the classifier (in the next stage of the process). Furthermore the locations of the white dots are disordered and confirm that this technique is not able

to locate and collect useful information. However, this major limitation of DCT can be removed by firstly applying radon transformation to the image. DCT is able to process the sine or cosine signals produced by radon transformation, which means that the facial features can now be detected and sorted. Moreover, they can run in real time on a normal PC.

Figure 6.14 shows sample images from experimental results showing the result of applying the radon transformation followed by DCT on the same images as in the previous figure (Figure 6.13).

Figure 4.2.6.2.8 Feature extracted image and its discrete cosine transform

It is clear that these important phases which represent facial features will congregate at the upper left corner according to the DCT process. These coefficients can be easily collected and selected. In this research, for each image of size 150 x 150 pixels, 1221 DCT coefficient features were used for the SVM input, instead of the 22,500 pixels which would be used as input for an unprocessed image. With this significant reduction in the number of feature dimensions, the performance of SVMs improved dramatically

as shown in the following figure (Figure 6.15).

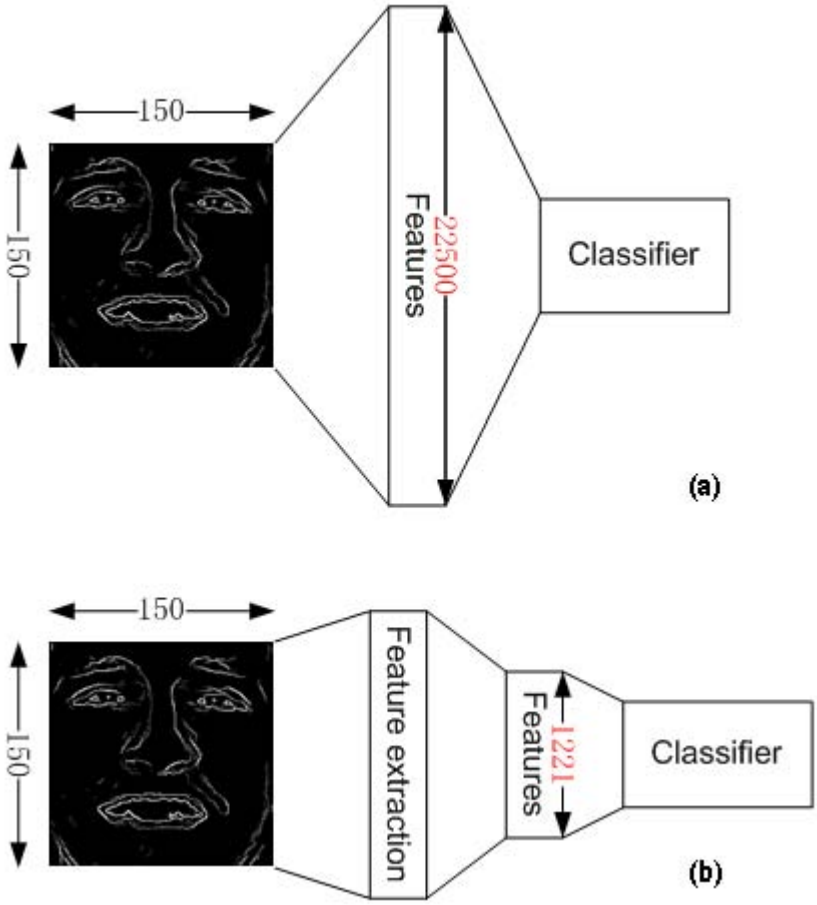


Figure 4.2.6.2.9 Features dimensions reduced using the feature extraction novel algorithm

(a) 22500 features, (b) 1221 features

The AAM (Cootes et al. 1995; Cootes et al. 1998; Cootes, Taylor 2001) method can accurately mark facial features and also uses significantly less points to represent these facial features. This means that training of either SVM or ANN system is easier. But, it also means that there is less information to use to detect some expressions, which can results in accuracy dropping below acceptable levels. For example, when people are angry or sad, the corners of the mouth go down, and also when people are sad or

angry, the brow region becomes furrowed. However, this does not mean the expressions are the same and the system needs to detect the differences in these expressions. Testing was undertaken to compare the results of analysis using AAM and the novel technique developed in this research. The results showed that the AAM (Ahlberg 2001; Taylor 2001; Ahlberg 2002.) technique can fail to detect and recognise the mouth features (shown in Figure 6.17(a)), but the novel method correctly detects that mouth corners which are down (Figure 6.16, b), (and that therefore this is an image of a sad face).

(a) AAM (b) My method

Figure 4.2.6.2.10 Comparison of the novel algorithm (b) with AAM (a)

It is clear that although AAM does reduce the number of features used to represent image information, it does so at the expense of detail and therefore accuracy, and means that the accuracy rate is too low to be deployed in a facial expression recognition system. This limitation is not shared by the novel method developed during this research which not only accurately represents facial feature but also effectively

and significantly reduces the number of feature dimensions.

6.3. Classification of facial expressions - ANNs and SVMs

6.3.1. Summary

The selection of a good classifier is crucial to achieve the stated goals as achieving the best classification performance and required accuracy can be a time consuming process. This research included experiments both on SVMs and ANNs. Numerous tests were carried out - not only in applying different image processing algorithms, but also on different machine learning strategies. The results of these tests have shown that despite ANNs being a natural way of simulating the structure of human brain, the learning ability and the confidence interval in statistics are not comparable with SVMs. This, combined with the more difficult training requirements of ANNs, meant that, for this research, SVMs were considered more appropriate, and focus was given to training an SVM for use within the system. The background on the capabilities of ANNs and SVMs and the process which led to the selection of the SVM is described in detail in the following sections.

6.3.2. ANNs Overview

Based on the study and simulation of the human brain and neural structure, ANNs have been designed and subsequently tested with a goal of replicating the strong learning

ability of the human brain. These tests have confirmed that ANNs are able to learn – although still only at an elementary and basic level. This reflects the fact that the level of understanding in today's science and technology of the human brain structure and how information is transmitted between neurons is still at an early stage.

While ANNs are immature when it comes to the electronic simulation of human brain, their limited ability has already generated great influence in the field of machine learning. The basic structure of an ANNs is shown in the following figure (Figure 6.17).

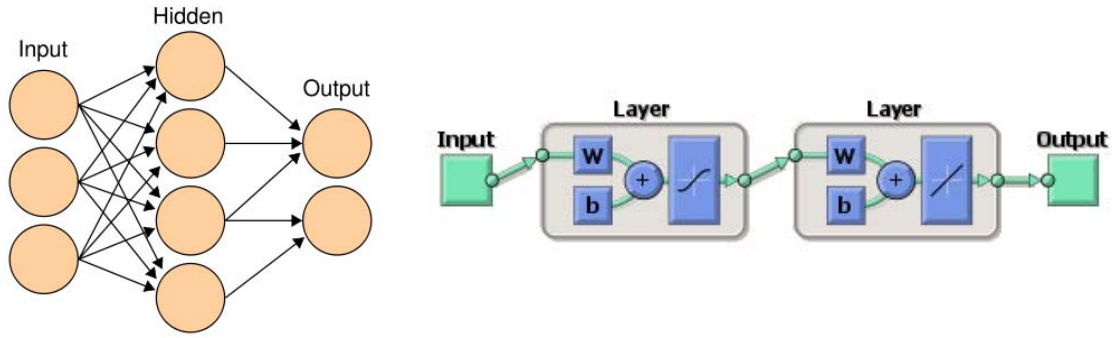


Figure 4.2.6.2.1 Structure of an ANNs

The back propagation learning algorithm is a supervised learning technique often used in ANNs. Back propagation was first described by Werbos in 1974, and further developed by Rumelhart and McClelland (1986).

Back propagation requires the transformation function to be differentiable. Similar to the perceptron learning algorithm, back-propagation attempts to reduce errors, which are the differences between the actual output and the desired result. The errors have

to be propagated back to the hidden neurons. In some circumstances, and after some iteration, an ANNs learns from the input pattern by calculating the error of each neuron's output and adjusting its weight value to match the desired output. Such a learning loop ends when the ANN gives a correct output or reaches a maximum number of iterations. The back propagation algorithm can be described using the pseudo code presented in Figure 6.18.

```
Randomly initialize weights of network
Repeat
    For each pattern P in the training set
        Calculate actually output of network (forward pass).
        Compare actually output with desired output.
        Backward calculate weight update value from hidden layers to input layer.
        Update the weights of the network
    End
Until all training patterns classified correctly or stopping when maximum iteration meet.
```

Figure 4.2.6.2.2 The Back Propagation Algorithm

In this research, the first step was to create a 4 layer feed-forward back-propagation ANN (shown in Figure 6.19) and facial features are set as the input of the ANN. The last layer corresponds to the output layer. The Levenberg-Marquardt back-propagation algorithm (Fletcher 1971) is applied. The tansig transfer function is selected (shown in Figure 6.20).

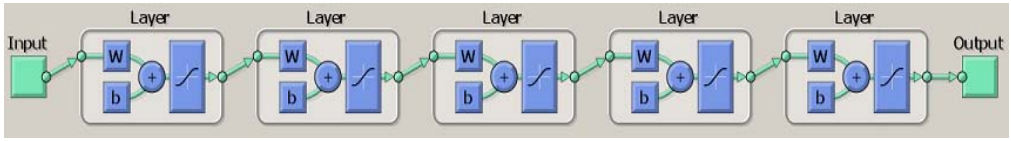
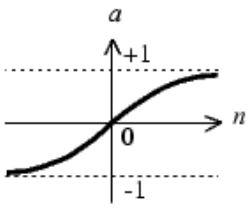


Figure 4.2.6.2.3 the structure of ANN for training facial features



$$a = tansig(n) = \frac{2}{(1+e^{-2n})^{-1}}$$

Figure 4.2.6.2.4 Tansig transfer function

Figure 6.21 shows the ANN performance at gradient of 1000 epoch. Figure 6.22 shows that the gradient is a falling line which means that the rate of learning drops significantly over time. Furthermore, the ANN structure also requires a large amount of memory space. Regression on an ANN system after 178,073 iterations is shown in Figure 6.23, indicating that at this point, the accuracy is reduced below acceptable levels.

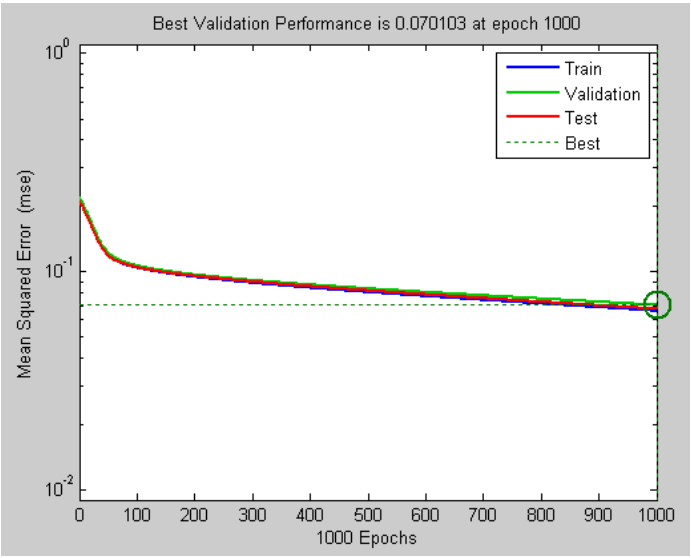


Figure 4.2.6.2.5 performance and training state for ANN

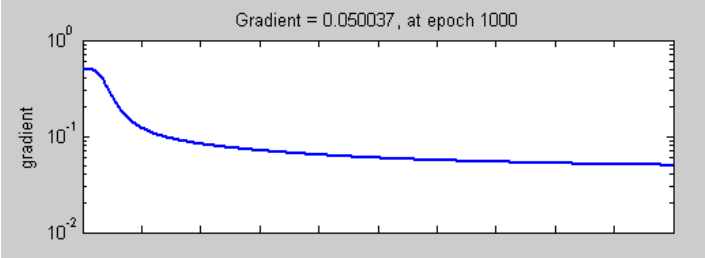


Figure 4.2.6.2.6 Gradient for training ANN

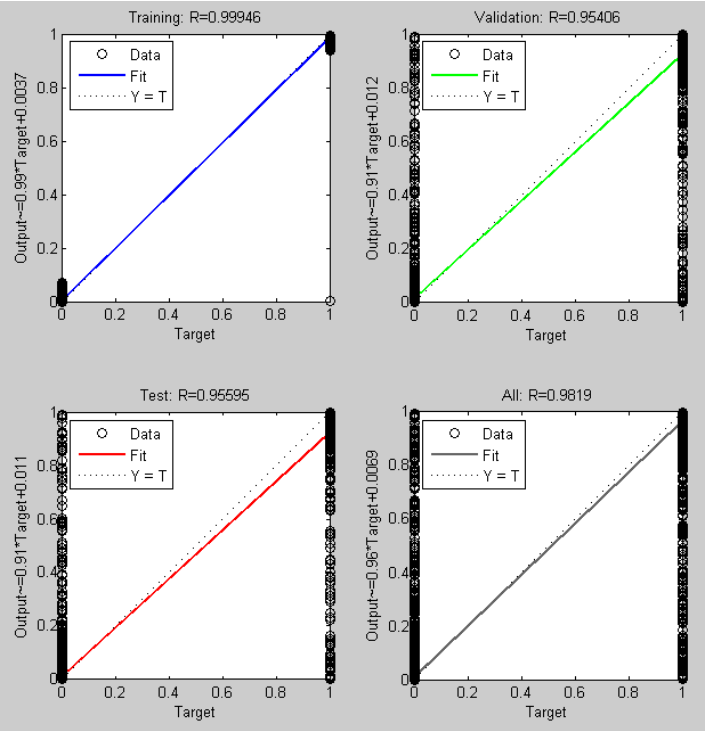


Figure 4.2.6.2.7 Regression on ANN (after 178073 iterations)

The disadvantages of the ANN's back propagation learning algorithm can be briefly summarised as follows. ANNs may converge on local minimum value, the training process for ANNs is very time consuming because the learning process takes some time to converge and the transformation function has to be differentiable. Furthermore, ANNs are attempting to simulate the regression ability of the human brain which is a process that researchers do not yet fully understand, meaning that its implementation into a computer system is even more difficult. Finally, ANNs are limited by their current structures which may be improved through future research.

6.3.3. SVMs Overview

In 1974, Vapnik (1974) introduced a statistical learning theory and based on this, the initial concept of SVMs was introduced by Vapnik, Steven, and Smola (1996). In 1984, Haussler (1990) introduced probably approximately correct learning (PAC). Based on the PAC learning algorithm Vapnik further developed the Structural Risk Minimization Method (1979) and refined the concept of SVMs (Vapnik 1995). Recent research shows that, with the addition of the kernel method, in 1990s (Haussler 1990), SVMs have a strong learning ability (Burges 1997; Osuna, Freund, Girosi 1997a; Osuna, Freund, Girosi 1997b; Schölkopf 1997; Cristianini, Shawe-Taylor 2000; Schölkopf et al. 2000; Schölkopf et al. 2001).

In the training algorithm used for SVMs, for the two class linear separable classification problem, let x denote pattern for training, and y denote the desired output. The training pattern can them be written as:

$$\text{Training set} \{x_i, y_i\}, i = 1, ..., l, \quad y_i \in \{-1, 1\}, \quad x_i \in R^d .$$

Suppose that there exists a hyper-plane, separating two classes. For all x on that plane, it has to satisfy the equation $w \cdot x + b = 0$. In SVM theorem, the best way to separate two classes is to find the maximum margin between the support vectors (marked in red color) and the two parallel hyper-planes as shown in Figure 6.25.

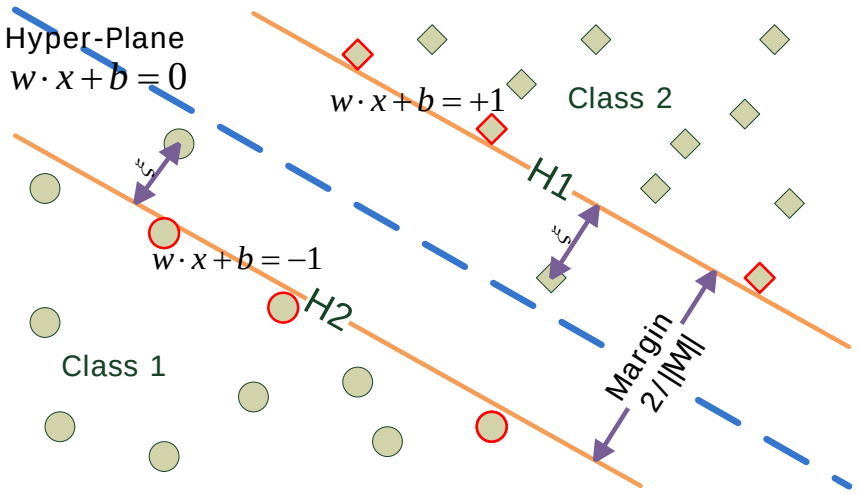


Figure 6.3.3.1 Separating two classes by hyper-planes

Class 1 satisfies $w \cdot x + b \geq +1$, and all the points on hyper-plane H1 satisfy $w \cdot x + b = +1$. Class 2 satisfies $w \cdot x + b \leq -1$, and all the points on hyper-plane H2 satisfy $w \cdot x + b = -1$. Combining Class 1 and Class 2 we obtain the inequality $y_i(w \cdot x_i + b) - 1 \geq 0$. The distance between the two hyper-planes is $2/||w||$. So finding the maximum margin is equivalent to minimizing $||w||^2$ subject to constraints $y_i(w \cdot x_i + b) - 1 \geq 0$. Then, applying Lagrange multipliers, we get:

$$L \equiv \frac{1}{2}\|w\|^2 - \sum_{i=1}^{l} \alpha_i \left[y_i (x_i \bullet w + b) - 1 \right] \qquad (1)$$

Next we take the derivative of L with respect to w, and obtain:

$$w = \sum_i \alpha_i y_i x_i \qquad (2)$$

Take the derivative of L with respect to b, and then get:

$$\sum_i \alpha_i y_i = 0 \qquad (3)$$

Substitute (2) and (3) into (1), and get:

$$L = \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j (x_i \bullet x_j) \qquad (4)$$

Therefore L is maximized with respect to α_i, subject to constraints $\sum_i \alpha_i y_i = 0$ and positivity α_i. Those vectors with $\alpha_i > 0$ are called support vectors, and they are on hyper-plane H1 or H2.

It is then easy to apply formula (4) to the two class nonlinear separable classification problem. This is because it is possible to map nonlinear data to another Euclidean space, which separates these classes by a linear separable formula inside that Euclidean space. In formula (4), only $(x_i \bullet x_j)$ which is the dot product is involved. The mapping function can be written as $\phi : R^d \mapsto \mathrm{H}$. And the dot product can be written as $K(x_i, x_j) = \phi(x_i) \bullet \phi(x_j)$,and K is the kernel function. A great advantage of kernel functions is that only the kernel function is needed when training the SVMs – it is not necessary to explore or study about the mapping function ϕ, which makes the training significantly easier and quicker. There are several kernel functions which were found during this research and these are:

Polynomial function: $K(x, y) = (x \bullet y + 1)^p$

Gaussian radial basis function (RBF): $K(x, y) = e^{-\|x-y\|^2/2\sigma^2}$

And sigmoid function: $K(x, y) = tanh(kx \bullet y - \delta)$

Maximizing the margin with noise was introduced by Vapnik (1999). To find the maximum margin the following equation is utilized, where C is the trade off parameter between error and margin, and ξ is error:

$$\frac{1}{2}\|w\|^2 + C\sum_{k=1}^{l}\xi_k \qquad \text{Subject to } y_i(w^T\phi(x_i)+b \geq 1-\xi_i,\ \ \xi > 0$$

This then becomes a quadratic problem and the maximum margin is:

$$\sum_i \alpha_i - \frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j (x_i \bullet x_j) \text{ Subject to } \sum_i \alpha_i y_i = 0 \quad (5)$$

Where, $w = \sum_i \alpha_i y_i x_i$ and $b = y_k(1-\xi_k) - x_k \bullet w_k$.

And then, the classification function can be written as:

$$f(x, w, b) = sign(w \bullet x - b)$$

The training of the SVMs is the process of finding the support vectors by solving formula (5). Because there is only dot products involved the entire training procedure is very fast. Also, in real world pattern recognition, a two class problem can be used to solve a multi-class problem by considering one class against all the rest of the classes – this can briefly be described as a “one against all” technique.

6.4. Training of the SVM

In this research, the training of the SVM is based on the theory of SVM. However, some areas of theory, for example, how SVMs map data to the hyper-plane, and what kernel functions give the best result, are still being researched. This means that in this

experiment it was necessary to test a variety of different kernels to determine which provided the best result. Similarly testing was necessary for the multi-class classification problem, “One class Against All” and “One class Against One” are both tested and compared(Allwein, Schapire, Singer 2000; Hsu, Lin 2002; Rifkin, Kalautau 2004).

6.4.1. “One against All” and “One against One” classification:

To reduce training complexity, it was necessary to start from the assumption that all input features are obtained from images with normalized face regions. This means that the facial expression analysis classifier used in this research only works on images with a normalized face region. The face region images are located by the face detection algorithm introduced in Chapter 4, and the facial expressions are separated to six classes. They are, natural (class ID=0), happy (class ID=1), laugh (class ID=2), sad (class ID=3), angry (class ID=4), surprise (class ID=5), disgust (class ID=6).

There are two important strategies for multi-class SVMs. The first is “One against All” and the other is “One against One” strategy. In “One against All” strategy, it is necessary to create one SVM for each class, and consider all other classes as another class. The classification of “One against All” is based on maximum output from all SVMs where the output with the highest value is considered correct. From the literature review alone, it is hard to conclude which strategy is better for classification, so testing

and comparison was required to determine which method produced the best results. The comparison results of these comparisons are shown in the following section.

In the “One against One” strategy, the SVM considers all possible pairs between every pair of classes. So for n-classes’ classification problem it requires $n \times (n-1)/2$ SVMs. And the ultimate classification decision using the “One against One” is based on the maximum voting sum from all SVMs outputs. Multiclass classification problems are still the subject of ongoing in research, and are therefore not completely reliable.

Testing was conducted, which confirmed that, while the accuracy and speed of both methods was comparable, the training time of the “One against One” technique was considerably longer. Also, as there is a need to undertake a multiclass classification process, it is also more complicated to implement. For these reasons, it was decided to use the “One against All” technique during this research.

6.4.2. Cross validation

Sometimes machine learning algorithms do not perform well with particular input data. In ANNs, this is called the over training problem and this issue also occurs in SVMs. It is caused when the training process converges on local optimal parameters but can be

avoided by choosing optimum parameters for the training. This is known as cross validation.

In the cross validation algorithm, training data which is already classified is collected in a random order. The training process randomly splits the data to several groups known as training sets and testing sets. As the name implies, the training sets are used for training, and the testing sets are used for predicting and calculating the accuracy. The total number of groups is known as the k-fold, and the process is generally known as k-fold cross validation. To ensure that not too much important training data is lost, only a small amount of data is used for testing purposes during each training process.

For training the facial expression analysis system, facial feature data was collected and labelled with the correct class ID. According to k-fold cross validation algorithm, a k equal to 5 was used, which means 5-fold cross validation, and the data was randomly sorted in 5 folds (or groups). In each training process, from these 5 folds, 4 folds were randomly chosen as the training set and 1 fold as testing set. The system was then trained and tested and the parameters were adjusted based on the latest set of results. Then the facial feature data was again grouped randomly into 5 folds, with 4 folds were randomly chosen as the training sets and 1 fold as testing set and a new set of training and testing was completed. This process continued with parameters being adjusted

after each set of testing until the optimum parameters were found.

To ensure that the best result is achieved, it was also important to collect and prepare a sufficient amount of data for the SVMs. About 4000 real images of different facial expressions were collected. However initial experiments showed that it was still not enough for this complex task and the results were not sufficiently accurate. To increase the amount of available data, approximately 8000 artificial facial expression images were generated using FaceGen 3.0, which is a face modelling tool. Examples of artificial face images are shown in Figure 6.25.

Figure 6.3.3.1 Artificial faces generated by FaceGen (natural expression)

There are three very important reasons for using these artificial expressions as training samples. The first reason is that these artificial facial expression images can be collected and classified easily. The second reason is that the artificial facial expression images are free from excessive noise, and variations in lighting which mean that the features are clearer. This means that the SVMs can learn from these important facial features rather than from noise. The final reason is that we can use regression theory

to separate the images and group the data in hyper planes.

6.5. Testing of kernel functions

Three important kernel functions, Polynomial kernel function, Gaussian radial basis function ("RBF"), and Sigmoid kernel (Burges 1997; Cristianini, Shawe-Taylor 2000) were tested using an SVM system. The results of applying different kernel models to our image database are presented in Table 6.2. From these results it is clear that the RBF kernel provided the most accurate results and was therefore chosen to be used in this research.

Table 6.2 Accuracy of different kernels.

	Linear Kernel Model	Polynomial Kernel Model	RBF Kernel Model
Normal	75%	89%	89%
Disgust	65%	75%	85%
Fear	68%	83%	86%
Smile	73%	87%	93%
Surprised	87%	93%	94%

The results of facial expression analysis testing undertaken with the novel technique developed in this research using the RBF kernel was also compared with result of Datcu and Rothkrantz's (2007) method. The results (shown in Table 6.3) show that the novel technique gives higher accuracy in recognising the facial expressions.

Table 6.3 Comparison of Datcu & Rothkrantz's method (2007) with the proposed method.

	Datcu & Rothkrantz		This work (RBF)
	Still image	*Video*	Average
Natural	*Not given*	*Not given*	89.03%
Surprise	*83.8%*	*88.67%*	93.68%
Sadness	*82.79%*	*85.86%*	92.81%
Anger	*75.86%*	*85.71%*	78.93%
Disgust	*80.35%*	*82.14%*	85.23%
Happy	*72.64%*	*79.62%*	83.78%

In this research, an exhaustive search is carried out to find the optimum parameters for use with the RBF kernel. Table 6.4 shows the results of the parameter searching using an RBF kernel.

Table 6.4 Exhausted SVM parameters searching

σ \ C	0.001	0.01	0.1	1	10	100	1000
1	96.354	96.354	83.397	60.432	40.457	20.159	12.724
10	99.714	96.354	87.397	61.437	40.457	20.159	12.724
100	99.714	99.714	90.724	62.571	45.021	20.159	12.724
1000	99.714	99.714	95.724	80.291	45.021	20.159	12.724
10000	99.714	99.714	95.724	90.291	50.76	20.159	12.724
100000	99.714	99.714	95.724	80.291	50.76	20.159	12.724

Observing the above table, the SVMs will reach boundary of the best parameters at C=0.01. The graph presented in Figure 6.26 also shows the SVM parameter searching in three dimensions. It is clear that choosing parameters within the best parameter region will get better result, in this thesis, C=0.001 and σ = 1000 were selected.

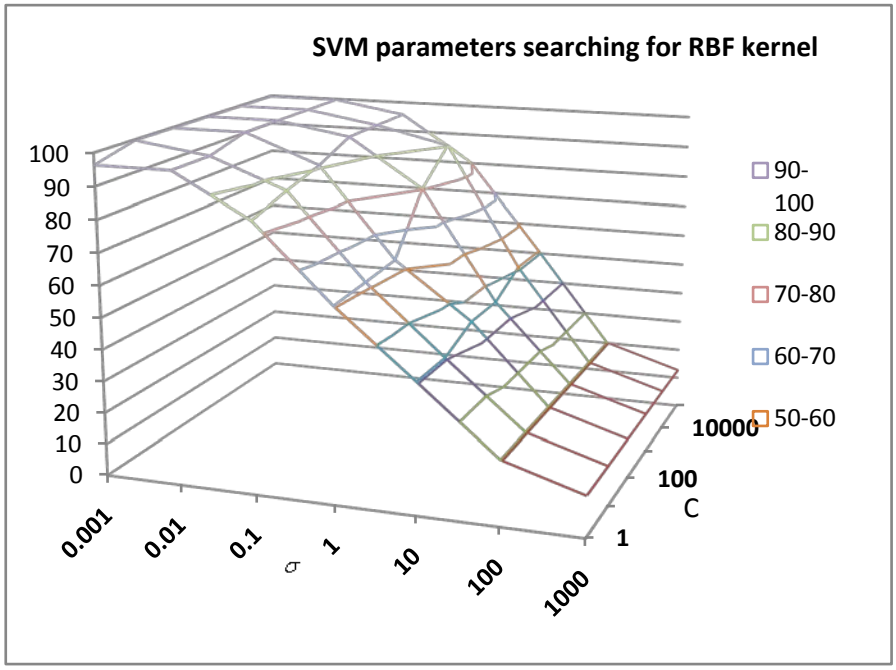


Figure 6.3.3.1 SVM parameter searching

6.6. Facial expression classification system - summary of findings

In summary, SVM gives more accurate results when compared with ANN systems. This is because:

- The solution to an SVM is global and unique, but ANN can be limited in local minima.
- ANNs are computationally more complex compared with SVMs.
- ANN is less prone to over-training when data is complex and ANN has difficulty distinguishing between image noise and important features of the image.
- SVM maps data to a hyper-plane by kernel functions, and separates them. This means that SVMs converge faster than ANN.

Also, this research completed a significant amount of comparison of different kernels, and the results shows that the RBF kernel gives the best result compared with other kernels for facial features classification.

6.7. Facial expression recognition – testing and results

6.7.1. Still images and video sequence analysis

The facial expression analysis algorithm developed in this research is able to analyse both still images and video sequences. The facial expression analysis of still images is relatively straightforward, but analysis of a video sequence is more difficult as it requires processing to be fast enough for real time processing.

The frame rate of Logitech camera used in this research is up to 30 frames per second. As stated previously, the face detection algorithm used in this research takes about 9 frames per second (compare with openCV face detection which takes about 12 frames per second). This means that the system developed is able to analyse facial expressions every 300 milliseconds as shown in Figure 6.27. The first row shows frame sequences and second shows the differences between the frames.

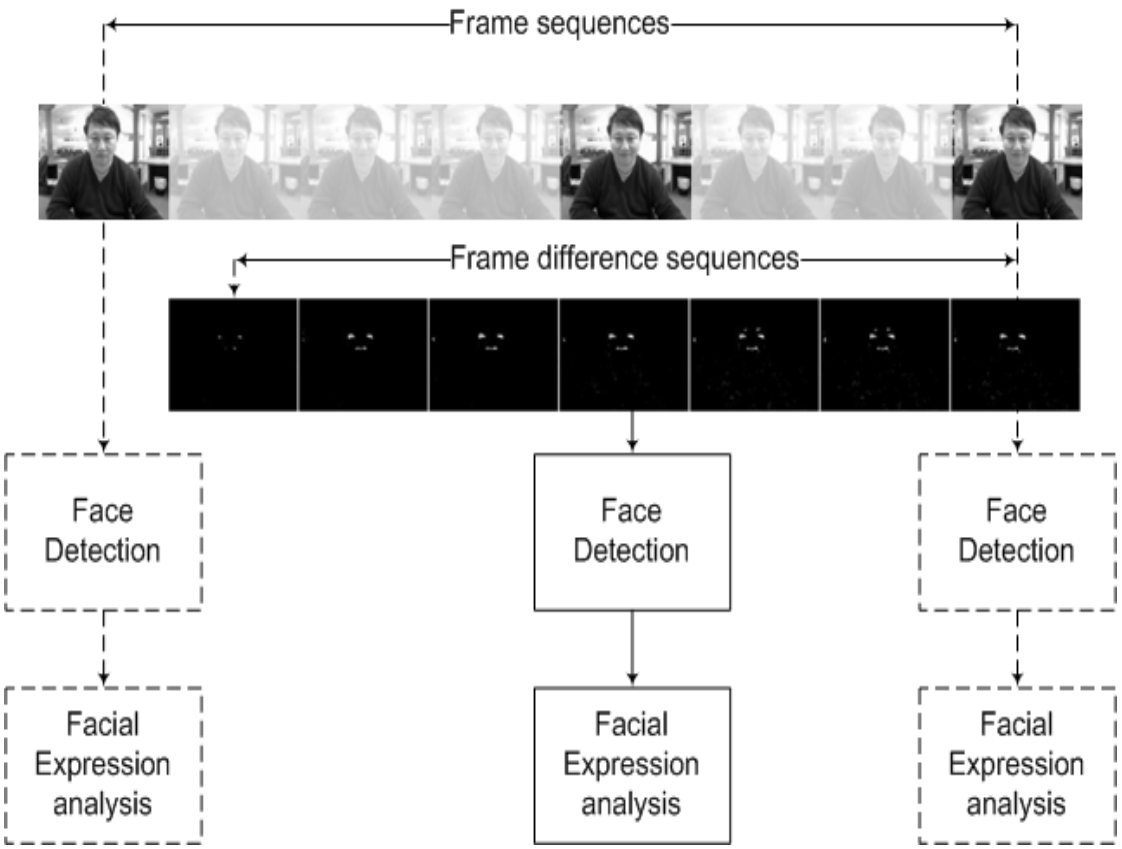
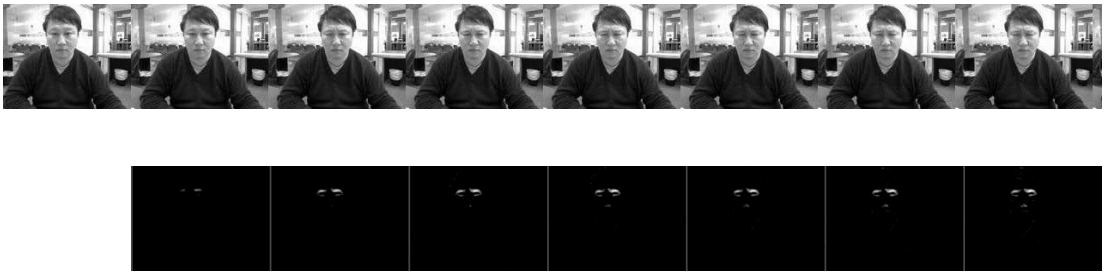


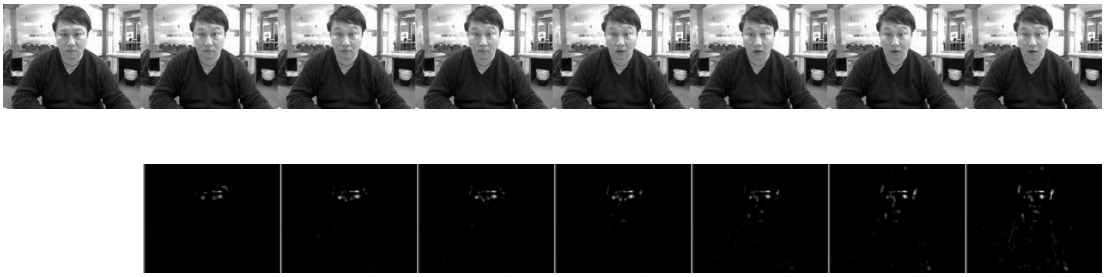
Figure 6.3.3.1 Video sequences analysis

In order to increase accuracy and ensure that the strongest emotions are detected over a period of one second (about 300 milliseconds per frame) the following process was used. Starting from the first frame, each frame is compared with the next using image subtraction with the difference in the face region measured by summing the pixel difference. This way, the most distinguished frame in each second (and therefore the one likely to have the strongest facial expression) is chosen for facial expression analysis. The sample of frame sequences difference is shown in Figure 6.28.

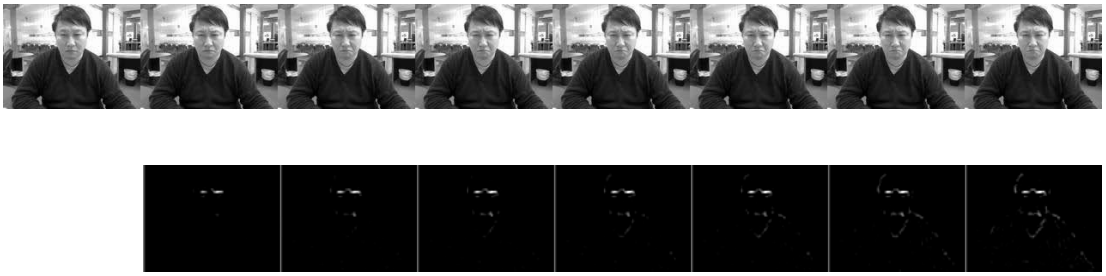
Anger



Surprise



Sad



Smile

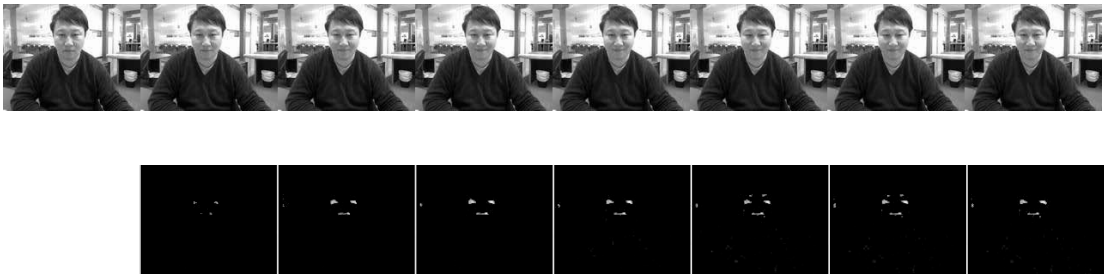


Figure 6.3.3.2 Frame difference on different facial expressions

6.7.2. Screen shots of real time facial expression analysis system

Figures 6.29 through to Figure 6.33 are captured from the real time facial expression analysis system in operation. For the sake of clarity, the results of face detection have been removed. The face region image processing results are marked in green. The output of the system is shown on the right of the screen, with the blue bars showing the system's certainty regarding each of the facial expressions. The maximum value corresponds to the system's correct detection.

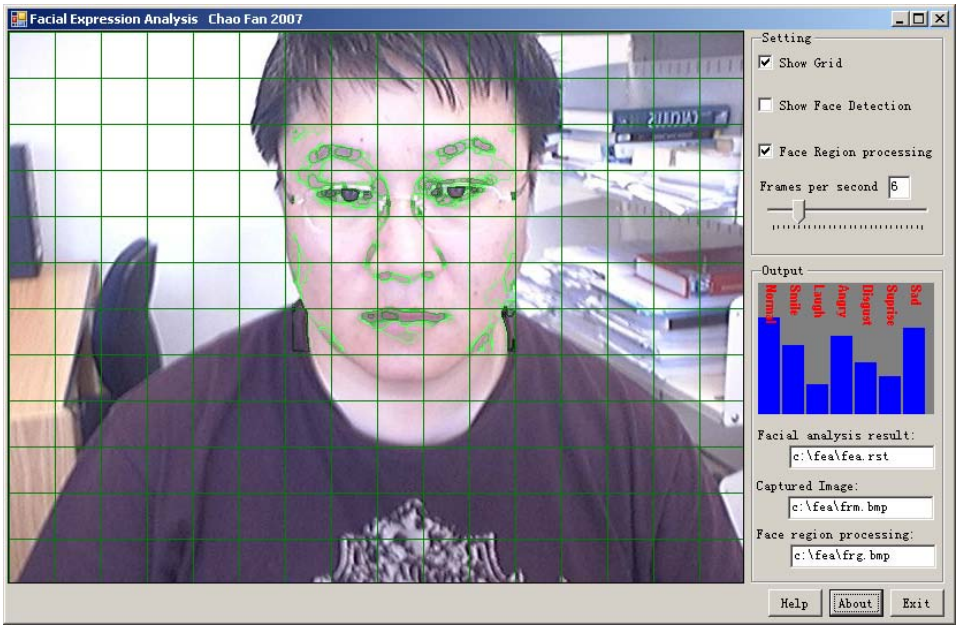


Figure 6.3.3.1 Natural expression detected

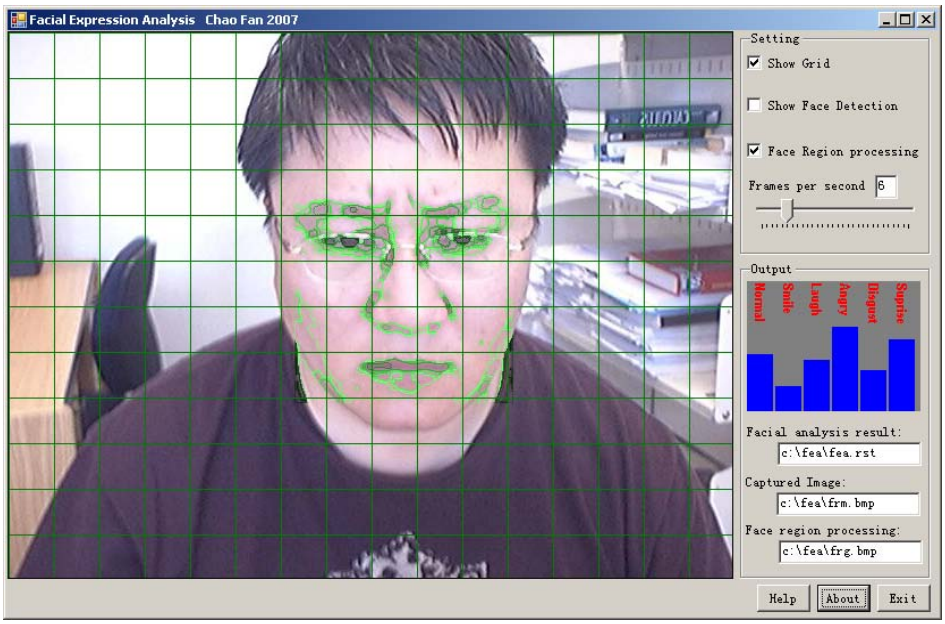


Figure 6.3.3.2 Angry expression detected

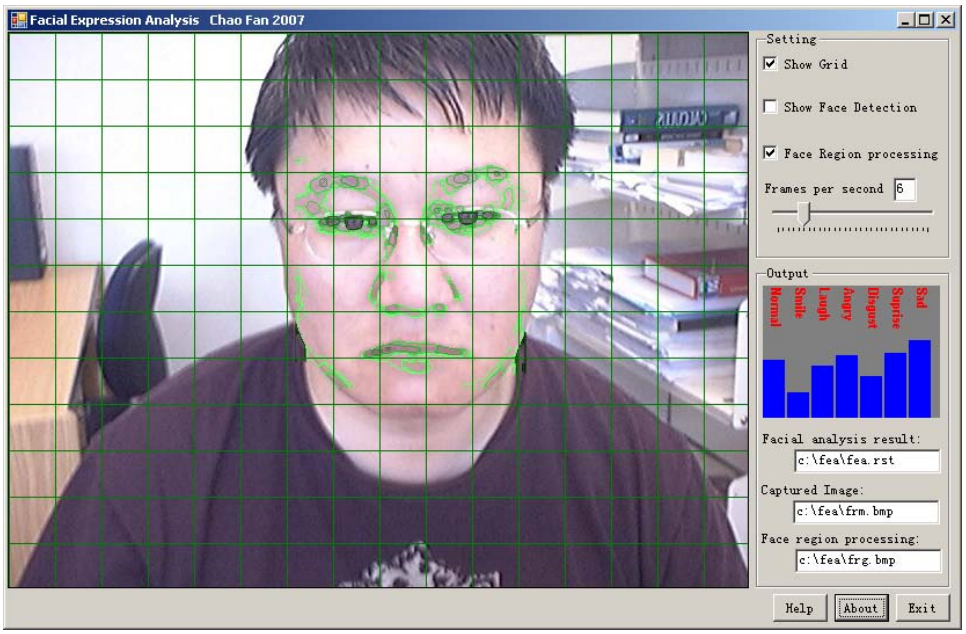


Figure 6.3.3.3 Sad expression detected

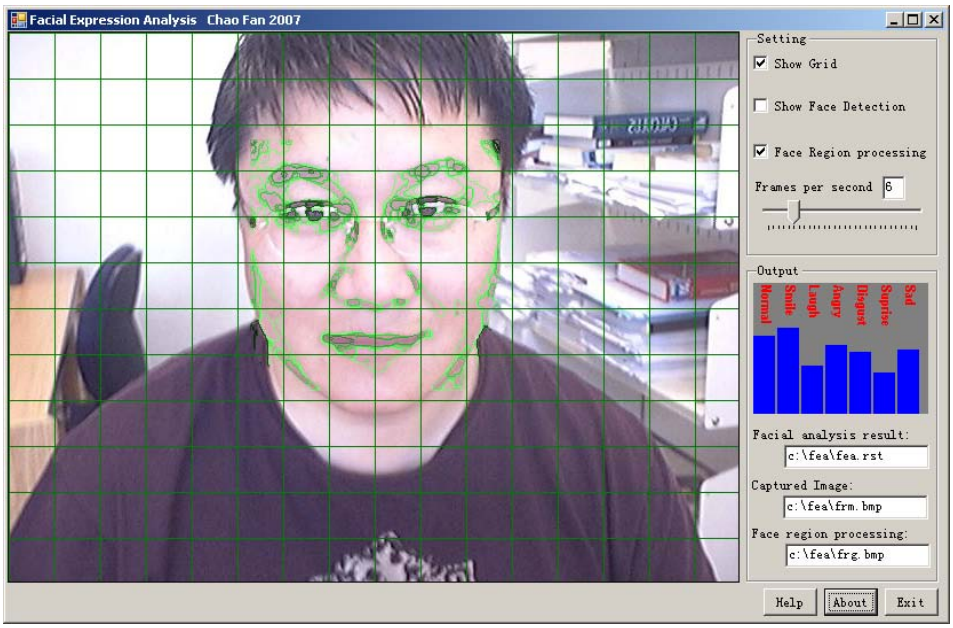


Figure 6.3.3.4 Smile expression detected

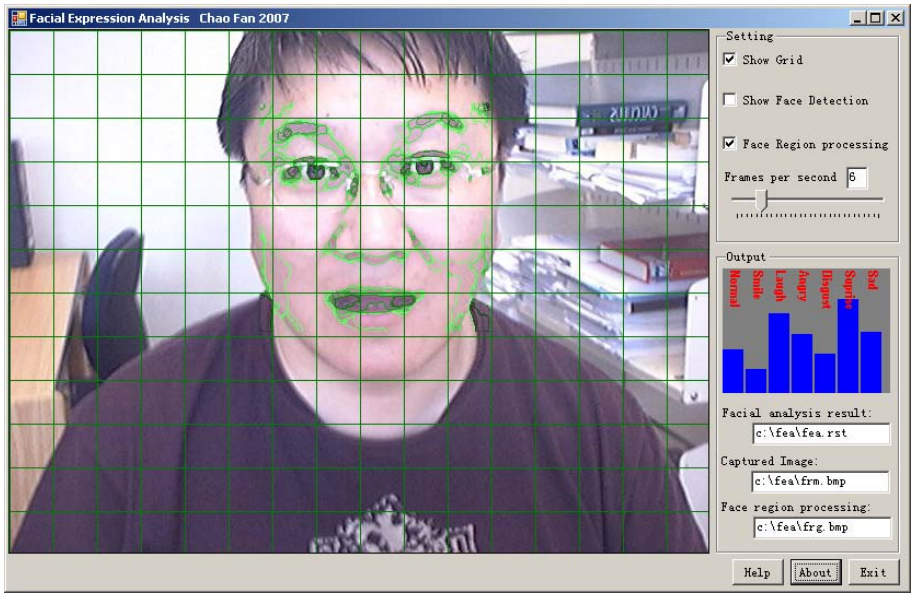


Figure 6.3.3.5 Surprised expression detected

6.7.3. Conclusion

In this research a novel method for facial features extraction has been presented and compared with other existing image processing methods. The technique presented

achieves the stated goals successfully and is suitable for real time applications as it reduces the number of facial dimensions so that the process can be completed by a standard PC, and still accurately detects and recognises facial expressions using an SVM classifier.

Chapter 7. Discussions and Conclusions

Overall, this research has achieved the stated goal of developing a system which can automatically detect and recognise human facial emotions in real time, and on a computer similar to those found in most homes in today's society. In addition, the system can provide accurate results despite the large variations in human faces due to age, gender and ethnicity, and also variations in lighting conditions.

Specifically, this research has introduced three key developments which have facilitated the creation of this system and added to the knowledge and techniques within the field of image processing. These three novel techniques are summarised below:

I) Real-time face detection, this algorithm is also suitable for detection of other objects by replacing the averaged face image with a sample image.

II) Facial feature enhancement and extraction.

III) Facial expression detection and analysis.

7.1. Real-time face detection

Firstly, it has been demonstrated that the essential task of quick and accurate face detection can be significantly improved with a novel method developed in this research called Optimized Candidates using Similarity Measurement (OCSM) which combines the phase congruency algorithm and similarity measurements using FFT. In relation to

face detection, this research has improved current techniques by focusing on reducing the classifier's computational requirements by more accurately determining which of the images do contain a face. An averaged face template is created from a large face database (including computer-generated faces if appropriate) and this is used as a basis for training an SVM. In addition, both the size and the orientation of the face template are varied within set parameters so that faces of a variety of sizes, and different orientations (which have been rotated any number of degrees) can be detected. The SVM training can be completed offline so does not add to the time or computational power required.

This means that this system is considerably more accurate than previous methods, as it compares possible face candidates with an actual face (in the form of the averaged face template generated from the database), and therefore the number of potential face candidates' indentified by the system and sent for further testing is significantly reduced. For example, in the Viola and Jones method (described in Chapter 2) potential faces candidates are compared only with a grey scale pattern which is considered to mimic the grey scale output of a human face. This leads to many face candidates which actually do not contain a face being sent to the second stage of processing, which results in the requirement for a considerable amount of time and computational power at this stage. Furthermore, as described in Chapter 4, neither Viola and Jones (2001), nor any other existing face detection methods reviewed during this research, could

detect, with acceptable accuracy, faces rotated beyond 10 degrees in either directions. This research has developed a technique which can accurately detect faces rotated to any degree, including completely upside down. Importantly, this is able to be completed in real time on a standard PC. This is a significant breakthrough as the technique can be easily modified to work with objects other than faces, meaning that it has the potential to be used in a wide range of applications.

7.2. Facial feature enhancement and extraction

The second major area of improvement introduced by this thesis is in relation to facial features enhancement and extraction which is achieved by combining a novel algorithm using normalized face images and the existing CLAHE technique.

This task is made even more difficult when a system is required to function accurately in a variety of lighting conditions and with a wider range of ethnicities – as is the case with the NGITS (Sarrafzadeh et al. 2004). A number of algorithms used in this area were reviewed and tested, including algorithms for image noise removal, image smoothing, and feature enhancement. Some of the techniques are suitable for large sized images, some achieve good results when certain parameters can be adjusted manually for each image, and other techniques can be considered successful when the amount of computational power (and time taken) is not an important factor. However, all these

existing methods have limitations, which meant that they were not suitable for a real time facial expression analysis system. Therefore this research has developed a facial feature enhancement and extraction system which overcomes these difficulties, and can accurately identify and enhance facial features in real time and over a variety of skin colors and lighting conditions using a standard PC.

As described in chapter 5, the system combines a number of techniques. A sigma filter carried out image smoothing and noise removal, and a morphing logic operation is used to estimate the main mass of the face region in order to make the adjustments required due to variations in lighting or skin color. Features are enhanced using a CLAHE (Zuiderveld 1994) algorithm, and an edge prevent smoothing algorithm is used to separate facial features from the skin of the face. Also as part of this stage, a key bottleneck in existing methods has been significantly reduced, as described in Chapter 6. This bottleneck is the large number of feature dimensions which are created during the feature extraction and enhancement process and require an excessive amount of computational power to process. Using a combination of radon transformation and DCT this research has introduced a novel and efficient method for reducing the number of feature dimensions to a degree that the required calculations can now be undertaken in real time, with no significant loss in facial features – and using a standard PC.

7.3. Facial expression detection and analysis

Finally this research included a detailed review of a variety of existing facial expression detection and analysis systems, focusing in particular on learning algorithms using SVMs and ANNs. This review concluded that the accuracy of both systems decreases significantly as the number of input feature dimensions (which is based on the complexity and the size of the image) increases.

Results of testing confirmed that the facial expression and analysis technique developed in this research using the reduced number of extracted feature dimensions and “One against All” training technique using the RBF kernel with an SVM produced more accurate results than any other technique reviewed regardless of variations in the subjects’ age, gender or ethnicity, or differences in lighting conditions.

This study has shown that machine learning abilities can be improved in two main ways. The first is by optimising feature extraction methods, which was the main research direction in this thesis, and the second is by improving the classification methods (ANN or SVM) which employ the extracted features.

7.4. Suggestions for future research

In this dissertation, two different, but highly related, machine vision and machine learning techniques (face detection system and facial expression analysis) are discussed. In general, face detection rotation invariance and scale invariance are important factors for future consideration, and are likely to require continued focus and research for the foreseeable future. In facial expression analysis, more detailed image property analysis is required, as more face detailed feature enhancement algorithms are likely to be more conducive for accurate facial expression analysis.

There are also some more specific research areas deserving of additional focus. Further research on face detection using SIFT method is also likely to be beneficial. Lowe (1999) presented a method called 'SIFT' for object matching in 1999, and updated it in 2004 (Lowe 2004) which has well defined properties and works very well for certain objects from certain viewpoints. However, it does not work well when dealing with different objects. This is because, in Lowe's method, the first (and most crucial step) is to find out key-points, with similarity and making comparisons based on these key-points (known as a "point similarity measurement"). Unfortunately, in face detection such key-points do not exist, which means that "point similarity measurement" will not work. Despite this many researchers have tried to increase the tolerance of the SIFT algorithm to employ it for face detection. To improve Lowe's algorithm, this research focused on matching small, well defined and selected regions – known as "region

similarity measurement". These region similarity measurements can be calculated and located by convolution operations in the frequency domain (using FFT). Additionally, these small regions can be considered as rotation and scale invariant. This is a key area which this researcher believes could be improved significantly, and any improvements could result in significant advances in the broader image processing field. This researcher, believes that one possible way forward in this area is to use Euclid distance correlation coefficients to apply for fast classification of these regions, which will allow more accurate matching to be done by an SVM or ANN.

Wavelet transform can be applied for image noise removal and this is also planned for in the future work. Multi-core processors are becoming more affordable with the advances in computer hardware. This system also can be running on multi-core PCs- e.g. one core for calculating FFT and another core for enhancing facial features.

Further research should also be explored on applying the AAM (Cootes, Edwards, Taylor 1998) method to a face model for locating facial features. Currently this method cannot represent facial features very accurately, but if this problem could be solved, this method would be very effective as it has fewer facial features to represent. Furthermore, instead of using one facial model, this method can be improved by creating many different facial models.

Finally, this researcher also believes that there is potentially close connections between statistical regression analysis and ANNs, and that finding a way to combine these elements may allow machines to learn automatically in the same way that the human brain does.

Bibliography

Adams, R. (1993). "Radial Decomposition of Discs and Spheres." Computer Vision, Graphics, and Image Processing: Graphical Models and Image Processing, Vol. 55, Number 5, pp. 325-332.

Ahlberg, J. (2001). "Using the active appearance algorithm for face and facial feature tracking." in International Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems, .

Ahlberg, J. (2002.). "An active model for facial feature tracking." EURASIP Journal on Applied Signal Processing, vol. 2002, pp. 566–571.

Alexander, S., Hill, S., Sarrafzadeh, A. (2005). "How do Human Tutors Adapt to Affective State?" Proceedings of the 10th International Conference on User Modeling, Edinburgh , 24-30 July.

Alexander, S., Sarrafzadeh, A. (2004). "Interfaces that adapt like humans." In M. Masoudian, S. Jones & B. Rogers (Eds.), Computer Human Interaction, APCHI'04, Lecture Notes in Computer Science, Germany: Springer.

Alexander, S., Sarrafzadeh, A., Fan, C. (2003). "Pay Attention! The Computer is Watching: Affective Tutoring Systems." E-Learn 2003, Phoenix , Arizona , USA , Nov. 7-11.

Allwein, E.L., Schapire, R. E., Singer, Y. (2000). "Reducing Multiclass to Binary: A Unifying Approach for Margin Classifiers." Journal of Machine Learning Research, vol.1, pp. 113-141.

Arivazhagan, S., Deivalakshmi, S., Kannan, K., Gajbhiye, B. N., Muralidhar, C., Lukose, N., Sijo N., Subramanian, M. P. (2007). "Multi-resolution system for artifact removal and edge enhancement in computerized tomography images." Pattern Recognition Letters 28(13): 1769-1780.

Avent, R., Ng, C. and Neal, J. (1994). "Machine Vision Recognition of Facial Affect using Backpropagation Neural Networks." presented at Proceedings of the Annual International Conference of the IEEE.

Ben-Yacoub , S. (1997). "Fast Object Detection using MLP and FFT." Research report IDIAP-RR 11, at IDIAP Research Institute, http://www.idiap.ch/.

Bhatia, A. B., Wolf,E. (1954). "On the circular polynomials of Zemike and related orthogonal sets." Proc Camb Phil Soc[C] .1954,50:40-48.

Brown, R. G., Hwang, Y. C. (1996). "Introduction to Random Signals and Applied Kalman Filtering. 3 ed. New York: John Wiley & Sons."

Brunelli, R. (1990). "Edge projections for facial feature extraction." Tech. Rep. 9009-12, IRST.

Burges, C. J. C., Schölkopf, B. (1997). "Improving the accuracy and speed of support vectoe machines." Neural information processing systems,9:7.

Burt, P. J., Adelson E. H. (1983). "The Laplacian Pyramid as a Compact Image Code." IEEE Transactions on Communications 31 (4), pp. 532-40.

Byrd, R., Balaji, R. (2006). "Real time 2-D face detection using color ratios and K-mean clustering." ACM Southeast Regional Conference, Proceedings of the 44th annual Southeast regional conference, Melbourne, Florida

Canny, J. (1986). "A Computational Approach to Edge Detection." IEEE Transactions on Pattern Analysis and Machine Intelligence 8 (6), pp. 679-698.

Chang H., Robles, U. (2000). Face detection. http://www-cs-students.Stanford.edu/~robles/ee368/skincolor.html.

Cohn, J. F., Kanade, T. K., Wu, Y. T., Lien, J., Zlochower, A. (1996). "Facial expression analysis: Preliminary results of a new image-processing based method." International Society for Research in Emotion, Toronto.

Cohn, J., Zlochower, A. Z., Lien, J.J., Kanade, T. (1998). "Feature-point tracking by optical flow discriminates subtle differences in facial expression." Proceedings of the 3rd IEEE International Conference on Automatic Face and Gesture Recognition.

Cooley, J. W., Tukey J. W. (1965). "An algorithm for the machine calculation of complex Fourier series." Math. Comput. 19: 297–301.

Cootes, T. F., Taylor, C. J., Cooper, D. H., Graham, J. (1995). "Active Shape Models - Their Training and Application." Computer Vision and Image Understanding 61 (1), pp. 38-59.

Cootes, T. F., Edwards, G. J., Taylor C. J. (1998). "Active Appearance Models." 5th European Conference on Computer Vision, Vol. 2, pp. 484-498, Springer, Freiburg, Germany.

Cootes, T. F., Taylor, C. J. (2001). "Constrained active appearance models." in ICCV, pp. 748-754, .

Cristianini, N., Shawe-Taylor, J. (2000). "Support Vector Machines and other kernel-based learning methods." Cambridge University Press.

Dadgostar, F., Sarrafzadeh, A., Fan, C., De Silva, L. and Messom, C. (2006). "Modelling and Recognition of Gesture Signals in 2D Space: A comparison of NN and SVM approaches." The 18th IEEE International Conference on Tools with Artificial Intelligence, ICTAI , Washington D.C., USA.

Dadgostar, F., Fan, C., Sarrafzadeh, A. (2005). "A Hybrid Approach for Robust Real-time Face Tracking in Video Sequences." In the proceedings of the Institute of Information and Mathematical Sciences

Postgraduate Conference (pp. 35-42), Auckland, New Zealand.

Darrell, T., Essa, I., Pentland, A. (1994). "Correlation and Interpolation Networks for Real-time Expression Analysis/Synthesis." presented at Neural Information Processing Systems.

Darwin, C. (1872). "The expression of the emotions in man and animals." London: Murray.

Datcu, D., Rothkrantz, L. J. M. (2007). "Facial Expression Recognition in still pictures and videos using Active Appearance Models. A comparison approach.".

Edwards, A. T., Taylor, G. J., Cootes, C. (1998). "Interpreting face images using active appearance models." Proceedings of the 3rd. International Conference on Face & Gesture Recognition, pp. 300-308.

Ekman, P. (1971). "Universals and cultural differences in facial expressions of emotion. (Ed.), Nebraska Symposium on Motivation." Lincoln: University of Nebraska Press. .

Ekman, P., Friesen, W. V. (1976). "Measuring facial movement." Environmental Psychology and Nonverbal Behavior, 1, 56-75. .

Ekman, P., Friesen, W. V. (1978). "The facial action coding system." Palo Alto, Calif.: Consulting Psychologists Press.

Elkan, C. (2003). "Using the Triangle Inequality to Accelerate k-Means." In Proceedings of the Twentieth International Conference on Machine Learning (ICML'03), pp. 147-153.

Essa, I., Pentland, A. (1995). "Facial Expression Recognition using a Dynamic Model and Motion Energy." presented at International Conference on Computer Vision.

Essa, I., Pentland, A. (1998). "Coding, Analysis, Interpretation, and Recognition of Facial Expressions." IEEE Transactions on Pattern Analysis and Machine Intelligence: 19(7),757-763.

Fan, C., Dadgostar, F., Sarrafzadeh, A., Gholam Hosseini, H., Johnson, M. (2005a). "Facial Expression Reconstruction Using polygon Approximation." Proceedings of the 7 th IASTED International Conference on Signal and Image Processing (SIP), August 15-17, Honolulu, Hawaii, USA.

Fan, C., Dadgostar, F., Sarrafzadeh, A., Gholamhosseini, H. (2005b). "Face and eye detection using support vector machines." Proceedings of the Third International Conference on Computational Intelligence, Robotics and Autonomous Systems , 13-16 December , Singapore.

Fan, C., Sarrafzadeh, A., Dadgostar, F. and Gholamhosseini, H. (2005c). "Facial Expression Analysis by Support Vector Regression." Proceedings of the Image and Vision Computing New Zealand Conference, University of Otago, Dunedin, 28 - 29 Nov.

Fasel, B., Luettin, J. (2002). "Automatic Facial Expression Analysis: A Survey." Pattern Recognition, 36(1):259-275,2003.

Feng, G. C., Yuen, P. C. (1998). "Variance projection function and its application to eye detection for human face recognition." Elsevier Science Inc. New York, NY, USA.

Fletcher, R. (1971). "A Modified Marquardt Subroutine for Nonlinear Least Squares." Rpt. AERE-R 6799, Harwell

Gonzalaz, R. C., Woods, R. E. (2001). "Digital Image Processing." Addison-Wesley Publishing company.

Haussler, D. (1990). "Probably Approximately Correct Learning." National Conference on Artificial Intelligence.

Horowitz, S. L., Pavlidis, T. (1994). "Picture Segmentation by a directed Split-and-Merge Procedure." Proc. 2nd Int. Joint. Conf. On Pattern recognition, Aug. 13-15, pp.424-433.

Hsu C. W., Lin C. J. (2002). "A comparison of methods for multi-class support vector machines." IEEE transactions on Neural Networks, vol. 13, pp. 415-425.

Hu, M. K. (1962). "Visual pattern recognition by moment invariants." IRE Trans. on Information Theory, IT-8:pp. 179-187.

Hu, T., de Silva, L. C., Sengupta, K. (2002). "A hybrid approach of NN and HMM for facial emotion classification." Pattern Recognition Letters 23(11): 1303 - 1310.

Ilhan, J. D., Meiyappan, S. (2003). "Face Detection using Template Matching." Digital Image Processing. Spring.

Jain, A. K., Dubes, R. C. (1988). "Algorithms for Clustering Data." PrenticeHall, Englewood Clis, New Jersey.

Jones, R., Soille, P. (1996). "Periodic lines: Definition, cascades, and application to granulometrie." Pattern Recognition Letters, Vol. 17, pp. 1057-1063.

Kass, M., Witkin, A., Terzopoulos, D. (1987). "Snakes: Active Contour Models." 1st International Conference On Computer Vision, pp. 259-268, IEEE Computer Society Press, .

Khotanzad, A., Hong, Y. H. (1990). "Invariant image recognition by Zernike moments. ." IEEE Trans. on Pattern Analysis and Machine Intelligence, 12(5): 489-497.

Kovac J., Peer, P., Solina F. (2003). "Illumination Independent Color-Based Face Detection." International Symposium on Image and Signal Processing and Analysis ISPA'03, Eds. S. Lončarić, A. Neri, H. Babić, pp. I:510-515, Rome, Italy, September

Kovalevsky, F. S. (1989). "A Method for the Structural Analysis of Noisy Patterns." Proceedings of the III International Conference on Computer Analysis of Images and Patterns.

Kovesi, P. (1993). "A dimensionless measure of edge significance from phase congruency calculated via wavelets." First New Zealand Conference on Image and Vision Computing.

Kovesi, P. (1999). "Image features from phase congruency." Videre: Journal of Computer Vision Research, 1(3):1-26.

Kumar, V. P., Poggio T. (2002). "Recognizing Expressions by Direct Estimation of the Parameters of a Pixel Morphable Model." Biologically Motivated Computer Vision 519-527.

Kuwahara, M., Eiho, S. (1976). "Processing of radio-isotope angiocardiographic images. ." Digital Processing of Biomedical Images (ed. by K. Preston and M. Onoe), pp. 187-203. Plenum Press, New York.

Lee, J. S. (1983). "Digital image smoothing and the sigma filter. ." Computer Vision, Graphics and Image Processing, 24:255--269.

Lien, J. J. (1998). "Automatic Recognition of Facial Expressions Using Hidden Markov Models and Estimation of Expression Intensity." doctoral dissertation, tech. report CMU-RI-TR-98-31, Robotics Institute, Carnegie Mellon University.

Lim, J. S. (1990). "Two-Dimensional Signal and Image Processing." Englewood Cliffs, NJ, Prentice Hall, p. 548, equations 9.44 -- 9.46.

Lin, D. Chcn, J. (1999). "Facial expressions classification with hierarchical radial basis function networks." presented at the International Conference on Neural Information Processing.

Lisetti, C. Rumelhart, D. (1998). "Facial Expression Recognition using a Neural Network." presented at Proceedings of the International Flairs Conference.

Liu, L., Sclaroff, S. (1997). "Color Region Grouping and Shape Recognition with Deformable Models." Boston University Computer Science Technical Report 97-019.

Lowe, D. G. (1999). "Object recognition from local scale-invariant features." International Conference on Computer Vision, Corfu, Greece (September), pp. 1150-1157.

Lowe, D. G. (2004). "Distinctive image features from scale-invariant keypoints." International Journal of Computer Vision, 60, 2 , pp. 91-110.

Ma, L. M., Zhou, Q., Celenk, M., Chelberg, D., (2004). "Facial event mining using coupled hidden markov models." International Conference on Image Processing (ICIP), pp. 1405-08..

Martinkauppi, B. (2002). "Face color under varying illumination - analysis and applications." Dissertation. Acta Univ Oul C 171, 104 p + App, http://herkules.oulu.fi/isbn9514267885/.

McCulloch, S., Pitts, W. (1943). "A logical calculus of the ideas immanent in nervous activity." Bulletin of Mathematical Biophysics, 5:115-133. .

Min, K., Bin, L. (2006). "Facial Expression Analysis Based on Graph Spectral Decomposition." COMPUTER TECHNOLOGY AND DEVELOPMENT Vol.16 No.4 P.33-34,37.

Morrone, M. C., Owens, R. A. (1987). "Feature detection from local energy." Pattern Recognition Letters.

Moses, Y., Reynard, D., and Blake A. (1995). "Determining Facial Expressions in Real Time." presented at International Conference on Computer Vision.

Nagao, M. A., Matsuyama, T. M. (1979). "Edge Preserving Smoothing." Computer Graphics and Image Processing, vol. 9, pp.374-407.

Ohta, H., Saji, H., Nakatani, H. (1998). "Muscle-Based Feature Models for Analyzing Facial Expressions." The Third Asian Conference on Computer Vision-Volume II, pp. 711 - 718.

Oliver, N., Pentland,A., Berard,F. (1997). "LAFTER: Lips and Face Real Time Tracker with Facial Expression Recognition." presented at Computer Vision and Pattern Recognition.

Osuna, E., Freund, R., Girosi, F. (1997a). "Improved training algorithm for support vector machines." NNSP'97.

Osuna, E., Freund, R., Girosi, F. (1997b). "Training support vector machins: an application to face detetion." CVPR'97.

Pizer, S. M., Amburn, E. P., Austin, J. D. , Cromartie, R., Geselowitz, R., Greer, T., Romeny, B. T. H., Zimmerman, J. B. (1987). "Adaptive histogram equalization and its variations." Computer Vision, Graphics, and Image Processing 39(3), pp.355-368.

Reisfeld, D., Wolfson, H., Yeshurun, Y. (1995). "Context Free Attentional Operators: the Generalized Symmetry Transform." Computer Vision, Special Issue on Qualitative Vision.

Rifkin, R. M., Klautau A. K. (2004). "In defence of one-vs-all classification." Journal of Machine Learning Research, vol. 5, pp. 101-141.

Rosenblatt, F. (1962). "A comparison of several perceptron models. ." In Self-Organizing Systems. Spartan Books, Washington, DC. .

Rowley, H., Baluja, V., Kanade, T. (1998). "Neural Network-Based Face Detection." IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 20, No. 1, January, pp. 23-38[Abstract].

Rumelhart, D., McClelland, J. (1986). "Parallel Distributed Processing:Exploitations in the Micro-structure of Cognition[M]." USA:Cambridge MIT Press.

Sarrafzadeh, A., Page, C., Hamid, G., Hosseini, H. & Bigdeli, A (2004). "A New Generation Intelligent Tutoring System: Using the Learners' Kinetic, Cognitive and Affective Behaviour to Individualise and Optimise the Learning Experience." Accepted for publication in the Journal for the Congress of the ooi Junior and Senior Academies

Sarrafzadeh, A., Page, C., Overmyer S. P., Fan, C., Messom, C. (2003). "The Next Generation Intelligent Tutoring System." Proceedings of the International Conference on Artificial Intelligence in Education (AIED), Sydney, Australia.

Schölkopf, B. (1997). "Support vector learning " R. Oldenbourg verlag, munich.

Schölkopf, B., Platt, J. ,Shawe-Taylor, J., Smola, A. J. and Williamson, R. C. (2001). "Estimating the support of a high-dimensional distribution." Neural Computation, 13, 1443-1471.

Schölkopf, B., Smola, A., J., Williamson, R. and Bartlett,P. L. (2000). "New support vector algorithms." Neural Computation, 12, 1207-1245.

Shawe-Taylor, J., Cristianini, N. (2004). "Kernel Methods for Pattern Analysis." Cambridge University Press.

Singh S., Vatsa M., Singh R., (2003). "A Robust Skin Color Based Face Detection Algorithm." *Tamkang Journal of Science and Engineering,* vol. 6, no. 4, pp. 227-234.

Sobel, I., Feldman,G. (1968). "A 3x3 Isotropic Gradient Operator for Image Processing", presented at a talk at the Stanford Artificial Project in 1968, unpublished but often cited, orig. in Pattern Classification and Scene Analysis, Duda,R. and Hart,P., John Wiley and Sons,'73, pp271-2.

Sobottka, J., Pittas I. (1996). "Segmentation and tracking of faces in color images." In Proc. IEEE Int. Conf. on Automatic Face & Gesture Recognition, pages 236-241.

Starr, T. (2005). Filtering A Noisy ECG Signal Using Digital Techniques. Downloaded from http://tzilla.is-a-geek.com/school/spring2005/ee401/design/writeup.pdf

Sterring, M., Andersen, H. J., and Granum, E. (1999). "Skin Color Detection Under Changing Lighting Condition." 7-th Symposium on the Intelligent Robotics Systems,1 87～ 1 95.

Su, M., Chou, C. H. (2001). "A Modified Version of the K-Means Algorithm with a Distance Based on Cluster Symmetry." IEEE Transactions,2001; 23 (6):674-680.

Teague, M. R. (1979). "Image analysis via the general theory of moments." Journal of the Optical Society of America, 70(8):pp. 920-930

Teh, C. A. Chin, R. T. (1988). "On image analysis by the method of moments." IEEE Trans. on Pattern Analysis and Machine Intelligence, 10(4):pp. 496-513.

Tian, Y., Kanade, T., Cohn, J. F. (2001). "Recognizing action units for facial expression analysis." IEEE Transactions on Pattern Analysis and Machine Intelligence, 23 (2), pp. 97-115.

Tomita, F. A., Tsuji, S. (1997). "Extraction of Multiple regions by smoothing in selected neighborhoods." IEEE Trns. Systems, Man and Cybernetics SMC-7, pp.107-109.

Torre, F. D. L., Campoy, J., Cohn, J. F., Kanade, T. (2007) "Simultaneous registration and clustering for temporal segmentation of facial gestures from video". Proceedings of the Second International Conference on Computer Vision Theory and Applications (VISAPP) (2), pp. 110-115.

Turing, A. (1950). "Computing machinery and intelligence." Mind, vol. LIX, no. 236, October 1950, pp. 433-460. Online version: [2] (with copyright permission).

Umbaugh, S. E. (1998). "Computer Vision and Image Processing, Prentice Hall, NJ, ISBN 0-13-264599-8."

van den Boomgard, R., van Balen, R. (1992). "Methods for Fast Morphological Image Transforms Using Bitmapped Images." Image Processing: Graphical Models and Image Processing, Vol. 54, Number 3, pp. 252-254.

Vapnik, C. (1995). "Support-Vector Networks." Machine Learning, 20.

Vapnik, V. (1979). "Estimation of Dependences Based on Empirical Data [in Russian]. ." Nauka, Moscow. (English translation: 1982, Springer Verlag, New York)

Vapnik, V. C. (1974). "Theory of Pattern Recognition [in Russian]." Nauka, Moscow. (German Translation: Wapnik, W.,Tscherwonenkis, A. 1979. Theorie der Zeichenerkennung, Akademie-Verlag, Berlin.

Vapnik., V. N. (2008). "Statistical learning theoy." J. Wiley, forthcoming.

Vapnik, V. N., Steven, E. G., Smola, A. (1996). "Support vector method for function approximation." regression estimation and signal processing.

Vapnik, S. K. (1999). "The Nature of Statistical Learning Theory." Springer-Verlag, ISBN 0-387-98780-0

Vapnik, S. K. (2006). "Estimation of Dependences Based on Empirical Data." Springer, ISBN: 0387308652, 510 pages [this is a reprint of Vapnik's early book describing philosophy behind SVM approach. The 2006 Appendix describes recent development].

Venkatesh, S., Owen, R. A. (1989). "An energy feature detection scheme." Int'l Conf on Image Processing, pp. 553--557.

Viola, P., Jones, M. (2001). "Robust real-time face detection." Computer Vision, ICCV 2001. Proceedings. Eighth IEEE International Conference on Volume 2, 7-14, pp.747 – 747.

Viola, P., Jones, M. (2002). "Robust real-time object detection." Int'l. J. Computer Vision, 57(2):137–154.

Wang, Y., Lucey, S., Cohn, J. (2007). "Non-Rigid Object Alignment with a Mismatch Template Based on Exhaustive Local Search." IEEE Workshop on Non-rigid Registration and Tracking through Learning.

Wang, P., Barrett, F., Martin, E., Milonova, M., Gur, R. E., Gur, R. C., Kohler, C., Verma, R. (2008). "Automated video-based facial expression analysis of neuropsychiatric disorders." Journal of Neuroscience Methods 168, pp. 224–238.

Werbos, P. (1974). "The Roots of Backpropagation." Harvard doctoral thesis.

Wiener, N. (1949). "Extrapolation, Interpolation, and Smoothing of Stationary Time Series. New York: Wiley."

Wu, Y. T. (1997). "Image Registration Using Wavelet-Based Motion Model And Its Applications." Ph.D. thesis, Dept. of EE , University of Pittsburgh.

Wu, Y., Kanade, T., Cohn, J., Li, C. (1998). "Optical Flow Estimation Using Wavelet Motion Model." International Conference on Computer Vision (ICCV).

Yacoob, Y., Davis, L. S. (1996). "Recognizing Human Facial Expressions From Long Image Sequences Using Optical Flow." IEEE Transactions on Pattern Analysis and Machine Intelligence 18(6), pp. 636-642.

Yoshitomi, Y., Kim,S.,Kawano,T., and Kitazoe,T. (2000). "Effect of Sensor Fusion for Recognition of Emotional States Using Voice, Face Image and Thermal Image of Face." presented at IEEE International Workshop on Robot and Human Interactive Communication.

Young, I. T., Gerbrands, J. J., and van Vliet L. J. (1995). "Fundamentals of Image Processing". Delft: PH Publications.Zernike, F. (1934). "Diffraction theory of the cut procedure and its improved form, the phase contrast method." Physica, 1:pp. 689-704.

Zuiderveld, K. (1994). "Contrast limited adaptive histogram equalization." Graphics gems IV, Academic Press Professional, Inc., San Diego, CA.

Glossary

AAM	Active Appearance Models
ANN	Artificial Neural Network
BP	Back propagation
CLAHE	Contrast-limited adaptive histogram equalization
CV	Cross Validation
DOG	Difference of the Gaussian
DFT	Discrete Fourier transformation
DFT	Discrete Fourier Transformation
FACS	Facial Action Coding System
FFT	Fast Fourier Transformation
RBF	Gaussian radial basis function
GST	Generalized symmetry transforms
HSV	Hue-Saturation-Value
IDFT	Inverse Discrete Fourier Transform
MLP	Multi-layer Perception
NGITS	Next Generation Intelligent Tutoring System
OCR	Optical characters recognition
PCA	Principal component analysis
ROI	Regions of interest
RGB	Red, Green, Blue

YCbCr	Y is the luma component and Cb and Cr are the blue-difference
HSI	Hue, Saturation, Intensity Color Space
SIFT	Scale Invariant Features Transformation
OCSM	Optimized Candidates using Similarity Measurement
SVM	Support Vector Machine

Index