FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY UNIVERSITY OF MALAYA

FACE RECOGNITION AND IDENTIFICATION SYSTEM (FaceRec)

KHEW JYE HUEI
(WEK 010116)

Under the supervision of
Mr. Amirrudin Hj. Kamsin

Moderator
Ms. Mas Idayu Md Sabri

SESSION 2003/2004

This project is submitted to the Faculty of Computer Science and Information Technology, University of Malaya, in partial fulfillment of the requirement of the Bachelor of Computer Science

ABSTRACT

Face recognition is a technique to identify an individual by his/her physical characteristic and has been of interest to a growing number of researchers due to its applications on security. Several different algorithms have been proposed for computer recognition of human faces. It describes the implementation and functions of a working system that performs the recognition and identification of human faces using the implement algorithms.

Java is use as programming language to develop the FaceRec system and Microsoft Access as the database to store and access the face images for the training set. Besides that, the system will build in Windows 2000 Professional Operating System.

The project approaches the research and development process as a two dimensional recognition problem, capturing the face image by using the web camera. The process begins by preparing the database of face images for the training set before proceeding to the recognition technique for identification using eigenfaces approach. Various experiments are carried out to evaluate the performance of the working system to determine its performance and results.

With FaceRec, this system can perform recognition and identification of human faces for security purposed under certain constraint conditions in a fast and flexible way. So that, the access control will be more secure. It also allow for further research and development on the same or new approach.

ACKNOWLEDGEMENT

I wish to thank the following persons for making the thesis possible.

First of all, I would like to thank my supervisor, Mr. Amirrudin Hj. Kamsin for his guidance, support and understanding throughout the development of my final project paper. His points of view have given me a broad perspective on image processing and recognition concept, as well as his valuable ideas during consultations, implementations and thesis writing.

My sincere appreciation also goes to my moderator, Ms. Mas Idayu Md Sabri for his constructive criticism with many valuable suggestions for this project.

Not forgetting my beloved parents for their love, care and encouragement. Without their love and support, I would not be who and what I am today, I thank you from the bottom of my heart.

Besides, I would like to thank Mr. Vincent Leong for his precious guidance in choosing the technology to use and always assisting me with all my questions and confusions. He is the one to introduce me to the concept of image processing and machine face recognition.

My gratitude and affection also goes to those who have inspired me: my friends, course mates, and lecturers who gave me the eyes to understand visual expression.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1: Introduction

CHAPTER 1: INTRODUCTION

Face recognition is a hot area in computer science. There are many approaches and now there are still continual researches on this area. Developing a computational model of face recognition is quite difficult, because faces are complex, multidimensional and meaningful visual stimuli.

Identifying people by their unique characteristics has always been a major issue. Using sophisticated computerized system, a person can now be identified by machines that compare certain characteristics of the person with a vast amount of shared stored data. Some of works on face recognition focus on detecting individual features on faces, such as eyes, nose, mouth, ears, and outline of face. They have proven difficult to extend multiple views, and have often been quite fragile.

Some examples of the characteristics that can be used to identify a person are face, voice, eyes, thumb, biometric, and many more. Current technologies in identifying a person by machines are fingerprint, retina scan, iris scan, voice recognition, face recognition, and others.

Face recognition technology is one of the fastest growing fields in the biometric industry. Interest in face recognition is being fueled by the availability and low cost of video hardware, the ever increasing number of video cameras being placed in the workplace, and the noninvasive aspect of facial recognition systems. Although face recognition is still in the research and development phase, several commercial systems are currently available and more than a dozen research organizations, such as Harvard

University and the MIT Media Lab, are working on the development of more accurate and reliable systems.

This is why I propose to face recognition and identification system (FaceRec). This project implements the eigenfaces approach to face recognition and identification for security purposed.

1.1 Project Overview

In the field of security, every security system must be precision, reliability, robust and accuracy. Traditionally, access to some security devices involves the use of a login id associated with a password, device based or personal identification number (PIN). But with this old type of identification method, many users feel that their information is not secure and facing many problems as below:

- **Password based**

  Disadvantages:

  Easy to forget password, user can tell another user their password, password can be written down and password can be reused.

- **Device based**

  Disadvantages:

  Must have device to use service so the user might forget it at home, easy target for theft and still doesn't actually actively identify the user.

Identification is defined as the ability to identify a person from among all those enrolled, like all those whose biometric measurements have been collected in the database. Identification seeks to answer the question "Do I know who you are?" and involves a one-to-many comparison.

Biometrics will change all of this. Biometric security is a rapidly developing field of technology that uses biological characteristics to identify individuals. Biometric scanning is the process whereby biometric measurements are collected and integrated

into a computer system, which can then be used to automatically recognize a person. Today there are commercial applications for computer access control, access to web site servers, access through firewalls, and physical access control to protect sensitive information.

The project that is going to develop is a user friendly Face Recognition and Identification System (FaceRec) using the eigenfaces approach, whereby users could move through the system as easy as possible without any confusion. The system also important to identify the details of user for security purposed.

Face Recognition and Identification Systems (FaceRec) is a real time facial recognition system where a web cam is used to capture the frontal face image of a person and the system will then choose an image in the database that is most nearest to the captured image and would output relevant information concerning the image to identify the user if successfully found.

1.2 Statement of Problems

Problems of current recognition system:

Recognition problem is one of the critical tasks of computer system in our daily lives. Efficient recognition system will need to ensure data in computer system not be stolen or hack by other people. With the help of biometrics, we can achieve better security system. Conversely, recognition system without the aid of biometric or the current existing security systems most likely face the following problem:

- ***Easy to forget password*** – The user need to enter the user name and the password or personal identification numbers (PINs) in order to log in to a computer. This will cause some problems if the user forgets the log in password.

- ***Password unsecured*** – The hackers had the ability to hack the password of the user. This will allow unauthorized people to achieve user information, especially confidential data.

- ***Device lost or stolen*** - Recognition systems that are based on something other than an intrinsic aspect of a human being are not always secure. For example, keys, badges, tokens and access cards (or things that you physically possess) can be lost, stolen or forgotten at home.

- ***Current recognition system is costly*** – Current recognition system in the market is high-cost system. For example, companies have to pay a huge sum of money to just for setting up the **smart card** system, hardware and maintenance. It's not

cost effective for small and medium companies because they have to pay for expensive services which are not fully used all time.

- ***Slower speed*** – The processing time for the existing recognition system is slower if compare to the biometric system, such as face recognition system. The speed to access the information is much faster by using face recognition system.

- ***Low accuracy*** – The existing recognition system is not accurate every time. This is because something other than an intrinsic aspect of a human being is not always secure. Face recognition system are not susceptible to this particular problem and have very high accuracy.

Due to the above problem, biometric technology especially face recognition should be adopted to achieve reliable system.

1.3 Project Objective

The objective of this project is to develop a user-friendly face recognition and identification systems (FaceRec) using the eigenfaces approach, that could be used by any users or programmer/researcher. By study existing techniques for face recognition, (eigenfaces), an implementation for a recognition and identification systems is develop. In the user's perspective, the system can perform recognition and identification of human faces for security purposed under certain constraint conditions in a fast and flexible way. In the researcher perspective, the system could be used as a guide for their further research & development on the eigenfaces approach or others new approach which is an extended version of eigenfaces for recognition.

The objectives I aim to achieve for this specific module are as following:

- ***Develop a user friendly face recognition and identification systems (FaceRec)***

  The frontal view of face image of a person is captured by using a web camera. The image captured then will be passed into the system developed with eigenfaces. If the person is recognized, the record of the person will be displayed. The FaceRec system would be user friendly, whereby users could move through the system and used it as easy as possible without any confusion.

- ***Enhanced traditional access system by a more secured system.***

  Traditionally access system which involved user name and password will be replaced by face recognition system for more secure, precision, high accuracy, reliability, and high-speed.

- ***Implementing Eigenfaces algorithm to perform face recognition.***

  By understand the existing research by others researchers, the eigenfaces algorithm need to be implementing into the system during the development of the FaceRec system. Improvement about the technique and technology also been done in order to achieve better result.

- ***As a guidelines for further research & development.***

  Face recognition system is still in the phase of researching and not much product in the market at this moment. Thus, the FaceRec system can be used as a guideline for other researcher during the development of the face recognition system either using the eigenfaces approach or others approach.

1.4 Project Scope

Face Recognition and Identification System (FaceRec)

This is a real time recognition system that will be used for face recognition and identification purpose. In this project, an algorithm, namely Eigenfaces will be implemented to develop a prototype for user identification.

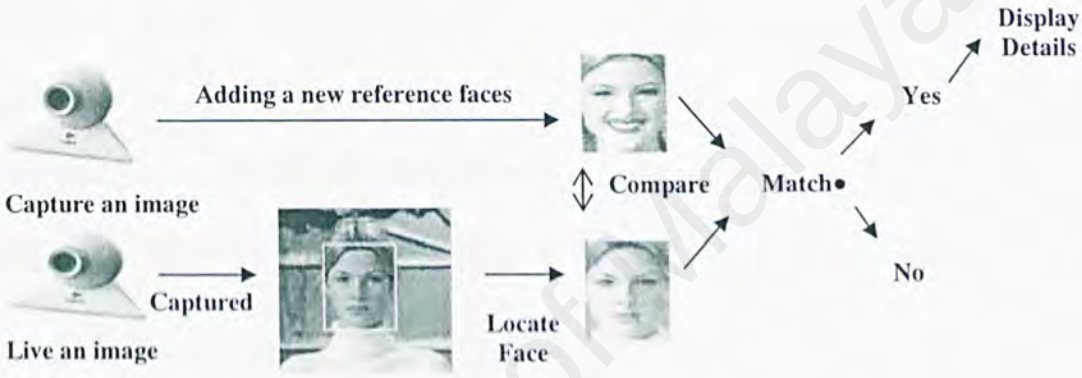


Figure 1.1 FaceRec Project Scope

The face recognition systems that will be developed would take as input a training set of face images to be compared and a test sample of face images to be recognized. Here, I am not going to handle the case that people are moving around. A webcam is used to capture a user's frontal face image in a quiet static manner. The process of capturing a face of a person and storing the image in the database is implemented using the webcam.

For the purpose of face recognition and identification, the current view of face image of a person is captured by a web camera. Then, pass the images to the system developed with eigenfaces. For the Eigenfaces algorithm, the input for the algorithm should be an image in grayscale. It is because the algorithms mainly extract the

characteristic features in the images. The web camera used will capture a colour image in JPEG format.

The image captured by the web camera will be compared to the stored images in the database to find out whether the image is present in the database. The system will then choose an image in the database that is most nearest to the captured image and if the person is recognized, the system would output relevant information concerning the image to identify the user. To perform this function, the captured image, as with the images stored in the database, would be first converted to a vector format or matrix format that could be manipulated in order to perform arithmetic calculation needed in implementing the eigenfaces approach in face recognition and identification.

For the purpose of the database that is required to store the training set of face images for comparison, a simple file-based database, called flat-file database would be used. The preprocessed face images could be input directly from the system by giving the location of the image file. It also accepts information about the person that is related to the face images.

The face recognition and identification system that would be developed would be user-friendly, whereby users could move through the system and used it as easy as possible without any confusion. The system also important to identify the details of user for security purposed.

1.5 Expected Outcome

The Face Recognition and Identification System (*FaceRec*) is expected to achieve the following outcome:

- A system that can perform recognition and identification of human faces for security purposed under certain constraint conditions in a fast and flexible way.
- System can perform some basic function and meet some importance criteria such as stability, consistency, reliability and user friendly.
- A system that can be used by any programmer/researcher for further research & development on the same or new approach for recognition method.
- A system than can be able to replace password and device based method as an identification tools.
- The final implementation should allow for future enhancement as well as additional modules to extend the system functionality.
- System able to facilitate the access authentication and ensure only authorized user can access the system.

1.6 Project Schedule

In order to organize the development phase of the system, a schedule is essential in order to develop the system in a more proper manner where the development phase follows certain time frame allocated. By using Gantt chart, a schedule of earliest possible start and finish times for the activities is given that will meet the earliest possible project completion date.

Below is a Gantt chart on the development phase scheduled along the intended time frame for each phase of the system.

WXES 3181 Project Schedule

ID	Task Name	Start	Finish	Duration
1	Introduction	16/06/2003	02/07/2003	15d
2	Literature Review	01/07/2003	21/07/2003	18d
3	Methodology	16/07/2003	26/07/2003	10d
4	System Analysis	23/07/2003	20/08/2003	25d
5	System Design	16/08/2003	06/09/2003	19d
6	Implementation	08/09/2003	07/01/2004	105d
7	System Testing	08/01/2004	09/02/2004	28d
8	Documentation	23/06/2003	12/02/2004	202d

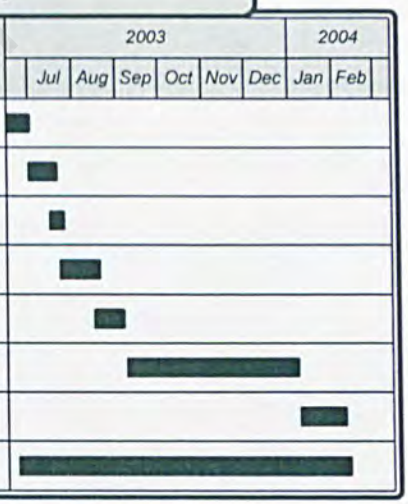


Figure 1.2: Project Schedule of FaceRec

1.7 Report Layout

This project proposal report consists of eight chapters. The purpose of this layout is to give overview of the major phases involved during development of the project. Below are the report layouts:

Chapter 1 : Introduction

This chapter gives an overview of the major phases of the project that includes the objective, project scope, project significance and project schedule.

Chapter 2 : Literature Review

This chapter gives brief explanation on topics researched and studies that are relevant to this project. It is the combination between literature search and literature review. Among the discuss topics are development tools and technology including operating system, web architecture, web server, programming languages and database.

Chapter 3 : Methodology

This chapter emphasizes on the justifications for the chosen project methodology. It also discusses the information gathering techniques and the explanation about the development software and platform chosen to develop this system.

Chapter 4 : System Analysis

This chapter describes the system analysis of the project including functional requirements, non-functional requirements, hardware and software requirements on different developing tools. It also explains how the requirements for this project were acquired.

Chapter 5 : System Design

This chapter explains the conceptual and technical design of the system. It covers the structure chart, data flow diagram, process flowchart, user interface and database design.

Chapter 6 : System Implementation

This chapter consist of the detail explanation of the implementation phase and the coding process involves transforming of the design into a programming language.

Chapter 7 : System Testing

This chapter will discuss about the testing phase. This is also a very important stage whereby testing is essential to assure quality of the system. The objective of testing is to find system error and fault.

Chapter 8 : System Evaluation

This chapter will see the system evaluation. System evaluation will touches various things like problems encountered during the development process, system strength and limitation and others.

1.8 Chapter Summary

This chapter focuses mainly on the introduction of this project. A brief introduction and definition are stated in the first part of this chapter, which is the Project Overview. Apart from that, relevant information and topics are also being discussed consequentially. Topics included are Statement of Problems, Project Objectives, Project Scope, Expected Outcome, Project Schedule, and Report Layout. The research and development of this proposed system will take about 8 months.

The next chapter literature review will carried out whereby current systems are surveyed to better understand how it is implemented, together with comparisons between different operating system platforms, development tools, databases and others.

Chapter 2: Literature Review

CHAPTER 2: LITERATURE REVIEW

2.1 Domain Studies

Review of a literature is a background study about the knowledge and information gained to develop this project. The purpose of this literature review is to gain a better understanding on the development methodologies used while developing a project.

Before the start of any project, it is important to perform various researches in the required field of interest. References, related articles and examples of previous theses have been searched and analyze to understand and recognize the existing technique used before. This is to provide relevant information as to the previous work done, technologies that was applied in developing those works and many more relevant information that can provide guidance and ideas to the development of the project. In this section, a brief overview of face recognition and its technologies is given, emphasizing in eigenfaces technologies, a long with the previous or current works done in this field.

2.1.1 Definition

Face recognition can be treated as a search space problem combined with a machine-learning problem. Given two sets of images, one is called learning set with images labeled while the other is called testing set. The main task is to tell whether each person in the testing set is recognized.

Automatic face recognition consists of four main processes: detection, capture, extraction, and recognition. Detection pertains to finding a human face and isolating it from the background and all other images. When a human face is detected, the face recognition system captures the face image using a video camera. After the face image is captured, the face recognition system extracts it from the video camera and converts it into a mathematical code that is stored in the computer as a template. Finally, recognition is performed by comparing the live face to a database of face templates to determine if there is a match.

Real-time recognition is another new aspect in face recognition. Here, a camera is used to capture the face image of a person at run time before compare with the image inside the database. Many researchers still try to develop a system which is able to identify people under different viewing angle in real-time.

- **Principal component analysis (PCA)**

Principal component analysis, based on information theory concepts, seek a computational model that best describes a face, by extracting the most relevant information contained in that face. Eigenfaces approach is a principal component

analysis method, in which a small set of characteristic pictures are used to describe the variation between face images. Goal is to find out the eigenvectors (eigenfaces) of the covariance matrix of the distribution, spanned by a training set of face images. Later, every face image is represented by a linear combination of these eigenvectors. Evaluations of these eigenvectors are quite difficult for typical image sizes but, an approximation that is suitable for practical purposes is also presented. Recognition is performed by projecting a new image into the subspace spanned by the eigenfaces and then classifying the face by comparing its position in face space with the positions of known individuals. A face recognition system, based on the eigenfaces approach is proposed. Eigenfaces approach seems to be an adequate method to be used in face recognition due to its simplicity, speed and learning capability. Experimental results are given to demonstrate the viability of the proposed face recognition method.

- **Face Identification**

Face identification can be divided into 3 categories. These categories are related to the nature of the question asked of the machine that is performing the recognition process. In each category, the machine/computer is trained with a database of faces, each associated with an identifier. The categories are as follows, depending on the question asked of the machine:-

a) **Face recognition**

The machine is supplied with a face and asked: what is the identifier associated with this face? The computer will then give as an output, the identifier for the given face.

b) **Face authentication/verification**

The machine is supplied by both a face and an identifier and asked: is this face associated to the identifier? The computer will then give as an output, a positive or negative verification.

c) **Face expression recognition**

The machine is supplied with a face that has an expression, such as smiling, and others. It needs to extract and then identify the expression on the given face. The problem involves face detection, facial expression on information extraction, and facial expression classification.

2.1.2 Current Application

Face recognition technology has numerous applications, ranging from commercial to law enforcement application. Face recognition, along with the fingerprint recognition, will remain a critical high technology strategic research area with significant potential impact on the reduction of crime. Though commercially, the use of face recognition technology is not prominent because of the cost involved, it has received a growing interest, especially in the case of credit card companies and their desire to reduce credit card frauds.

Face recognition and identification is gaining popularity and therefore applications. Applications for face recognition technology exist within corporate, industrial, government, and public sectors include:

- **Airport Security:** A Malaysian airport security force is the first in the world to incorporate face recognition technology into its airline passenger and baggage security system. The system scans passengers to ensure only real passengers are allowed to enter the aircraft. Each passenger's boarding pass contains a microchip with the passenger's face image embedded in it. The face recognition system matches the live face with that of the image stored in the boarding pass microchip before allowing the passenger to board the airplane. Additional video cameras monitor the departure lounges for identification of known airline terrorists (Page, 1998).

- **Facility Access Control:** Face recognition systems can be used to limit access to high security facilities, such as private laboratories and penitentiaries.

- **Time and Attendance Monitoring:** Face recognition systems can be used to deter fraudulent time and attendance recording. Employees cannot "clock in" without presenting their own face to the time clock, eliminating one employee clocking in for several other late or absent peers.

- **Financial Organizations:** Banks and other financial organizations can benefit from increased fraud protection during automated teller machine (ATM) transactions and large electronic fund transfers using face recognition systems.

- **Computer Network Security:** Face recognition software can be utilized to "log on" to a corporate computer network or gain access to secure files, such as payroll and accounting records. Face recognition systems can also be used to limit access to secure web sites on the internet.

- **Government Applications:** Verifying the identity of welfare and food stamp recipients can be performed quickly and easily with face recognition technology. Unemployment agencies, Motor Vehicle Departments, and other government offices can also benefit from the fraud prevention capabilities offered by face recognition technology.

2.1.3 Face Recognition Technologies

Face recognition involves the process of identifying and matching the face. Recognition is comparing the captured face to other faces that have been saved and stored in a database. There are many approaches in face recognition, and it can generally be divided into:-

i) Earlier Approaches

In the earlier approaches, measurements of distance between the eyes, are used in face recognition. Generally given a set of feature point distances of an unknown person, nearest neighbour or other classification rules are used for identifying the label of the test image. In (Kelly, 1970), measurement from the face are taken, such as the width of the head and distances between the eyes, top of the head to the eyes, between the eyes and nose, and the distance from the eyes to the mouth. These measurements will be stored for future identification. The nearest neighbour rule was used for identifying the class label of the test image and the leave-one-cut strategy was used.

Another method that was used was geometrical parameterization of the face. This model encodes the distance and angles between different face features. Many people have explored geometrical features based methods for face recognition. A face can be recognized using purely geometrical information at very coarse resolution, even when the details of the individual features (such as eyes, nose and mouth) are no longer resolved. The faces are sufficiently represented by the position and size of main face

features: eyes and eyebrows, nose and mouth. This information can be supplemented by the shape of the face outline.

ii) Statistical Approach

This approach uses mathematical equations or algorithms to detect and identify faces. There are a lot of techniques that used this approach. Orthogonal transformation techniques use either intensity–bused (pixel) or feature-based information to exploit regularities in images. These regularities are encoded in linear methods, as in eigenvector and eigenvalues, to classify images. Eigenfaces, Eigenfeatures and Local Feature Analysis are some of the method using statistical approach.

iii) Neural Network Approach

The neural Network models assume that facial images are like any other pattern. They use cognitive pattern recognition algorithms to recognize the facial images. The architecture of the neural network is based largely on current hypothesis on how the human brain works. The brain is composed of a large number of cells called neurons. Each neuron has thousands of input and outputs, which are all connected to other neurons. The input signals to a neuron are integrated and then fed to other neurons via a threshold transfer function. The neurons are modeled as processing elements in the neural network and are known as 'units'. Each unit are connected to each other and the activity of one unit will influence others. The units can be classified as input units, hidden unit, and output units. The activity of each unit can increase or decrease, by modifying the weights between the connections. Neural Networks provide non-linear

generalization of many statistical methods such as clustering. The input units are similar to sensory receptors in the nervous system and receive external information. The information normally consists of a grid of grey-level intensity value.

Neural Network has been applied to the area of facial recognition because of its ability to handle large amount of information and to encode hidden information from a training set of images. The networks can be categorized into two: single or multi layered network. In both categories, there exist two types of networks, which are the 'feed-forward' and the 'feedback' network. The feedback network has a feedback path whereby the result of the previous operation may be used by the next operation. This will provide the network with a dynamic behaviour, which gives a better performance than the feed-forward network. The use of neural networks in face recognition has addresses several issues, such as gender classification, face recognition and classification of facial expressions.

2.1.4 Method/Algorithms in Face Recognition

i. Eigenfaces Recognition

Eigenfaces recognition derives it's name from the German prefix "eigen", meaning "own" or "individual" (Ponti, 1999). The eigenfaces recognition approach was developed by (Turk and Pentland, 1991), both colleagues from MIT, in 1987. The eigenfaces method of face recognition is considered the first working face recognition technology.

The eigenfaces recognition system begins by collecting a large number of face images in a database. The system then creates a set of eigenfaces by combining all of the face images in the database and comparing commonalities and differences between groups of individual face images. The eigenfaces developed by the system appear as two-dimensional sets of light and dark areas arranged in a particular pattern. When a face is presented to the eigenfaces system for identification, the locations of the eyes are found first. The eye location provides a reference point so the head can be located and scaled to a standard size. Next, the system concentrates on the face only, leaving out clothing and hair, and removing brightness and contrast variations caused by the cameras settings. Then the program compares the live face's eigenfaces characteristics with those in the database and determines a "degree of fit" score, between -1.0 and +1.0, for the target face. If the target face produces a high enough degree of fit score when compared to a face stored in the database, it is recognized and accepted by the system. Practically any individual can be identified using a database of 100 to 150 eigenfaces.

Advantages:

1. The system fast on recognition and efficiency.
2. Easy to implement.
3. Reduces the amount of data needed to identify an individual to 1/1000th of a full sized image.
4. Provides accurate recognition rates.
5. Combined with the eigenfeatures method to become more versatile and greater accuracy.

Disadvantages:

1. Have problems in identifying faces in different light levels and pose positions.
2. The size and location of each face image must remain similar.
3. Finding the eigenvectors and eigenvalues are time consuming.

ii. Local Feature Analysis

Local Feature Analysis is a derivative of the eigenfaces method. The Local Feature Analysis method was developed by Dr. Joseph Atick, Dr. Paul Griffin, and Dr. Norman Redlich of the Visionics Corporation. Local Feature Analysis utilizes specific features for identification instead of the entire representation of the face. The system selects specific areas of the face, such as the eyes or mouth, to use as the defining features for recognition.

Local Feature Analysis face recognition systems first collect a database of several photographs and derive eigenfaces from them. The system then selects sets of blocks, or features, in each face that differ from other faces in the database. The Local Feature Analysis system uses 32 to 50 block areas to identify a face. The most common points used by Local Feature Analysis are the nose, eyes, mouth, and areas of definite bone curvature differences, such as the cheeks. The computer considers the chosen blocks as "elastic", in other words, the computer knows the blocks can move slightly in combination with one another without losing the basic structure of the face.

The identification process of Local Feature Analysis is a four step process. First, the system uses a camera to capture an image of the subject. Second, the computer determines the patterns of points that make the subject different from most other people, whether it's the nose, mouth, etc. Next, the system starts creating patterns and comparing them to the patterns of the subject's face. After a pattern is created that matches the subject's face the system searches its database for a person enrolled in the system with the same pattern. Finally, if a match is found the system allows access.

Advantages:

1. Not as sensitive to face deformities and lighting variations as eigenfaces method.
2. Can recognize faces that are not presented to the camera as a complete frontal view.

Disadvantages:

1. Difficult to implement.

iii. Mapping Recognition

Face recognition systems are based on frontal-view images of the face. However, three-dimensional face recognition systems can recognize individuals based on the three-dimensional contours of their face and head. The system uses a laser range scanner, instead of a camera, to measure the distance between thousands of specific points on the face and head. The laser range scanner produces highly accurate shape and depth data that can be interpreted by a computer. The scanned image results in a polygonal mesh of thousands of points measured on the face and head. A geometric equation is then used to compute the curvatures from point to point in the polygonal mesh. The curvature measurements of the live subject are compared to the sets stored in the database, if the measurements from the live face match a stored set the face is recognized (Gordon, 1995).

A variation of three-dimensional mapping, using only the frontal and profile views of the face to perform recognition is also being developed. When using this variation, a "mug shot" of the face is photographed with a standard camera eliminating the need for a laser range scanner. Several characteristics of the face shape, such as the forehead, nose, and chin can be used to generate a partial polygonal mesh of the subject similar to the one produced in three-dimensional mapping. The same geometric equation is then used to compute the surface curvatures of the available areas of the face to perform recognition.

Advantages:

1. Use of the entire head minimizes the possibility that the system can be fooled by a mask or similar frontal view charade.

Disadvantages:

1. Need a laser range scanner.
2. High-cost.
3. No three-dimensional mapping systems being used by the general public due to its complexity.

iv. Neural Network

Artificial Neural Networks imitate how the human brain actually works. Neural network computer systems can learn input/output relationships from their own experiences. The relationship between given inputs and their relational outputs give the computer system the ability to "learn" to recognize human faces.

Neural networks consist of several processing units called "neurons". The neurons are linked in a web or network, and communicate with one another using a "neural transfer function". The neural transfer function consists of; (1) the input signals collected from each neuron, and (2) the relational output of the input signals, which is called the "numerical result". The numerical result is the actual output of the neural network and is used when determining whether or not an image is a human face. Differentiating between an image and a human face is accomplished by training the

network with several thousand example face templates that allow the system to recognize a face even when different hair styles, face sizes, locations, and lighting conditions exist (Miros, 1999a).

Neural networks are not only used to distinguish between an object and a human face, but they are also used to verify and identify subjects in a face recognition system. Learning is performed using a trial and error method. The network is presented with several pairs of training faces and it must decide whether or not the faces are a match. If the neural network decides correctly, the next pair of faces is presented. If the neural network decides incorrectly, the system adjusts itself to provide the correct answer, completing the learning process (Miros, 1999b).

Advantages:

1. Adept at isolating a single face within a complex background because of the thousands of different facial images used to train a neural network facial recognition system.

Disadvantages:

1. Require a large database training set to achieve acceptable recognition rates.
2. The use of a database makes neural network systems larger and sometimes slower than other face recognition technologies.
3. Depend on a global representation of the face and the entire face must be visible and stationary during the recognition process in order for the system to work.

v. Thermal Imaging

Thermal imaging Face recognition systems are based on the principle that infrared cameras can capture unique heat emission patterns from a person's face. The first thermal imaging recognition system was developed by Francine Prokoski and Technology Recognition Systems in 1996 (Garfinkel, 1996).

Thermal imaging face recognition systems use an infrared camera to capture a face's unique heat patterns. Thermal imaging is possible because the vascular system, or veins, beneath the skin produce a distinctive thermal pattern when heat passes through the skin and enters the atmosphere. The heat pattern is called a "thermogram". The thermogram is converted to a digital format by the computer and saved as a template in the recognition system's database. During the digitizing process, the computer program removes the vascular patterns created by the ears and nose. The ears and nose are eliminated because of their high sensitivity to temperature fluctuations (Technology Recognition Systems, 1999). The other areas of the face template are not as easily affected by temperature and display a fairly constant heat pattern that can be consistently compared to the thermogram of a live subject.

Advantages:

1. The ability to capture and identify facial images in the dark.
2. Be useful in both law enforcement and security applications.
3. It does not recognize subjects based on the surface of the facial skin and proves to be helpful when dealing with faces that have changed due to scarring, facial hair, or eyeglasses.

Disadvantages:

1. The hardware to implement this system can be very expensive.
2. Due to high cost of infrared cameras, applications are limited to access systems where high level security is needed.

2.1.5 Existing System Research

In this section, we will not only discuss on the existing system, but also the similar system that may be thought relevant. Indeed, after the survey done, there are many implementation of the face recognition program using the eigenfaces approach have been developed using several existing programming languages such as C and Matlab. Thus, we will look into some implementation of this system, advantages and disadvantages of the examples.

Case Study 1

Photobook/Eigenfaces Demo (Vismod, 1998)

Most face recognition experiments to date have had at most a few hundred faces. Thus how face recognition performance scales with the number of faces is almost completely unknown. In order to have an estimate of the recognition performance on much larger databases, it has conducted tests on a database of 7,562 images of approximately 3,000 people.

The eigenfaces for this database were approximated using a principal components analysis on a representative sample of 128 faces. Recognition and matching was subsequently performed using the first 20 eigenvectors. In addition, each image was then annotated (by hand) as to sex, race, approximate age, face expression, and other salient features. Almost every person has at least two images in the database; several people have many images with varying expressions, headwear, face hair, etc.

This database can be interactively searched using an X-windows browsing tool called Photobook as shown below:

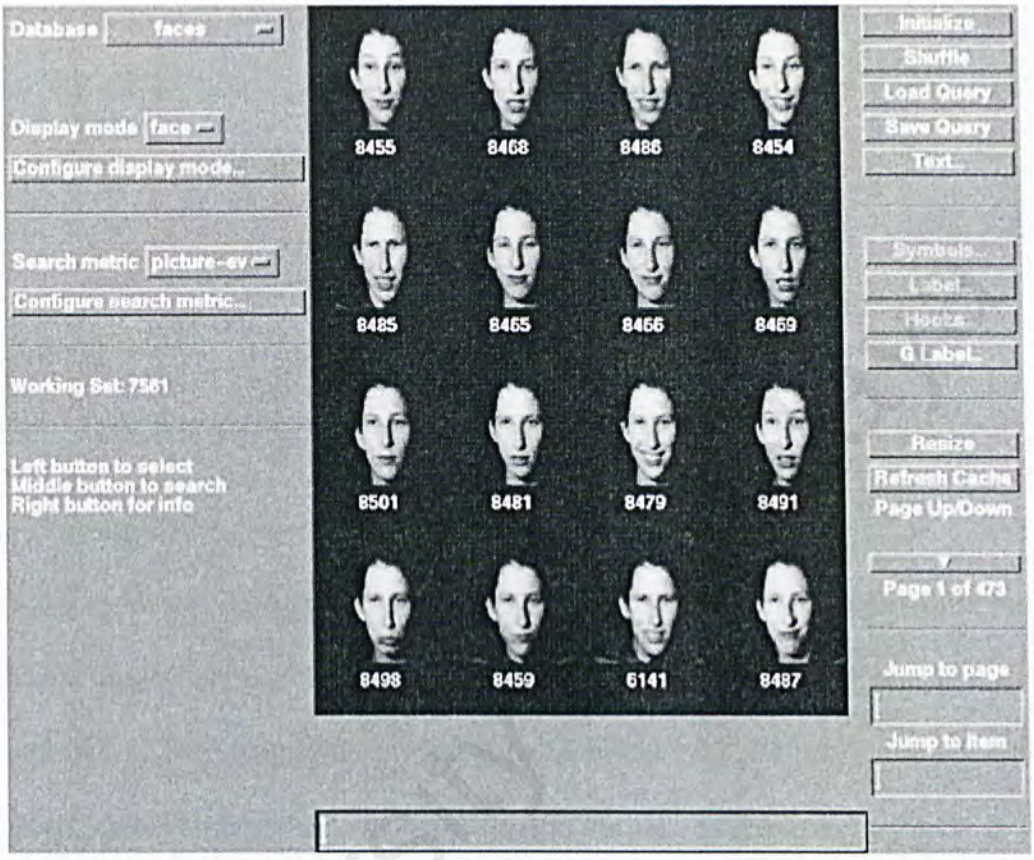


Figure 2.1 MIT Media Lab Database Photobook

The user begins by selecting the types of faces they wish to examine, for example senior Caucasian males with mustaches, or adult Hispanic females with hats. This subset selection is accomplished using an object-oriented database to search through the face image annotations. Photobook then presents the user with the top matches found in the database. The remainder of the database images can be viewed by ``paging" through the set of images. At any time the user can select a face from among those presented, and Photobook will then use the eigenvector description of that face to sort the entire set of faces in terms of their similarity to the selected face. Photobook

then re-presents the user with the face images, now sorted by similarity to the selected face.

The photobook program and the eigenfaces approach were extended to support modular eigenfaces, such as eigen-eyes, eigen-nose and eigen-mouth, and modular recognition and reconstruction.

Case Study 2

Eigenfaces Face Recognition program (Eigenfaces Group, 1999)

The eigenfaces approach in face recognition was also implemented by Tim Danner and Indraneel Datta, calling them the Eigenfaces Group. The implementation of the face recognition algorithm was done using the Python programming language, which is also known as the "executable pseudo code" programming language. By adding the Numerical Extensions and the Imaging Toolkit to the implementation, they have built a flexible and fast program that handles the eigenface approach to face recognition.

As input, the program takes in face images in the form of greyscale GIF files of fixed resolution of 250x300 pixels. These face images were first taken using Olympus 340R digital camera and then cropped (so that the face image between the eyes and the chin is taken into consideration) using the Gimp software and scaled/converted/pre-processed using the Image Magick software. The training set consists of 31 people with 3 images taken for each person giving a total of 86 images.

This program not only performs recognition on the test image of faces by comparing it with the training set of images, it also performs reconstruction of images from the calculated vectors. This will provide the ability for face checking, for determining whether an image is a face and also for image compression.

Case Study 3

"Eigenfaces" Face Recognition Program ("Facerec") by Thad Starner, Massachusetts Institute of Technology 1992 (Starner, 1992)

This is a simple implementation of the system done by Thad Starner from Massachusetts Institute of Technology, using the Eigenfaces face recognition program by Matthew Turk. The programming language used was C and run under the X Windowing System, Sun 4's, DecStation 5000's, VaxStation 2000's and 3100's, IBM RT's, and IBM RS/6000's. This method is relatively robust over different light source orientations and head orientations.

The "Facerec" program takes as input either a set of images as a training set or a set of images to be recognized. These images are stored in a directory that is taken/read as input to the program. The training_set and recognition_set images can be specified in two ways. The first format is illustrated in the file ls1, for example:-

/images/turk/face-set/bill-jarrold/111.2

/images/turk/face-set/brad/111.2

/images/turk/face-set/david/111.2

/images/turk/face-set/foof/111.2

/images/turk/face-set/irfan/111.2

Each entry is a directory containing a raw 8 bit data file and a descriptor file. The descriptor file provides information as to the dimensions of an image and how to scale its pixel values, whereas the data file stores the images in raw format. For example:-

```
(_data_sets 1)                  ; number of images per file
(_channels 1)                   ; one channel
(_dimensions 120 128)           ; x and y dimensions of image
(_data_type "unsigned_1")       ; single byte data
(scale 0.720252826167088)       ; pixel value = data * scale + pedestal
(pedestal 33.314976)
```

This is the format used by the image manipulation program "obvius" which is available via anonymous ftp at whitechapel.media.mit.edu in dist/obvius/. The training sample contains images of the 16 people taken with different light source orientations, whereas the set of images to be recognized contains images of the same people but taken with slightly different head orientation. Other files that exist for experiment purposes with this program are ls2, ls3, ts2, and ts3, also in the format shown in 1s1.

Additionally, another format can be used for specifying the training or recognition set is:-

```
facerec /v/images/face1 /v/images/face2
```

where /v/images/face1 is a directory containing

data0	data1
data10	data11
data12	data13
data14	data15

data16 data17

data18 data19

data2 data20

data21 data3

descriptor

Here each data file is a raw 8-bit image and the descriptor is

(_data_files 22) ; number of data files

(_data_sets 1)

(_channels 1)

(_dimensions 128 128)

(_data_type "unsigned_1")

No other files should reside in such directories. Otherwise, these files will be loaded as images.

To run the program in X windowing System, a user must first tailor the Makefile for the host machine and run "make". Once this is done, the binary "Facerec" should be produced. A point to be noted is that because the images are kept and read as input in directories, the paths must be changed properly to reflect the local directory structure. After this is done, many simple experiments can be done, for example by typing

facerec ls1 ls2

where 1s1 and 1s2 are the directories that contain the images to be tested/recognized.

An example of the output that the user will get will be in the format as shown below:-

/images/turk/face-set/brad/121.2/data matches/images/turk/face-set/brad/111.2/data

d=2103.989014 dffs=3419.458252

where, d: the distance of the image's projection to the face

dffs: the "distance form face space", or how far the image is for the region of space where the faces in the training set lie.

Other experiments, such as experiments with the number of eigenfaces needed for recognition, can be done by typing:

facerec is ls1 ls2 –ef 10 –d –d2

where "-d –d2" provides additional information through a text and graphics interface, respectively.

2.2 Technology Review

Before any project is started, as a developer should review the technology available to him, so he or she may choose the most idea one to be used. There more the one aspect of technology to be review on, we may have programming language, platform, technique or algorithms, and object reuse technology. They all have pro and con; a developer of the project should have them reviewed to ensure the quality of the output developed.

2.2.1 Development Models

Methodologies Review

A methodology is a collection of procedures, techniques, tools and documentation aids which helps system developers in their task of implementing a new information system. It consists of a set of phases, which consist of a set of sub phases. This guides the developers to the choice of techniques at various stages in the project and helps them to plan, manage, control and evaluate info systems project.

In the development of any software, one important aspect that we should consider is the aspect of software engineering. The method that we choose and used should be relevant and according to the system requirements of what we are going to develop. In software engineering, there are a few software process models that can be implemented to develop a system. Among the models are the 'waterfall' models, evolutionary development, formal system development, reuse-oriented development, system development life cycle, spiral development, incremental development and more.

Software engineering is an engineering discipline which is concerned with all aspects of software production from the early stages of system specification through to maintaining the system after it has gone into use.(Sommerville, 2001)

- ***Waterfall Model*** is well-defined development process in which one phase has to be finished before the next phase. The model is very simple to use. The model can be used if the requirement is well understood and defined.

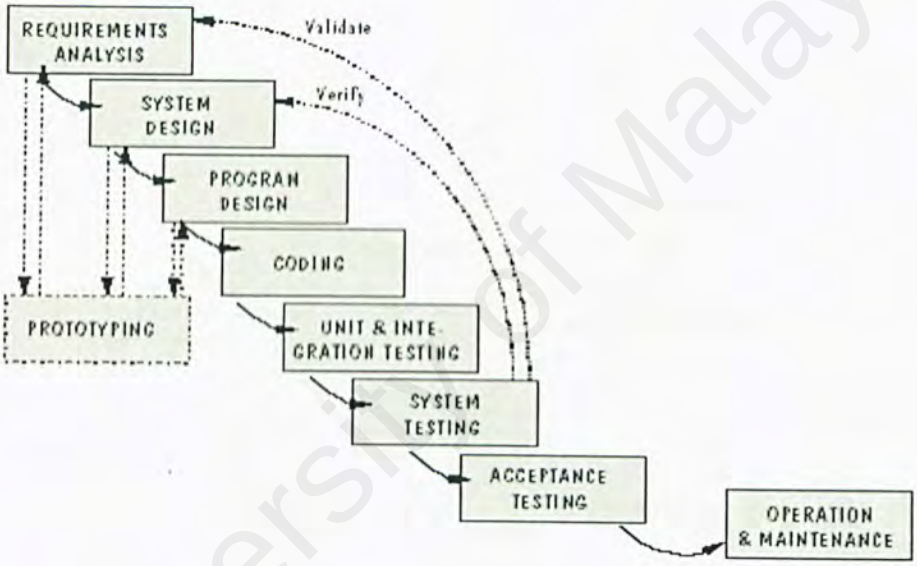


Figure 2.2: Waterfall Model

Waterfall model is used if the problem is very well understood. Usually, designers use the waterfall model to develop a simple system because it is hard to change if the model is used. If customers come by and ask for changing requirements, designers will have to start from the scratch because there is no fast way to design the system with the new requirement. Also there is a big problem for testing later if there is a change in requirement. The testers do not have well-define

set of test cases, so the test is easy to fail. In brief, waterfall model is used if designers have a well-defined list of user requirements.

- ***V Model*** is similar to waterfall model. The difference is that each test phase matches each development phase: requirements with system testing, high-level design with integration testing, and detailed design with unit testing.

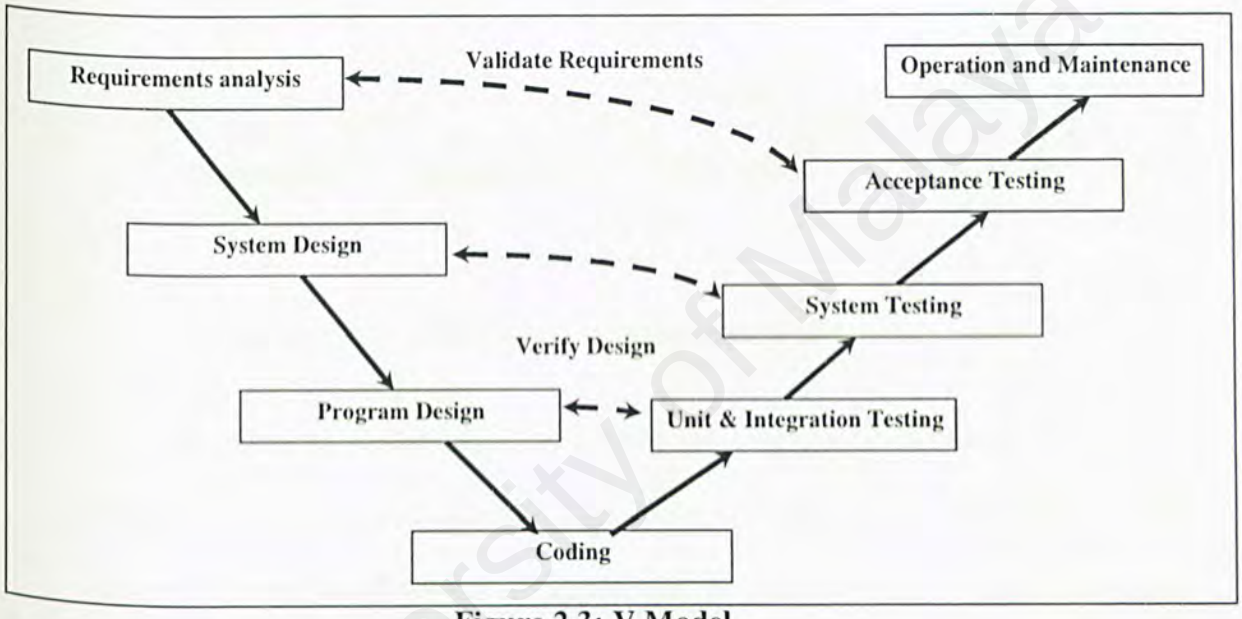


Figure 2.3: V Model

V Model is an improved version of waterfall model. V Model does not run into the problem that the software is impossible to be tested because system test, integration test, and unit test are planed ahead. For example, when we plan the requirement, we also plan for system testing. Therefore, when the system is built, we have a whole set of test cases for system testing. By that way, the system does not meet user requirements.

- ***Incremental Model.*** The designers develop the software in a number of stages and are able to deliver the product early. At each phase the designers have a goal to deliver certain features to customers. Incremental model is good for fast delivering product to the marker place.

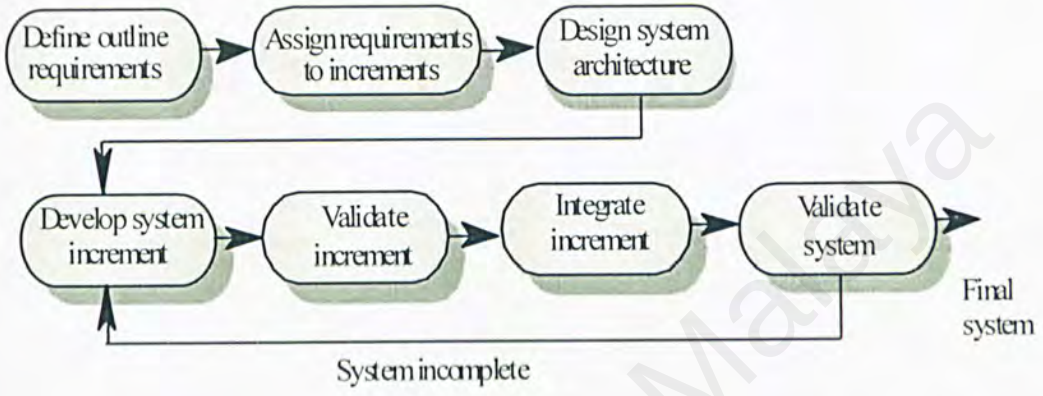


Figure 2.4: Incremental Model

Incremental model has many advantages over the other techniques. One of advantages is that the system can be developed at several stages. Each stage has its own requirement; usually it has certain features or core of the system. Each stage can use V-shape, prototype or waterfall model to develop the requirement for this stage. Regardless what kind of model is used in each stage, the product with certain features must be done at the end of the stage. Incremental model satisfies the requirement of fast delivery to the market place, so business people are interested in this model.

- ***Spiral Model*** is an iterative approach. The model carefully takes risks into account. The designers develop a small part of the project and evaluate the risks. If the risk is low, designers keep developing more features. For each iteration, there are six steps:

- Determine objectives, alternatives, and constraints.
- Identify and resolve risks.
- Evaluate alternatives.
- Develop deliverables and verify that they are correct.
- Plan the next iteration.
- Commit to an approach for the next iteration.

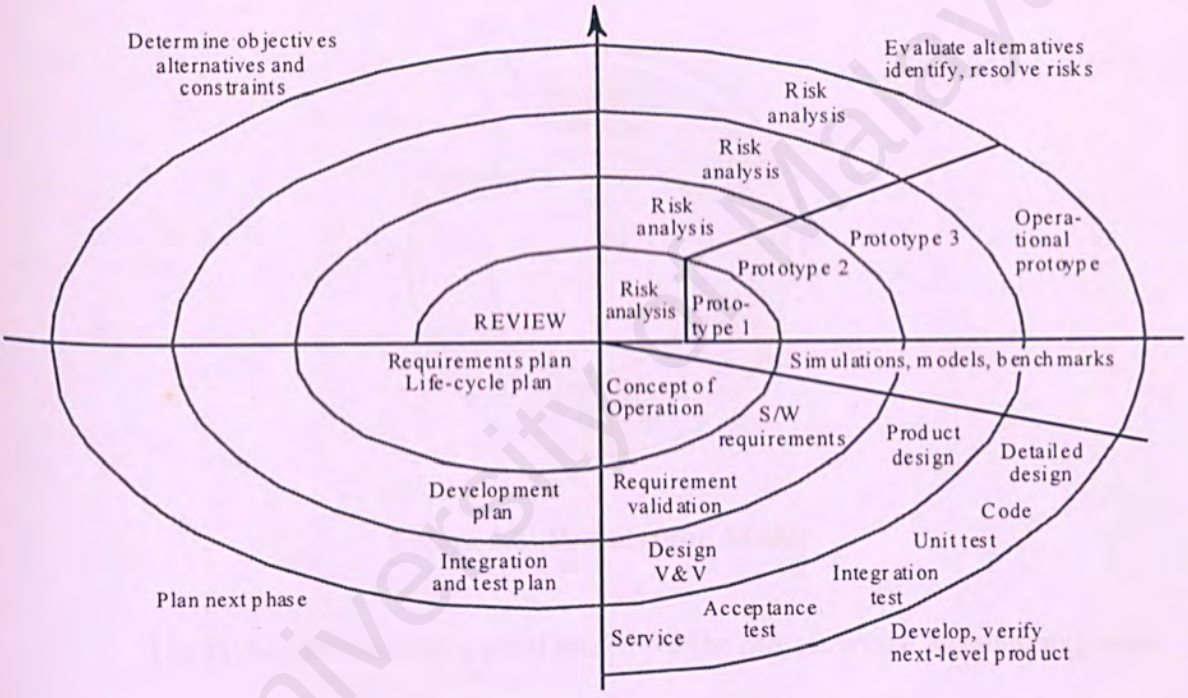


Figure 2.5: Spiral Model

Spiral model is heavily involved in risks management. If you have a project with a very high risk, you should use spiral model. Every iteration, you have a chance to evaluate the risks and to forecast whether the project keeps going or stops. For each iteration, similar to incremental model stage, designer can use V-shape, prototype, or

waterfall. Spiral model is usually used in the large project such as financial system, wireless cellular communication management system, network management system.

- ***Prototyping Model*** is the technique which helps designers and users to clarify the requirement of the system. A throw-away prototype is developed by designers and is evaluated by users. From feedback of users, designers will understand the system better and improve the prototype.

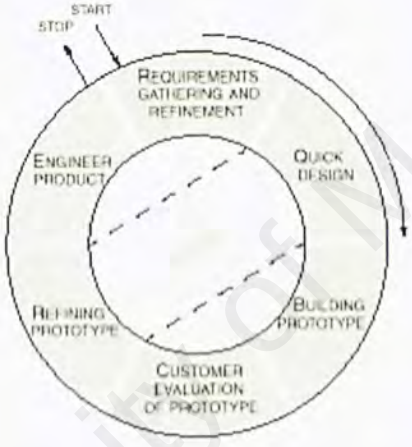


Figure 2.6: Prototyping Model

The Prototype model is a good model for the project which has unambiguous user requirement. The model will helps users to understand what they actually want. A throw-away prototype is developed so that users can realize what the system like. Human computer interface (HCI) is a big problem in requirement engineering because HCI depends on different user groups. Different user groups have different need or desire for the interface; therefore, designers have to build the prototype so that users can see and feel it.

The prototype model is also good for deploying the new technology. Before the technology is used, users are interested in know whether the technology works or not. Therefore, the prototype is a neat way to demonstrate the idea to users or customers.

2.2.2 Application Platforms

Platform Overview

For any development of the application that runs on PCs, the platform or more specific, the Operating System is always taken into accounts. It tells how far a developer needs to develop, and what has already been taken care of. Before the review is entered, a brief review on the market will be mentioned.

A research done by an organization (w3schools, 2003) stated that Microsoft Windows has dominated nearly 70% of the home user market; and more than 50% over the commercials, and there are thousand and thousands of developers are developing application than mainly adapts to it.

Beside the commercial aspects, the more important element is the technological issues. In the following descriptions, we will go through some points over the technological elements.

Operating System	Jul 02	Oct 02	Jan 03	Apr 03	Jul 03
Windows 98 / ME	64%	62%	60%	59%	40%
Windows XP					19%
Windows 2000	20%	24%	27%	29%	30%
Windows 95	4%	3%	2%	2%	2%
Windows NT	5%	4%	3%	3%	2%
MAC	1%	2%	2%	2%	2%

Table 2.1 Operating System Statistics

a) UNIX

UNIX is an operating system developed at AT&T Bell Laboratories. Traditionally used on mini computers and workstations in the academic community has started to choose UNIX for its openness. UNIX, like other operating systems, is a layer between the hardware and the applications that run on the computer. UNIX includes the traditional operating system components. It has functions that manage the hardware and the executing of applications separately. In addition, it includes a set of libraries, file system and process control.

UNIX supported the openness of the distributed system, unlike other operating system, UNIX is not limited to any specific computers using a particular microprocessor as a CPU. Instead, it runs on all sizes of computers using wide range of microprocessors.

Advantages of UNIX:

- It is portable from large system to medium-sized system to single-user systems.
- It is a powerful and mature OS and network-based platform.
- UNIX is not known only for its longevity and versatility as an operating system, but also for the variety and number of utility programs that called tool.
- UNIX is the consistent way in which it treats files. It is very easy for the users to work with files because users do not need to learn special commands for every new task.

Disadvantages of UNIX:

- Its commands are so brief that novice users find it unfriendly.
- There is on single standardized version of the operating system.

- It is too expensive to use.
- It need very powerful workstations and therefore not cost effective.

As a conclusion, UNIX is written by programmer for programmers, and it is quite popular among those fluent the ways of programming. Those unfamiliar with UNIX often have the opposite opinion. They cite as disadvantages its system commands and the existence of many version of UNIX with varying degrees of compatibility. The UNIX system does not from the current trend of hiding the system from the users. Windows and menu-oriented interfaces help to mask the commands the command, file system, connectivity method and other function from the users. In UNIX, the more aware users are of the system's internal functions, the better they can control them and improve their productivity. However, because UNIX has been adopted by a diverse community, facilitating tools called "user agent" have been furnished to shield novice user from many of the system's intricacies.

b) Linux

Linux is another version of UNIX based OS. It is builds from the long, varied tradition of UNIX command-line culture. It has become quite popular worldwide and a vast number of software programmers have taken Linux source code and adapted it to meet their individual needs. At this time, there are dozens of ongoing projects for porting Linux to various hardware configuration and purposes.

Linux supports a wide range of software, from TeX (a text formatting language) to X (a graphical user interface) to the GNU C/C++ compilers to TCP/IP networking. It is well suited to function as a development environment for web applications. Its

superior stability is a feature that cannot be beaten even by Windows. Linux is capable of running 24 hours 7 days a week without system failures or crashes. Memory management is dynamic and used memory is released after a particular application ends unlike Windows.

Advantages of Linux:

- It is as stable as UNIX.
- It is developed under the GNU General Public License and its source code is freely available to everyone.
- Highly cost-effective ability to scale the size of the site as traffic grows.
- It is capable of multitasking.
- *Has support for Netware clients and servers.*
- Includes a LAN Manager/Windows Native (SMB) client and server.
- It multi platform, means that it can run on any processor.
- Has memory protection between processors ensuring that a program cannot crash the entire system.

Disadvantages of Linux:

- It is developed by people worldwide, therefore lack proper organized support. It is more difficult to fund staff talented in any particular arbitrary combination of Linux / Apache / Jrun / Mod_Perl / PHP / Locomotive / whatever than it is to find staff talented in NT / IIS / COM.
- It is missing any pieces required to build a real application. Those pieces are problematic.

- Linux is inherently unsafe because every malicious cracker in the universe has the source code to the web site that develops under Linux.

c) Windows 98

Windows 98 is based on the popular Microsoft Windows 95 Operating System, and is designed for the consumer market. Windows 95/98 were designed for backward compatibility with older DOS and 16bit programs, as well as providing a platform for the newer (back in 1995) 32 bit programs.

Windows 98 works better by making it simple to access the Internet and by providing better system performance along with easier system diagnostics and maintenance. With Windows 98, users' system plays better as well with support for the latest graphics, sound, and multimedia technologies, the ability to easily add and remove peripheral devices with support for Universal Serial Bus (USB), and it also enables users to watch TV on PC. Besides that, Windows 98 is compatible with more software (including games) and hardware.

d) Windows NT

Windows NT is the operating system for personal computer created for users and business requiring advanced capability. It is actually comprised of two products: Microsoft NT Workstation and Microsoft NT Server. The workstation is a little safe than Windows 98 and Windows 95. It is designed for users especially business users, who need faster performance.

The server is designed for business machines that need to provide services for LAN-attached computers. Together with an Internet server such as Microsoft's Internet Information Server (IIS), it is required for a Windows system that plans to serve Web pages.

Advantages:

- The system would need to run on different hardware platforms with minimal changes.
- It could be locked down through software, meeting NSA's C2-level criteria.
- It would be POSIX-compliant, run existing Windows applications, and support open international standards.
- It would support symmetric multiprocessing (SMP).
- It could be easily expanded on by writing to a well-defined application programming interface (API).
- It could easily be ported to run in numerous different languages and writing systems, with minimal modifications to the software.

Disadvantages:

- Require license fee.

e) Windows 2000 Professional

Microsoft Windows 2000 Professional is Microsoft's latest version of popular Windows NT Operating System. Windows 2000 Server has big improvement over Windows NT 4.0. The changes, both fundamental and cosmetic, have made Windows 2000 faster, more reliable, heavier-duty, and easier to use.

Advantages:

- Point-to-point Tunneling Protocol (PPTP) of Windows 2000 provides a way to use public data networks, such as the Internet, to create a virtual private network-connecting client PCs with servers. PPTP offers protocol encapsulation to support multiple protocols via TCP / IP connections and data encryption.
- System Policy Editor and User Profiles of Windows 2000 allow system administrators to manage and maintain users' desktops in a consistency manor. System policies are used for the standardization of desktop configurations and control the user work environment and actions.
- New application programming interfaces for server application developers and better server performance deliver improved throughput and scalability for server applications such as Microsoft SQL Server.
- The Task Manager of Windows 2000 is an integrated tool for monitoring applications and reports key performance metrics of the Windows 2000 system. It provides information on each application and process that are running on the workstation, as well as memory and CPU usage.
- Improved Windows 2000 Diagnostics Tool allowed for easy examination of the system resources such as IRQ, DMA and IO addresses, all presented in an easy-to-view graphical tool.
- Network Monitor of Windows 2000 is a powerful network diagnostic tool allows examining network and from the server at the packet-level. It allows capturing network of traffic for later analysis, this making troubleshooting network problems easier.

- Windows 2000 allows Object Linking and Embedding (OLE). It can combine the information from several applications into one compound document using the special OLE capabilities of Window-based application.

Disadvantages:

- Insignificant changes to a Windows 2000 configuration require a shutdown and reboot in order to make the changes take effect. Change the IP address of the default gateway also has to reboot.
- The ongoing maintenance and support requirement of Windows 2000 can make them much more costly to run.

f) Windows XP

Windows XP is the next version of Microsoft Windows beyond Windows 2000 and Windows Millennium. Windows XP brings the convergence of Windows operating systems by integrating the strengths of Windows 2000—standards-based security, manageability and reliability with the best features of Windows 98 and Windows Me—Plug and Play, easy-to-use user interface, and innovative support services to create the best Windows yet.

Microsoft Windows XP offers many new features. And it's excellent overall performance which includes dramatically faster boot and resume times, along with highly responsive applications—creates the conditions for a very satisfying user experience.

Advantages:

- *Memory and Performance* - Microsoft recommends that computers running Windows XP have at least 128 megabytes (MB) of random access memory (RAM) installed. For this memory size, Windows XP has shown itself to be consistently superior to previous versions of Windows. Performance only gets better with additional resources, particularly when run memory-intensive multimedia applications.

- *Performance Evaluation* - Assessing system performance requires a variety of approaches and realistic workloads that represent how the operating system will be used. The Windows Performance Team used both externally- and internally-written benchmarks, in addition to side-by-side comparisons, to measure the performance of successive versions of Windows XP.

- *Resume from Standby and Hibernation* - Standby and hibernation are two alternatives that are particularly important for laptop users seeking to conserve battery life. Windows XP significantly increases the speed of going into and coming out of standby and hibernates.

Disadvantages:

- *Run-Time Performance* – Windows XP ran much slower than Windows 2000, even when they turned off.

- *High cost* – Windows XP is more expensive compare to others Microsoft programming language.

2.2.3 Programming Language

a) C++

The most essential part of support for object-oriented programming is the class/object mechanism, which originated in SIMULA 67 and is the central feature of SmallTalk. C++ provides a collection of predefined classes, along with the possibility of user-defined classes. The classes of C++ are data types, which, like SmallTalk classes, can be instantiated any number of items. Such as instantiation in C++ are merely objects, or data declarations. Class definitions specify data object (data member) and the functions (called method). Classes can name one or more parent classes, providing inheritance and multiple inheritances, respectively. Classes inherit the data members and member functions of the parent class that are specified to be inherited.

Operations in C++ can be overloaded, meaning the user can create operators for existing operators on user can define more than one function with same name, provided either the number of their parameter are different.

Dynamic binding in C++ is provided by virtual class functions. These functions define type-dependence operators, using overload functions, within a collection of classes are related through inheritance. A pointer to an object of class A can also point to an overloaded virtual function, the function of the current type is chosen dynamically.

Both function and classes can be template, which means that they can be parameterized. For example, a function can be written as a template function to allow it to have versions for a variety of parameter types. Classes enjoy the flexibility. C++

includes exception handling that is significantly different from that of Ada. One of the difference is that hardware-detectable exceptions.

Advantages:

- Object-oriented programming methodology
- Availability of good and inexpensive compilers.
- Almost completely downward compatible with C and possible to link C++ code with C code.
- C++ program will normally run much faster than a BASIC program.

Disadvantages:

- It inherited most of the insecurities if the C and makes it less safe than language such as Delphi, Ada and Java.

b) Java

Java designer started with C++, removed numerous constructs, changed some, and added a few others. The resulting language provides much of the power and flexibility of C++, but in a smaller, simpler, and safer language.

Java, like many programming languages, was designed for an application for which they appeared to be no satisfactory existing language. In the case of Java, however, it was actually a sequence of applications, the first of which was the programming of the embedded consumer electronic devices such as toaster, microwave ovens, and interactive TV system. It may not seem that reliability would be an important factor in the software for a microwave oven. If an oven dad malfunctioning software, it

probably would not pose a grave danger to anyone and probably would not lead to large legal settlements. However, if the software in a particular model was found to be erroneous after a million units had been manufactured and sold, the recall would entail significant cost. Therefore, reliability is an important characteristic of the software in consumer electronic product.

Advantages:

- Object-oriented programming methodology
- Smaller, simpler, and safer language.

Disadvantages:

- Java compiler is slow compare to other programming language compiler.
- Is not suitable for huge program which need to "run" faster.

c) Microsoft Visual C++

Microsoft has built into its C++ development environment to enable developers to create very advanced applications for the Windows and NT platforms. When Microsoft's developers first came up with the idea behind Visual C++, they decided to take their world-class C++ compiler and create a development environment and set of tools that would enable developers to create Windows applications with a level of ease and speed that was unheard of among C++ development environments. Since that first version, Microsoft has continued to improve the tools that are a part of Visual C++ to make it even easier to create Windows applications. As Microsoft has introduced new

technologies into the Windows platforms, it has also introduced tools into the Visual C++ suite to make it easy to integrate new technologies into developers' applications.

Visual C++ is a tool for building 32-bit applications for Windows 95 and Windows NT. These applications are much larger and more complex than their predecessors for 16-bit Windows or older programs that didn't use a graphical user interface. Yet, as program size and complexity has increased, programmer effort has decreased, at least for programmers who are using the right tools.

Visual C++ is one of the right tools. With its code-generating wizards, it can produce the shell of a working Windows application in seconds. The class library include with Visual C++, the Microsoft Foundation Classes (MFC), has become the industry standard for Windows software development in a variety of C++ compilers. The visual editing tools make layout of menus and dialogs a snap. The time programmers invest in learning to use this product will pay for itself on programmers' first Windows programming project.

d) Microsoft Visual Basic

Visual Basic programming language is fairly simple and uses common English words and phrases for the most part. Visual Basic has evolved from the simplest programming language for Microsoft Windows to an exceedingly complex development environment, capable of delivering virtually anything from tiny utilities to huge n-tier client/server applications.

Controls are tools on the Toolbox windows that you place on a form to interact with the user and control the program flow.

Microsoft Visual Basic 6.0, the latest and greatest incarnation of the old BASIC language, gives a complete Windows application development system in one package. Visual Basic (or VB, as we often call it) lets ones write, and test Windows applications. In addition, VB includes tools ones can use to write and compile help files, ActiveX controls, and even Internet applications.

A survey of the new Visual Basic 6 features: Visual Basic 6 includes many new features, especially in the database and Internet areas. Among these are ADO, DHTML application, and Web Classes, just to mention the outstanding ones.

A demystifying tutorial on object-oriented programming; one have been able to build classes using Visual Basic since version 4, yet a relatively small number of developers actively use objects in their applications. This isn't surprising, after all, since most of the available OOP code examples are "toy classes" that deal with dogs, fleas, and other animals. This hardly stimulates the imagination of programmers who must deal with invoices, products, consumers and orders.

- **C++ vs. Java**

In the last couple of years, there has been a great deal of interest in the Java programming language. Why should developers choose C++ over Java? In the first place, a compiled program will always faster than an interpreted program. Think about a high- performance spreadsheet program with cell formulas and macros. Now imagine

the Java virtual machine interpreting the code that, in turn, interprets the formulas and macros. Not pretty, is it? With just-in-time compilation, it's necessary to compile the program every time programmers load it. Will that code be as good as the optimized output from a C++ compiler?

Execution speed is one factor; access to the operating system is another. For security reasons, Java applets can't perform such tasks as writing to disk and accessing serial ports. In order to be platform-independent, Java application programs are limited to the "lowest common denominator" of operating system features. A C++ program for Microsoft Windows is more flexible because it can call any Win32 function at any time.

Java will be an important language, but we believe it's just another language, not a revolution. If developers need an Internet applets or a truly platform-independent application, choose Java. If developers need efficiency and flexibility, choose C++.

2.2.4 Authoring Tools

a) Microsoft Visual Studio 6.0

Microsoft Visual Studio 6.0 is a complete development tools suite that provides corporations with easy-to-use tools for building solutions. A World class software, from the leading web browsers to mission-critical corporate applications. Visual Studio is the most productive tool for the highest-performance development for Windows and the Web.

Microsoft Visual Studio 6.0 contains a set of technologies and training materials designed to help developers take advantage of new Windows 2000 features. It provides the usage of the Windows 2000 Developer's Readiness Kit for fast access to in-depth, step-by-step training and technical resources. Developer can build highly reliable, self-repairing applications using the new Visual Studio Installer. Programmer can create desktop and shared solutions that easily scale up to SQL Server™ 7.0 using the new Microsoft® Data Engine (MSDE) for Visual Studio.

Benefits:

- Includes new versions of the Microsoft Visual Tools, enabling developers to build component-based solutions.
- Easy to create, consume and reuse components with all of the Visual Studio tools.
- Improved integration and productivity enhancements across all the Visual Tools.
- Visual Studio 6.0 was designed to increase developer efficiency and reduce application development time.

- Offers new productivity features throughout the application development life cycle.
- provides a new generation of database programming and design tools based on Microsoft's Universal Data Access technology.

Drawback:

- Includes new versions of the Microsoft Visual Tools, enabling developers to build component-based solutions.
- Easy to create, consume and reuse components with all of the Visual Studio tools.

b) Microsoft Visual Studio .NET

.NET Framework is an environment for building, deploying and running Web services and other applications. It consists of three main parts: the Common Language Runtime, the Framework classes, and ASP.NET.

It's basically a single platform that anybody can develop for, using a system similar to Java/JVM. Only unlike Java/JVM, there's no language barrier with .NET: The languages available to the developer upon installation of the SDK (software development kit) are Managed C++, C#, Visual Basic and JScript; even more are on the way.

An itemized list of the technical components making up the .NET platform:

- **C#**, a "new" language for writing classes and components that integrates elements of C, C++, and Java, and adds additional features, like metadata tags related to component development.
- A "**common language runtime**", which runs byte codes in an Internal Language (IL) format. Code and objects written in one language can, ostensibly, be compiled into the IL runtime, once an IL compiler is developed for the language.
- A set of **base components**, accessible from the common language runtime, that provide various functions (networking, containers, etc.).
- **Win Forms and Web Forms**, new UI component frameworks accessible from Visual Studio.
- **ADO+**, a new generation of ADO data access components that use XML and SOAP for data interchange.

Advantages:

- It offers multiple language support.
- It has a rich set of libraries, a la JVM.
- It's open-standard friendly (e.g., HTTP and XML) -- it may even become a standard itself.
- Its code is compiled natively, regardless of language or deployment (Web or desktop).

Disadvantages:

- It's yet another platform to consider, which generally means rewriting and learning new tricks.
- Microsoft tends to have good ideas, but mediocre implementation.

- Currently, it's only available on Windows.
- Microsoft claims C#, IL, and CLR/CLS will be submitted to ECMA, but there's still no clear view on what will be standardized from the platform.

c) JCreator 2.5

JCreator 2.5 is a powerful IDE for Java. JCreator provides the user with a wide range of functionality such as: Project management, project templates, code-completion, debugger interface, editor with syntax highlighting, wizards and a fully customizable user interface.

Advantages of JCreator 2.5:

- **Improved Reliability** - Can directly compile or run the Java program without activating the main document first. JCreator will automatically find the file with the main method or the html file holding the java applet, and then it will start the appropriate tool.
- **Increased Performance** - JCreator is written entirely in C++, which makes it fast and efficient compared to the Java based editors/IDEs.
- **Developer Productivity** - Professionally designed to meet windows interface guidelines, JCreator users can expect a familiar and intuitive user interface.

2.2.5 Database Management System

There are few types of database management systems:

- *Hierarchical Model*

  A hierarchical DBMS represents data as tree structures, composed of a hierarchical of data records.

- *Network Model*

  A network DBMS represents data as records linked together, forming intersecting sets of data.

- *Relational Model*

  But even the most complex hierarchical and network database can be represented as a simple connection of two-dimensional tables, otherwise known as a relational DBMS.

  In a relational system, only one type of data stricture exists – the table. This uniformity gave rise to a new type of high-level database language. A relational system provides automatic navigation to the needed data. There is no need to acknowledge how data is represented in storage to get information in and out of a relational database. More than any other feature, it is automatic navigation that makes data readily accessible to end-users.

- *Object Oriented Model*

  The Object Oriented Model allows directly storing and managing of objects through standard language interfaces, including C++, java, Smalltalk and SQL, using traditional programming techniques and tools.

a) Microsoft SQL Server 2000

Microsoft SQL Server 2000 is the latest version of Microsoft's relational database management system (RDBMS) for the Windows platform. It is a complete database and analysis solution for rapidly delivering the next generation of scalable Web application. The database component of SQL Server 2000 is a Structured Query Language (SQL) – compatible, scalable, relational database with integrated XML support for Internet application. Customer needs and requirements have driven significant product innovations in ease of use, reliability and scalability and data warehousing. SQL Server 2000 runs on Windows NT 4.0 or Windows 2000.

SQL Server 2000 exceeds dependability requirements and provides innovative capabilities that increase employee effectiveness, integrate heterogeneous IT ecosystems, and maximize capital and operating budgets. SQL Server 2000 provides the enterprise data management platform that organization needs to adapt quickly in a fast-changing environment.

With the lowest implementation and maintenance costs in the industry, SQL Server 2000 delivers rapid return on data management investment. SQL Server 2000

supports the rapid development of enterprise-class business applications that can give company a critical competitive advantage.

Benchmarked for scalability, speed, and performance, SQL Server 2000 is a fully enterprise-class database product, providing core support for Extensible Markup Language (XML) and Internet queries. Below are the lists of the features of Microsoft SQL Server 2000:-

Features:

- Provides a powerful and comprehensive data management platform.
- Ease of installation or upgrade.
- Ease of deployment where size of a database change automatically
- Includes several features that extend the scalability of the system.

b) Oracle 8i

Oracle is the world's leading vendor of database software. Oracle's ability to have all data and documents stored in a small number of high-performance databases benefits customers by centralizing all their data, making information management and access easier, more reliable and less expensive. The ground-breaking capabilities of Oracle 8i's Internet File System (IFS) provides a single, easy to use data management interface for all data types, thus minimizing customer's reliance on a proprietary operating system. Oracle is an open solution and it supports all kind of platform.

Oracle uses a Java-based utility that provides everything needs to get a pre-tuned and pre-configured Oracle 8i database up and running. Oracle Enterprise Manager

provides a single integrated management console for central administration of multiple servers. It also contains some advance functionality for tuning and diagnosing the database, and managing complex change in the database environment. Oracle's advanced security features allow for enforced granular privileges, advanced auditing, enhanced access control, secure distributed processing and replication, and the ability to use additional external authentication mechanisms.

c) MySQL

MySQL is a true multi-user, multi-threaded SQL database server. SQL is the most popular and standardized database language in the world. MySQL is a client/server implementation that consists of a server daemon mysqld and many different client programs and libraries.

The main goals of MySQL are speed, robustness and ease of use. MySQL was originally developed because we needed a SQL server that could handle very large databases an order of magnitude faster than what any database vendor could offer to us on inexpensive hardware. We have now been using MySQL since 1996 in an environment with more than 40 databases containing 10,000 tables, of which more than 500 have more than 7 million rows. This is about 100 gigabytes of mission-critical data.

The base upon which MySQL is built is a set of routines that have been used in a highly demanding production environment for many years. Although MySQL is still under development, it already offers a rich and highly useful function set.

➢ **Microsoft SQL Server 2000 Vs Microsoft Access 2000**

Microsoft Access 2000 is a good Database Management System but it is not comprehensive and large enough to fit in all the image. The comparison between Microsoft SQL Server 2000 and Microsoft Access 2000 are shown below:-

Features	**MS SQL Server 2000**	**MS Access 2000**
Data capacity	1 Terabyte per database	1.2 Gigabytes of database
Solution for transaction based database downtime	Automatic rollback, reduction of downtime expenses	Do not support automatic recovery, loss of data possible
Backup ability	Dynamic backup	No built-in back-up capability
Security	Validation using Windows 2000 or NT. Also uses Login ID and password, user permission and encryption	Customizable security available depending on developers needs
Application runtime	Fast	Slow
Application number of users	32,767	255
Distributed transaction	Yes	No

Table 2.2 Microsoft SQL Server vs. Microsoft Access

2.2.6 Data Access Technology

Universal Data Access

Universal Data Access is a set of common interfaces that anyone can use to represent data kept in relational databases, spreadsheets, project and document containers, VSAM, email, and file systems. The data stores simply expose common interfaces - a common data access model - to the data.

The Microsoft Universal Data Access strategy is based on OLE DB. OLE DB is a set of low-level C/C++ interfaces designed to efficiently build database components. They allow an individual data store to easily expose its native functionality without having to make the data look like relational data if it isn't, and also allow generic, reusable database components to augment the functionality of simpler providers when needed. Because they are low-level interfaces that deal with pointers, structures, and explicit memory handling for optimizing how data is exposed and shared between components, they are not suitable for calling directly from Visual Basic® or Java.

a) ADO

ADO looks very similar to Data Access Objects (DAO) and Remote Data Objects (RDO) because Microsoft took the best features of both models and put them together in a common programming model.

ADO is easy to use, language-independent, and provides extensible interfaces for programmatic access to all types of data. Since ADO consumes OLE DB interfaces, whose sole purpose is to expose native functionality from diverse types of data stores,

we can get to all types of data through ADO. ADO was designed as a set of interfaces that would be common for any high-level data access library that is going to supply a common programming model. And it really is easy to use. Language-independent means that ADO is an automation server that supports dual interfaces. Dual interfaces means that ADO supports an IDispatch as well as a VTBL interface, so that applications written in Visual Basic, C++, and Java can bind directly to the VTBL interface and get fast performance. Scripting languages can use the same objects through IDispatch.

All objects in ADO can be instantiated on their own, except the Error and Field objects. The hierarchy of objects found in previous models like DAO and RDO is de-emphasized in the ADO model to allow greater flexibility in reusing objects in different contexts.

b) Microsoft Open Database Connectivity (ODBC)

ODBC is the standard interface to SQL relational data. There is an ANSI/ISO specification that is an extension to ANSI SQL92, ISO's SQL specification for a Call-Level Interface based on ODBC. In fact, the new ODBC 3.0 is a fully compliant superset of that international standard.

There is a great deal of incentive for relational database programmers to write to ODBC. But ODBC does require that our data look like a relational database, so it's not always the best way to expose data. ODBC's greatest strength is that it closely follows the embedded SQL model. So if we have a SQL relational database, ODBC is almost trivial for us to implement. If we don't have a SQL database, it can be very difficult to

write an ODBC driver to our data because we basically have to write a relational engine on top of our data.

To integrate ODBC into the Universal Data Access strategy, Microsoft built an OLE DB-ODBC provider to ensure that anybody's existing ODBC data works well in the world of Universal Data Access. It implements the OLE DB object types that share component data objects on top of any ODBC driver. It does so in a very efficient manner; we can access the data through OLE DB with the same performance that we would get if we accessed it through ODBC. Without losing our investment in ODBC, we can communicate with ODBC data stores directly through OLE DB interfaces or ADO since ADO consumes OLE DB interfaces.

2.3 Chapter Summary

All these researches were done mainly to gain information for this project. The information gathered includes the algorithms to apply in face recognition system, reviews on the existing research, development methodologies, and development tools.

Research on algorithm or method in face recognition is to have a better understanding and the benefit for each algorithm such as eigenfaces and neural networks. The review on previous research was done by browsing through the internet.

As for the development methodologies, this review of literature focuses mainly on five development models that are the Waterfall model, V-shape model, prototyping, Incremental model and Spiral model. Each modal has its own features, which are different from various resources.

The information gathered in development tools for the project was analyzed. All these information was obtained from the internet using search engines and other resources. Information on four different server platforms such as Microsoft Windows NT, Microsoft Windows 2000, Linux and Unix had been gathered and analyzed. Each platform has its strengths and weaknesses.

The information gathered on programming languages such as Visual C++, Visual Basic and Java are also being analyzed. As for the authoring tools, information was gained from sources like books and Internet. All the development technologies and programming languages chosen for this project will be mention in next chapter.

After research and technology consideration, the proposed project – Face Recognition and Identification System (FaceRec) is feasible to develop.

Chapter 3: Methodology

CHAPTER 3: METHODOLOGY

3.1 Software Development Life Cycle (SDLC)

Software Development Life Cycle (SDLC) is a process model of systems development which used by organizations to describe their approach to producing computer systems. Software development life cycles are abstract model that define the activities, ordering of activities and information flow associated with the development of software. Life cycles exist for describing the entire development process as well as for particular focused activities such as system test or preparing to release software. Basically, SDLC contains 7 phases in the whole process, which are:

- Identifying problems, opportunities and objectives
- Determining information requirements
- Analyzing system needs
- Designing the recommended system
- Developing and documenting system
- Testing and maintaining the system
- Implementing and evaluating system

3.2 Methodology Consideration

A system development methodology is a collection of procedures, techniques, tools and documentation aids which helps system developers in their task of implementing a new information system. It consists of a set of phases, which consist of a set of sub phases. This guides the developers to the choice of techniques at various stages in the project and helps them to plan, manage, control and evaluate info systems project.

The main objectives of following a methodology is to make the development cycle as efficient as possible, to complete development within lowest possible cost keeping the highest quality, and to achieve the fastest turn-around. Another important objective is to make future maintenance easier and faster. The development cycle for each and every project is some way unique, depending on system requirements and their unique operating environment. Design and development methodology also varies depending on the software, hardware technologies chosen.

3.2.1 Characteristics of a Good Methodology

A good methodology has the characteristics as below:

- Easy to use for average analyst and programmers.
- Well quality documentation is available.
- Good vendor support in terms of training and consultancy.
- Covers all phases of system development.
- Relevant to the type of application being developed (Transaction Processing System, Management Information System or others).

3.2.2 Benefits of Good Methodology

A good methodology that able to provide the effective ways of system development is best defined before the project starts and then becomes the framework to development staff.

Some benefits offers by a good methodology:

- Provides a standard framework that the developer does not have to reinvent the wheel for each project.
- Each method or tool in the methodology results in successful completion of each development task.
- Reviews procedures are available to identify any errors, inconsistencies and discrepancies during development.
- Increase the system quality by forcing the developer to produce flexible systems and adequate documentation.
- Provides better understanding of user needs and validation of user needs.
- Provides the management with tools to review project progress and checklist to access tasks and deliverables.
- Improves communication among management, analyst, programmers, users and other stakeholders by providing a communication base.
- Facilitates planning and controlling the project.

3.2.3 Conclusion on Development Methodology

The System development methodology is a method that to create a system with a series of steps or operations or can be defined as system life cycle model. Every system development process model (see figure 3-1) includes system requirements (user, needs, resource) as input and a finished product as output.

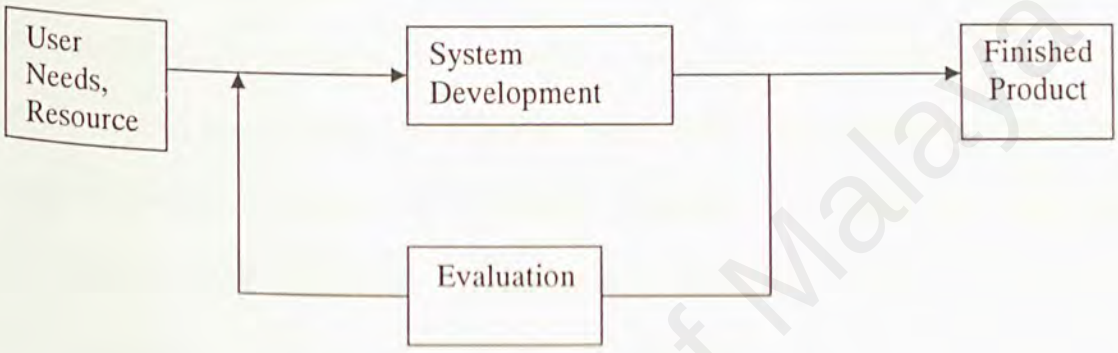


Figure 3.1: System Development Process Model

The process model that has been chosen to develop FaceRec system is **V Model**. The V Model demonstrates how the testing activities are related to analysis and design. As shown in (figure 3.2) below, coding forms the point of the V, with requirement analysis, system design and program design on the left, unit & integration testing, system testing, acceptance testing and operation & maintenance on the right.

Unit and integration testing addresses the correctness of the program. The V model suggests that unit and integration testing should also be used to verify the program design; during unit and integration testing, the developer should ensure that all aspects of the program design have been implemented correctly in the code. Similarly, system testing should verify the system design, making sure that all system design

aspects are correctly implemented. Acceptance testing will be conducted by the customer besides testing and validates by the developer. The customer validates the requirements by associating a testing step with each element of the specification; this type of testing checks to see that all requirements have been fully implemented before the system is accepted.

The model's linkage of the left side with the right side of the V implies that if problems are found during verification and validation, then the left side of the V can be re-executed to fix and improve the requirements, design, and code before the testing steps on the right side are reenacted (Pfleeger, Shari Lawrence. 2001). The focus of the waterfall is often documents and artifact whereas the focus of the V model is activity and correctness.

Below, coding The V Model is a variation of the waterfall model that demonstrates how the testing activities are related to analysis and design.

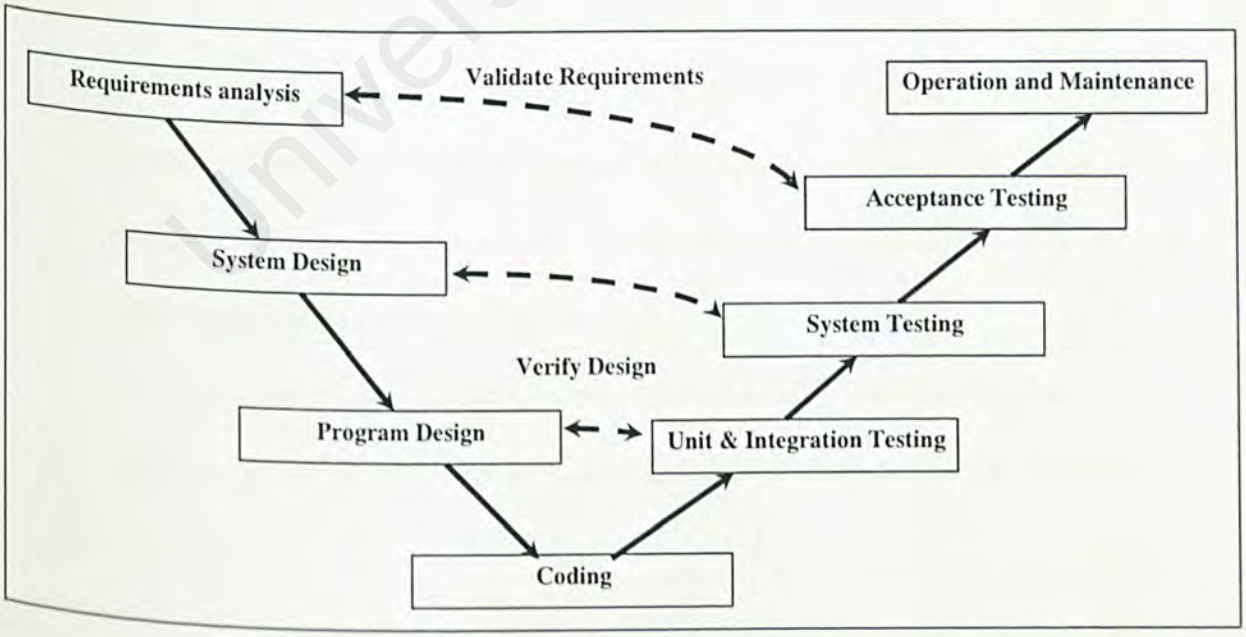


Figure 3.2 The V Model

The V Model consists of 8 stages that are depicted as cascading from another but in the form of a "V" shape. The development stages are:

- **Requirements Analysis**

  This stage includes analyzing the problem at hand and concludes with a complete specification of the desired external behavior of the system to be built; call functional description, functional requirements and specifications by others. Identifies problem, define information requirement and analyst system requirement (system analysis) are the processes in this stage.

  To developing the FaceRec, the requirements gathered would include the recognition and identification algorithms, functionality, performance, interface and constraints of the system. The requirements including functional and non-functional are documented and reviewed.

- **System Design**

  In this phase the established requirements, flowing from the requirements analysis and definition, are identified as software or hardware requirements. Besides, choosing the suitable system development methodology, tools and analysis on the system architecture, functional requirement and non-functional requirement are also made during this phase.

  The overall system architecture, content design, interface representation, data structure, conceptual design and technical of FaceRec are also required in this stage.

- **Program Design**

The information gathered is needed to accomplish the logical design of the system in this phase. It can be assessed for its quality before the code generation begins. Here, the design of overall system structure, flow charts and accurate data flow diagrams are planned so that the processes within the system are functional and correct. In this phase, algorithms are defined and document for each module in the FaceRec system that will be realized as code. It involves drafting out data flow diagrams that resembles the functionality of the system and its subsystem.

- **Coding**

This phase involves the transformation of the algorithms defined during previous phases into a computer understandable language. The program will be coded using selected programming languages and application development tools following the design specification.

During this phase, eigenfaces algorithms are apply into coding using Java programming language.

- **Unit and Integration Testing**

The purpose of unit testing is to ensure that each module behave accordingly to its specification defined during program design phase. It checks each module for the presence of bug.

This phase allow modifications to be made on the programs if any errors occur. Therefore I can test through all the units to ensure them meet their specifications and without any program failure.

- **System Testing**

All the units are combined and now the whole system is tested. The main objectives of System Testing are to detect the faults or errors in combined programs so that they can be corrected before the system is fully operational. When the combined programs are successfully tested the software product is finished. Otherwise, verification has to make on the System Design phase.

This phase tests the complete of FaceRec. All the program units will be combined together (integrated) and tested as a complete system to verify that the software requirements have been met. The system is delivered to the user after the testing process.

- **Acceptance Testing**

At this phase, the whole system can be delivered to the users. Users will go through the system to make sure it meets their requirements. Otherwise, validation of requirements needs to be done on the Requirements Analysis phase, to ensure that the system has implemented all the requirements in the specification. During this phase, FaceRec will be fully tested and able to put into practical use.

- **Operation & Maintenance**

Maintenance takes place after the system is put into practical use. Maintenance involves correcting errors that have gone undetected before (in earlier stages of the life cycle), improving the system implementation, adding performance or functional enhancements or making changes due to accommodate changes in the software external environment for overall project.

Although FaceRec is put into practical use, but it still need some simple changes to correct the undetected errors. All those changes can be made in this phase.

3.2.4 Justification of Methodology

The V Model was chosen as the development methodology for FaceRec system because:

- V Model forces analysis and planning before actions are taken. It forces a discipline process to avoid the pressures of writing code long before it is known what is to be built.
- It is one of the standard models used in this phase because it is simple to comprehend.
- It is a systematic model.
- The ability to verify the testing during the development stages is a much welcomed method.
- Iterative and reworks at any time.

3.3 Information Gathering Methods

Method of gathering information regarding a system is necessary in order to establish understanding of the state and future requirement on the system study and provide the groundwork for the system design.

There is no underlying standard or procedure to be followed strictly as each single project is unique and data-gathering may be vary to suit the needs of each particular project. However, there are a certain number of methods that are commonly used in gathering-information such as collecting hard data like written documents or reports, interviewing, using questionnaires, observation and sampling.

As for this project, due to cost and tight schedule constraints as well as the difficulties in finding and getting domain experts whom are willing to help, method such as interviewing becomes the intermediary who obtained the user's requirement from the real estate company. The main data sources for system analysis were written documents, reference books, observation and other sources from the Internet.

I. Internet Surfing and Research

Internet has been the main source of research especially in getting information regarding related software, developing software and method to implement for face recognition and identification technique. Through the search engine such as Google, Yahoo! and AltaVista, I can collect some idea from the similar research and case study about the system. Besides, I also get a lot of information on development tools and technologies, programming languages, database, project methodology, and others knowledge. The result from this research has been elaborated in detail in Chapter 2.

II. Books and References

Plenty of time spent in the document room in Faculty of Computer Science and Technology to refer to the previous theses done by other student. Book and references are used to get the information that needed to develop the system. This including information from information system references, development tools references, programming references and database references.

By referring to these theses, it helps in giving a rough idea on the requirements of the final project paper and the issues that are needed in developing the software. The format type of the report, organization of the heading and the content of the report can be referred from senior's thesis. The main theses that have become the main source are listed in the bibliography page.

III. Discussion with Supervisor

A discussion with supervisor has been practiced from time to time in order to get help and advices during the development of the project. He has consulted me on my project scope and his point of view has broadened my perspective on face recognition and identification tools. Through discussion with my supervisor, I have plenty of ideas regarding how the application should be and how to develop this application.

3.4 Conclusion on Tools and Technology

After all the technologies and algorithms have been reviewed and analyzed, the most suitable and appropriate tools and algorithm for developing the system are identified and selected. The tools to be selected include the development software as well as the entire platform on which the development of the project is occurred.

➢ **Selected algorithms**

The **eigenfaces** algorithm was chosen as the algorithm to implement the face recognition system due to the advantages below:-

- **Fast and relatively simple.**

  It is fast and easier to implement compare to others techniques such as Neural Networks and template based method.

- **Insensitivity to small change.**

  It relative insensitivity to small or gradual changes in the face image and also proof to work well under constrained environment.

- **Recognition speed is faster that other algorithms.**

  Because the eigenvectors only need to be computed once and are easy to find, this is very fast compared to other methods.

- **Simple preprocessing**

  Other methods require a significant amount of preprocessing, where this does not in Eigenfaces. (i.e. calculating the edges within an image)

- **Work well if the background is not complex.**

  Eigenfaces are accurate and can work well if the background is not complex and small lighting variation.

➢ **Selected Platform**

Platform/Operating System that referred here means operating system that operate underlying the application or system being developing, instead of operating system used in the developing phase. Operating system that fall into the consideration domain with be evaluated from the perspective of end-users and developers, which they are available in the market, such as Microsoft Windows 98, Microsoft Windows NT Server 4.0, Microsoft Windows XP, Mac OS, UNIX and LINUX.

Review has been done in purpose of choosing a suitable platform/operating system for this project. After all the consideration, I have decided to choose **Windows 2000 Professional** as my system platform, as it is the most suitable operating system that are able to support all the development tools I choose to bring this project into success.

• Usability

Firstly, we consider the element of usability, which mean how easily the operating system can be used.

From the view of the installation of the operating system itself, user need a simple, straight forward, user friendly and wizard guided operating system installation. In survey done, we found that UNIX, LINUX and Server 2000 are more difficult to

install and configure. Unlike Windows 2000 Professional, UNIX or LINUX is not an end-user-oriented operating system known for its user friendliness, the user need to have certain level of IT knowledge fitness in order to get them installed well. In a production environment, this is not very desirable because valuable time will be wasted on learning intricate details of various applications.

• Technological Ability

Windows 2000 Professional is built on Windows NT technology which highly supportive in developing and machine application. Due to its usability, there are uncountable of software are developed exclusive for it. It also promises a full compatibility to the product from the same vendor – Microsoft Corporations, the biggest software developing organization in the world. In other words, if the UNIX or LINUX is used, developer will have to facing incompatibility in using some (most) of the application, especially developing software, such as Microsoft Visual C++.

Most of the software, such as APIs, are developed exclusively for Microsoft Windows operating system, while they were originally developed for other system such as UNIX and Mac OS, but false for the reverse.

• Performance

The advancements made throughout Windows 2000 Professional are accentuated by the operating system's speed. As research done for applications, with 64 MB of RAM, Windows 2000 Professional was 32 percent faster than Windows 95 and 27 percent faster than Windows 98. It is also significantly faster than Windows NT 4.0 on configurations with 32 MB of RAM.

• Maintenance

Maintenance is a very critical in a Software Developing Life Cycle (SDLC), because of the strong background, the technical support is provided well by its vendor. All the documentation of Windows 2000 Professional is documented well and freely available online. Unlike Windows 2000 Professional, UNIX and LINUX are free and opened software, no specific party is responsible for the technical support. There is no standard version of LINUX and UNIX, thus, it would be a nightmare for the developers as searching for technical and maintenance information.

• Cost

Among the entire platform, the pricing of Windows 2000 Professional is not so expensive, even some may say some other operating system like UNIX and LINUX are well known as license free Operating System. Though, cost is only evaluated from the license paid. Besides the cost for buying the license, for this operating system, there is cost of setting up the operating system. For UNIX, which have cryptic user interfaces are hard to manage and give way to high administration costs. In this trade-off, Windows 2000 Professional still stands up to be cost-effective.

• Reliability

From the experience of the developer and the information collected from the survey done, Windows 2000 Professional is the most stable and reliable operating system among the product of Microsoft has ever produced compare to Windows 98, Windows 98SE, Windows NT 4.0 and Windows ME. Undeniable, UNIX and LINUX are very reliable as well, but in the previous evaluation such as maintenance and cost effective aspects, they has been removed from the consideration list.

• **Hardware**

Windows 2000 Professional take advantage of new hardware devices, such as those with universal serial bus (USB) connections. In addition, support for existing hardware makes Windows 2000 Professional ideal for developer especially using external devices such as handy drive.

For all the **advantages** stated above, Window 2000 Professional has been chosen as the development platform.

➢ **Selected Programming Language**

Java is a portable object-oriented computing language that has brought its benefits into mainstream computing. One of its main benefits is portability; compiled Java programs do not have to be re-compiled to run on different platforms. Others include its object orientation and its large library of APIs (libraries) that are almost as portable as the language itself.

The following list of Java's attributes:

- Simple. Java's developers deliberately left out many of the unnecessary features of other high-level programming languages. For example, Java does not support pointer math, implicit type casting, structures or unions, operator overloading, templates, header files, or multiple inheritance.

- Object-oriented. Just like C++, Java uses classes to organize code into logical modules. At runtime, a program creates objects from the classes. Java classes can

inherit from other classes, but multiple inheritances, wherein a class inherits methods and fields from more than one class, are not allowed.

- Statically typed. All objects used in a program must be declared before they are used. This enables the Java compiler to locate and report type conflicts.

- Compiled. Before you can run a program written in the Java language, the program must be compiled by the Java compiler. The compilation results in a "byte-code" file that, while similar to a machine-code file, can be executed under any operating system that has a Java interpreter. This interpreter reads in the byte-code file and translates the byte-code commands into machine-language commands that can be directly executed by the machine that's running the Java program. You could say, then, that Java is both a compiled and interpreted language.

- Multi-threaded. Java programs can contain multiple threads of execution, which enables programs to handle several tasks concurrently. For example, a multi-threaded program can render an image on the screen in one thread while continuing to accept keyboard input from the user in the main thread. All applications have at least one thread, which represents the program's main path of execution.

- Garbage collected. Java programs do their own garbage collection, which means that programs are not required to delete objects that they allocate in memory. This relieves programmers of virtually all memory-management problems.

- Robust. Because the Java interpreter checks all system access performed within a program, Java programs cannot crash the system. Instead, when a serious error is discovered, Java programs create an exception. This exception can be captured and managed by the program without any risk of bringing down the system.

- Secure. The Java system not only verifies all memory access but also ensures that no viruses are hitching a ride with a running applet. Because pointers are not supported by the Java language, programs cannot gain access to areas of the system for which they have no authorization.

- Extensible. Java programs support native methods, which are functions written in another language, usually C++. Support for native methods enables programmers to write functions that may execute faster than the equivalent functions written in Java. Native methods are dynamically linked to the Java program; that is, they are associated with the program at runtime. As the Java language is further refined for speed, native methods will probably be unnecessary.

Because of the above features and other features that Java supports, it seems to be the best option of a suitable programming language to implement and developed the FaceRec system.

➢ **Selected Development Tools**

JCreator 2.5 is a powerful IDE for Java. JCreator provides the user with a wide range of functionality such as: Project management, project templates, code-completion,

debugger interface, editor with syntax highlighting, wizards and a fully customizable user interface.

Advantages of JCreator 2.5:

- **Improved Reliability** - Can directly compile or run the Java program without activating the main document first. JCreator will automatically find the file with the main method or the html file holding the java applet, and then it will start the appropriate tool.

- **Increased Performance** - JCreator is written entirely in C++, which makes it fast and efficient compared to the Java based editors/IDEs.

- **Developer Productivity** - Professionally designed to meet windows interface guidelines, JCreator users can expect a familiar and intuitive user interface.

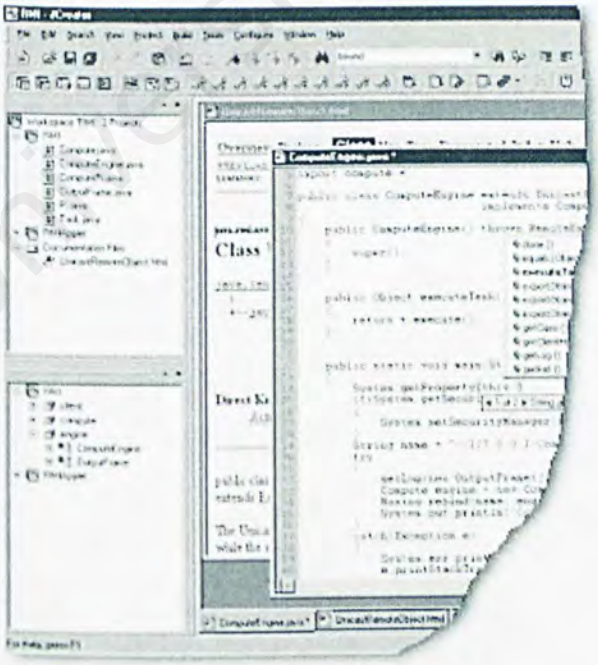


Figure 3.3 JCreator 2.5

➢ **Selected Database Server**

Microsoft Access 2000 works well with databases of any size. It contains all the user-friendly features, works more efficiently and has the ability of handling thousands of photos without affecting performance. Therefore, Microsoft Access 2000 will be chosen to act as the database management software for the development of FaceRec.

Advantages of Microsoft Access 2000:

- It is able to handle larger amount of photos during recognition process.
- Most viable solution to accommodate the vast storage requirements.
- Scalability and high performance

3.5 Chapter Summary

In order to produce a more efficient and better quality system, this project will be develop through several stages, by implementing the V Model methodology. The first two stages, which are requirement analysis and system design have been completed in the first semester. While for system coding, system testing, and system operation, will be implement in the second semester.

The V model approach is selected for the development of FaceRec system because a standalone recognition system needs a well organized and structured planning of system design and implementation. This method is adopted because simple and ease of implementing. In addition, the ability to verify the testing during the development stages is a much welcomed method.

Through the system development life cycle, system methodology is adopted to understand the current problem situation. Careful analysis and detail research about the existing system and method to be used has been conducted to determine the feasibility of the system and what is required of it. The system requirements are identified, translated into design and finally implemented via coding. The finished system is evaluated to meet the system objectives and requirements specification.

Chapter 4: System Analysis

CHAPTER 4: SYSTEM ANALYSIS

4.1 Introduction

System Design is the evaluation of alternative solutions and specifications of detailed computer-based solution. Before the system design can begin, the type of software that would be used in order to implement the system should be first determined. This is in order to know whether the software that would use could support the design plans of the system and also help in designing the system.

In this chapter, the software that would be used to implement the system would be explained briefly, before going into a detailed explanation of the design of the functionality of the eigenfaces approach in face recognition.

4.1.1 Functional Requirements

The functional requirements for a system describe the functionality or services that the system is expected to provide and how the system should react to particular inputs and how the system should behave in particular situations.

Functional requirements for a software system may be expressed in a number of different ways. Here are a number of functional requirements for FaceRec system which are divided into several modules:

a) **Photo manipulation module**

- **Photo capturing**

The FaceRec system takes as input a training set of face images (to be compared) and a test face image (to be recognized). Webcam is used to capture the frontal face image of a person as the training set or test image. The photo captured must be clear in order for the system to perform well.

b) **Database manipulation module**

- **Adding a new record**

The administrator has the power to add a person to the training set together with all the record of the person. The face image that captured by the web camera will be added into the database with all the details of the person.

- **Edit & select an existing record**

The administrator has the power to edit the record of a person in the existing database. This can be done by select the **Search** command for an existing record to be displayed and edit the data of the person to update the person details.

- **Deleting an existing record**

The administrator has the power to remove a person from the training set and the person deleted cannot be recognize and identify by the system. The entire data of the person in the existing database would be deleted together with the photo which does not want to be used anymore.

c) **Face recognition and identification module**

- **Perform recognition & identification.**

The recognition and identification module is responsible for the face recognition process. Once the person face image that want to be recognize is captured using the web camera and done some image processing, the recognition process can be done using eigenfaces approach. To perform the recognition and identification process, it would take some time in performing the process. Once the recognition and identification complete, a message would be displayed stating whether a match is found. The data that identifies the person would be displayed, along with the picture that used for comparison.

4.1.2 Non-Functional Requirements

In order to ensure the quality of the system produced, certain quality factors must be conformed. Non-functional specifications are the constraints under which a system must operate and the standards which must be met by the delivered system (Sommerwille, 2001). The non-functional requirements have been considered for FaceRec system are as shown below:-.

- User-friendliness
- Usability
- Flexibility and Scalability
- Reliability
- Efficiency
- Manageability
- Maintainability
- Correctness
- Security

a. User-Friendliness

User interfaces design creates an effective communication medium between a human and a computer. Therefore, it is very important to make sure that the interfaces fulfill user-friendliness so that it would not cause trouble to users. The Golden Rules (Mandel, 1997) coins three rules:

- **Place the user in control**

  This will define interaction modes in a way that does not force a user into unnecessary or undesired actions. Besides, it also provides flexible

interaction for different users for instance via mouse movement and keyboard commands.

- **Reduce the user's memory load**

  One of the principles that enable an interface to reduce the user's memory load is by reducing demand on short-term memory. The interface should be designed to reduce the requirements to remember past actions and results.

- **Make the interface consistent**

  The interface design should apply to consistent fashion where all visual information must be organized according to a design standard that is maintaining throughout all screen displays. Apart from that, input mechanisms are constrained to a limited set that are used consistently throughout the application. Lastly, mechanisms for navigating from task to task are consistently defined and implemented.

b. Usability

The system must prove to be useful and beneficial to the potential users. It must be able to provide the information and services that are needed quickly, clearly and minimum fuss. All the function and services provided must be able to carry out the result. The should not in any way hinder or limit the user from carrying out the desired operation, such as retrieving data from the databases, as long as they are authorized to do so.

c. Flexibility and Scalability

The system should have the capability to take advantages of new technologies and resources and can be implemented in multi-platform. It should be able to enhance

for any requirement. Beside that, system should be having the ability to migrate or adapt to machines with greater or lesser power whether as server side or client side.

d. Reliability

Reliability is the extent to which a program can be expected to perform its intended function with required precision (Pressman, 2001). When the system is reliable, all the functions will be executed precisely and smoothly. High reliability will promote user confidence in the system and will encourage them to use the system more frequently and they trust that the system will meet their needs efficiently. The author will carry out thorough testing to ensure that the system reliability is not compromised in any way, at any time of the implementation or operation.

e. Efficiency

Undeniable, efficiency is the main key for implementing the new meetings management system. Efficiency is understood as the ability of a process procedure to be called or accessed unlimitedly to produce similar performance outcomes at an acceptable or credible speed (Sommerwille, 2001). Those efficiencies are referred to the consumption of the response and processing time performance, loading speed during the run time. The lesser it uses, the higher efficiency.

f. Manageability

The hardware and software applications of the system must be proficient to be managed effortlessly and easily operated. This is crucial to ensure that future maintenance and enhancements can be done on the system without having major problems.

g. Maintainability

System maintenance accounts would require more effort if the system is not design according to good programming practices. Maintainability is the ease with which a program can be corrected if an error is encountered, adapted if its environment changes, or enhanced if the customer desires a change in requirements (Pressman, 2001). As the to-be-developed FaceRec will be built by using Java which has to be developed with object-oriented concept, therefore, it is strongly believed that bugs or system faults can be detected and fixed in the shortest time. This is because object-oriented design makes sure that each class or object will only strictly handle one particular task or functionality.

h. Correctness

A program or system must operate correctly or it provides little value to its users. Correctness is the degree to which the software performs its required function. To ensure this application quality, lots of testing and trial-and-errors will be carried out.

i. Security

The proposed system has also security measures to minimize the risk of data exposure to unauthorized people.

4.2 Technology proposed

➢ **What is eigenfaces?**

Eigenfaces are a set of characteristic feature images. They are the principal components of the initial training set of face images, and can be described as the "basic face images". Eigenfaces are calculated by using a method called the principal component analysis, and they can be seen as the "basic faces" that best describe the distribution of face images within the entire face space.

Using this analysis, the principal component of the distribution of faces is first calculated. These principal components, also known as eigenvectors of the covariance matrix of the set of face images, together characterize the variation between the face images. With the image location contributing (more or less) to each eigenvectors, the eigenvectors can then be displayed as something similar to a ghostly face. These ghostly images are called the eigenfaces.

➢ How?

In face recognition using eigenfaces, each face images can be reconstructed by weighted sums of a small collection of the characteristic images, or eigenfaces, Recognition is then performed by comparing the feature weights needed to approximately reconstruct the face image, with the weights associated with the known individuals.

In summary, recognition is performed by:-

i) First acquiring the training set of face images.

ii) Calculating the eigenfaces from this training set, which will define the face space.

iii) Calculating a set of weights based on the training set of face images and the calculated eigenfaces by projecting the face images onto each of the eigenfaces.

iv) Acquiring the test image that needs to be recognized.

v) Calculating a set of weights based on the test image and the calculated eigenfaces, in a similar way as in (iii).

vi) Finding the weight that is nearest (or approximately closes) to that of the known individuals.

The weight of the known individuals that is approximately close to that of the test image will be chosen as the best match for recognition.

➢ **Algorithm**

In face recognition using the eigenfaces approach, two main steps exist: representation and matching. Briefly, the first step that needs to be done is in calculating the eigenfaces. In calculating these eigenfaces, the representation of the face images are important in order to allow mathematical manipulations of these images. Once this calculation is done, the weights attributed to each image in connection to the eigenfaces are calculated. With these calculated weight, the recognition process can proceed by incorporating the concept of pattern matching. In this section, the mathematical algorithms involved in handling these two problems are look into in more detail.

✓ Calculating the eigenfaces

Let a face image I (x,y), either be:-

- A two-dimensional N_1 by N_2 array in the image.
- A vector of dimensional N_1 by N_2.

where N_1 x N_2 is the number of pixels in the image.

And let the training set of face images be,

$$\Gamma_1, \Gamma_2, \Gamma_3, \ldots, \Gamma_M$$

where M is the number of images I the training set.

The average face can then be calculated using the formula:-

Average image, (Ψ)

$$\Psi = \frac{1}{M} \sum_{k=1}^{M} \Gamma_k$$

Figure 4.1 Average Image (Ψ)

Each face differs from the average face by the vector,

Deviated images, (Φ_k)

$$\Phi_k = \Gamma_k - \Psi$$

where k = 1, 2, 3... M

Covariance matrix, (C)

$$C = \frac{1}{M}\sum_{k=1}^{M}\Phi_k\Phi_k^T = AA^T$$

where the matrix $A = |\Phi_1\Phi_2\Phi_3 \ldots \Phi_M|$

After the covariance matrix is formed, the eigenvectors can be obtained by the eigenvectors definition:

$$C\,\mu_i = \lambda_i\,\mu_i$$

The matrix C is N^2 by N^2. To determine the N^2 – dimensional eigenvectors and eigenvalues, direct solving is computationally expensive. A feasible method is existed by constructing a M by M matrix. Consider the eigenvectors v_i of A^TA,

$$A^TA\,v_i = u_i\,v_i$$

Now, pre-multiply both sides by A,

$$A\,A^TA\,v_i = u_iA\,v_i$$

Here, we could see that Av_i are the eigenvectors of C ($=AA^T$). Therefore, instead of solving the covariance matrix, C directly, first we construct the A^TA and obtain its

eigenvectors and eigenvalues. Then pre-multiplying the eigenvectors by A results the eigenvectors of the covariance matrix. As we could see, the eigenvalues of the covariance matrix are the eigenvalues of A^TA, which are already we find the eigenvectors of A^TA.

Figure 4.2 Face Space of the Training Set

✓ **Matching**

Once the eigenfaces for the training set of face images have been calculated, the next step in implementing the eigenfaces approach in face recognition is finding the solution to the problem of matching. The approach to solving this problem is in pattern matching or in a minimal distance classifier.

An image is transformed into its eigenfaces components, or in other words, projected onto the face space according to the formula:

$$w_k = u_k^T(\Gamma - \Psi)$$

for k = 1,2,... ... ,M where Γ is the image, Ψ is the average face, w_k is the weight of the corresponding eigenvector u_k and M is the number of the best eigenvectors chosen.

The weights from a vector $\Omega^T = [w_1, w_2, \ldots\ldots w_M]$ that describes the contribution of each eigenvector in representing the image, treating the eigenvectors as a basis set for face images. The vector may then be used in a standard pattern recognition algorithm to find which predefined face best describes the images. The simplest way to do this is to find the class k that minimizes the Euclidian distance,

$$\varepsilon = \| (\Omega - \Omega_k) \|$$

where Ω_k is a vector describing the kth face class.

A face is classified as belonging to class k when the minimum ε_{min} is below some chosen threshold $\emptyset$. Otherwise, the face is classified as "unknown". This distance threshold $\emptyset$ can be calculated by computing half the largest distance between any two face classes, as shown in the formula below;-

$$\emptyset = 1/2 \max_{j,k} \{ \| \Omega_j - \Omega_k \| \}; \; j, k = 1, \ldots, N_c$$

There are several steps in solving the pattern-matching problem:-

a) Calculating a set of weights for the training set, w_k

Using the eigenfaces calculated previously, a set of weights based on the training set of face images can be calculated by projecting each face images onto each eigenfaces in the face space. This set of weights describes the contribution of each eigenfaces in

representing the input face image and would be stored in a vector. This vector would be used in the recognition process.

This set of weights could be calculated b first finding the matrix transpose of each eigenfaces, and then performing matrix multiplication between the matrix transpose calculated and the matrix of the image deviation. This calculation is performed for each image deviation for each image in the training set.

b) Acquiring a test face image to be recognized and calculating its weights

The recognition process will actually begin once a test face image is acquired for recognition. As with images in the training set, the test image would also be represented first in a matrix format (using similar method to those mentioned in the previous sub-section) so that it could manipulated for matrix calculation. Once it has been transformed into a one dimensional matrix, its weight would be calculated by projecting the image onto each eigenfaces in the training set. This can be done using the same calculation as was in (a).

c) Find the nearest match based on the set of weights calculated from (a) and (b), ε_{min}

Once both set of weights for the training set and the test image has been calculated, the pattern recognition process can begin. Both set of weights would be compared and the nearest match would be taken as the identified face.

This can be done by performing subtraction on the set of weights for each image in the training set with the set of weights for the test image. The smallest value calculated will be the nearest match.

If ε_{min} is less a threshold ø, the testing image can be identified as the identity of the matched face. If ε_{min} is larger than the threshold ø, the testing image can be claimed to be an unknown identity.

4.3 Hardware and Software Consideration

This section contains the introduction to various the hardware and software components used in the development of this system. Also included are the explanations and justifications of choosing the hardware and software that best suits this project.

4.3.1 Hardware Requirements

In this section, the hardware requirement will be divided into two aspects, which is:

- System requirements for developer
- System requirement for user

4.3.1.1 System Requirements for Developer

The hardware and computer system that are needed to develop the Face Recognition and Identification (FaceRec) should have a very fast processor and huge RAM because the processing time is important.

- **Developers**

Minimum Requirements	
Processor	Intel Pentium II 333 MHz or higher processor.
Operating System	Windows 2000 Professional
Memory	128 MB RAM (higher preferred)
Hard Disk	1 GB of Hard disk space.
Drive	CD-ROM drive

Display	VGA or higher-resolution monitor required. Super VGA monitor recommended.
Mouse	Microsoft Mouse or compatible pointing device.
Web Camera	Webcam for capturing image (high resolution preferred)

Table 4.1 Hardware Requirements for Developer

4.3.1.2 System Requirements for User

These are the system requirements for the users. As an overall, a better system should have the following hardware requirement for the FaceRec System to perform well.

- **Users**

Minimum Requirements	
Processor	Intel Pentium II 333 MHz or higher processor.
Operating System	Windows 2000 Professional
Memory	128 MB RAM (higher preferred)
Hard Disk	1 GB of Hard disk space.
Drive	CD-ROM drive
Display	VGA or higher-resolution monitor required. Super VGA monitor recommended.
Mouse	Microsoft Mouse or compatible pointing device.
Web Camera	Webcam for capturing image (high resolution preferred)

Table 4.2 Hardware Requirements for Users

4.3.2 Software Requirements

The software requirement is the software that required to supports the goal stated in the developed system. According to all the research that has been made, developing a Face Recognition program requires software that can support object-oriented programming. Besides, Microsoft Access 2000 is using as database to store all images and detail of the person. Operating system such as Windows 2000 is important in building blocks within any technical solution. While development tools such as JCreator 2.5 is important to develop program.

- Windows 2000
- Java
- Microsoft Access 2000

4.3.3 Others

1) Testing images from University of Surrey, Olivetti-Oracle Research Lab. (AT&T Laboratories Cambridge, 1999).
   - Act as an image obtained from a web camera for development and testing.
2) JAMA (Java Matrix Package)
   - JAMA is a basic linear algebra package for Java. It provides user-level classes for constructing and manipulating real, dense matrices implementation, which is useful during implement Eigenfaces algorithms.

3) JMF (Java Media Framework API)

- JMF enables video and other time-based media to be added to applications and built on Java technology. It is need when implementing web camera into the system.

Chapter 5: System Design

CHAPTER 5: SYSTEM DESIGN

5.1 Introduction

A system design is a description of the structure of the software to be implemented, the data which is part of the system, the interfaces between system components and sometimes the algorithm/method used. Designers do not arrive at a finished design immediately but develop the design iteratively through a number of different versions. The design process involves adding formality and detail as the design is developed, with constant backtracking to correct earlier designs.

The design process may involve developing several models of the system at different level of abstraction. As a design is decomposed, errors and omissions in earlier stages are discovered. These feedbacks to allow earlier design models to be improved.

As the design process continues, these specifications become more detailed. The final results of the process are precise specifications of the algorithms/method and data structures to be implemented. The specific design process activities are:

- System Functionality Design
- Database Design
- User Interface Design

5.2 System Functionality Design

5.2.1 System Structure Charts

The objective of system structure chart is to show how the modules in FaceRec are related to each other.

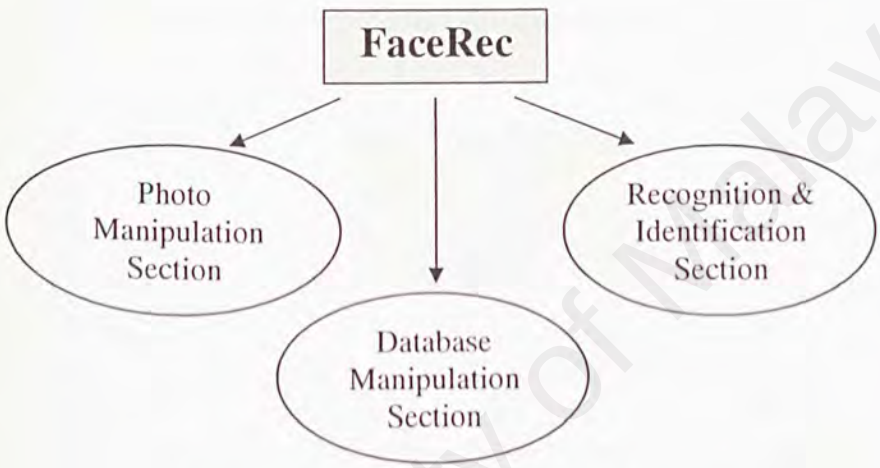


Figure 5.1: Structure Chart for FaceRec

FaceRec consists of three major parts, which are the database manipulation section, photo manipulation section and recognition and identification section. Basically, photo manipulation section is important where the photo need to be captured using webcam and undergo some process before the image is pass into the system as training set image or test image. The database manipulation section allows the administrator to add a new record, edit or delete existing record of a person into the system. For the recognition and identification section, the system will perform recognition and identification using eigenfaces approach.

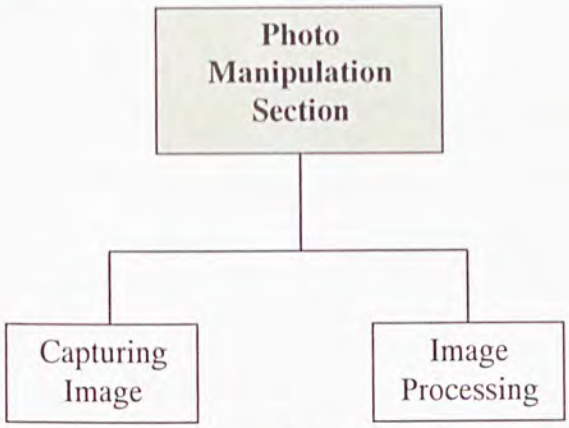


Figure 5.2 Structure Chart for Photo Manipulation Section

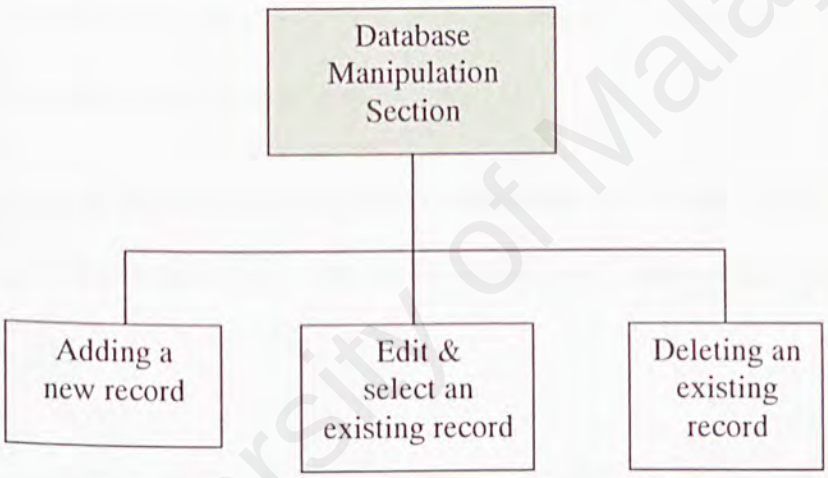


Figure 5.3 Structure Chart for Database Manipulation Section

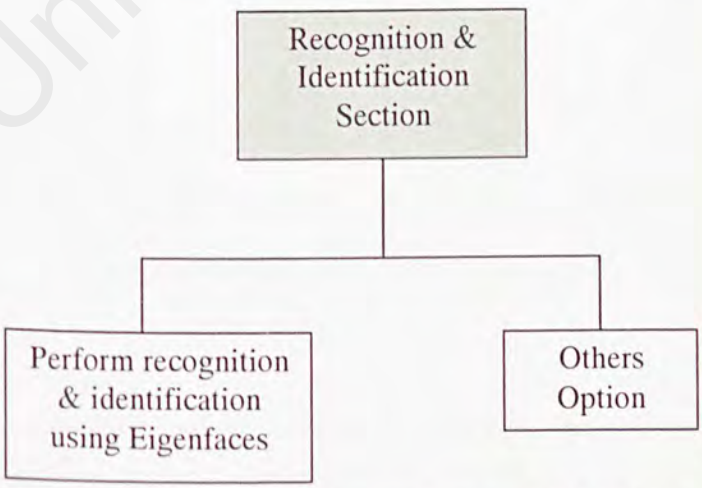


Figure 5.4 Structure Chart for Recognition and Identification Section

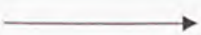
5.2.2 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) is a method used to graphically characterize data processes and flows in FaceRec. DFD will depict the overview of the system inputs, process and outputs.

The advantages of using DFD are:

- Further understanding of the interrelatedness of modules and sub modules of FaceRec.
- Analysis of a proposed system to determine if the necessary data and processes have been defined.

DFD is easy to be understood as it has symbols that specify the physical aspects of implementation. There four basic symbols in DFD: entity, flow of data, process and data stores (see Table 5.1).

Symbols	Attribute
	Entity
	Flow of Data

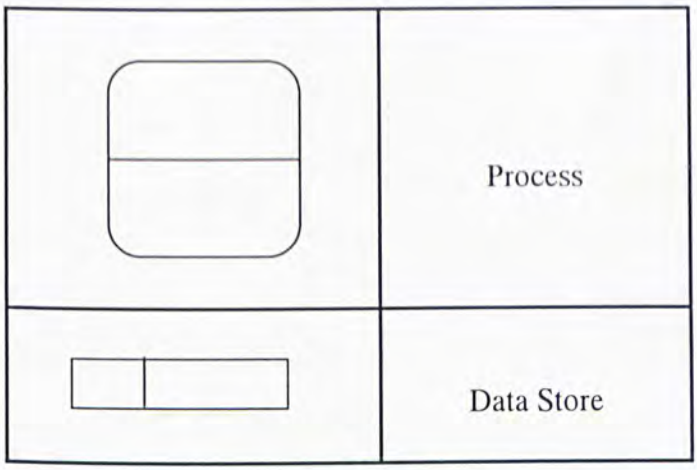


Table 5.1 DFD Symbols

The convention, which is used to design DFD are based on the work by C.Gane and T.Sarson. The data flow is conceptualized with a top-down perspective. So, the Context Level Diagram will be drawn, followed by the Diagram 0. Diagram 0 is an overview process of all the major modules in FaceRec that includes all the data stores, entities and process involved.

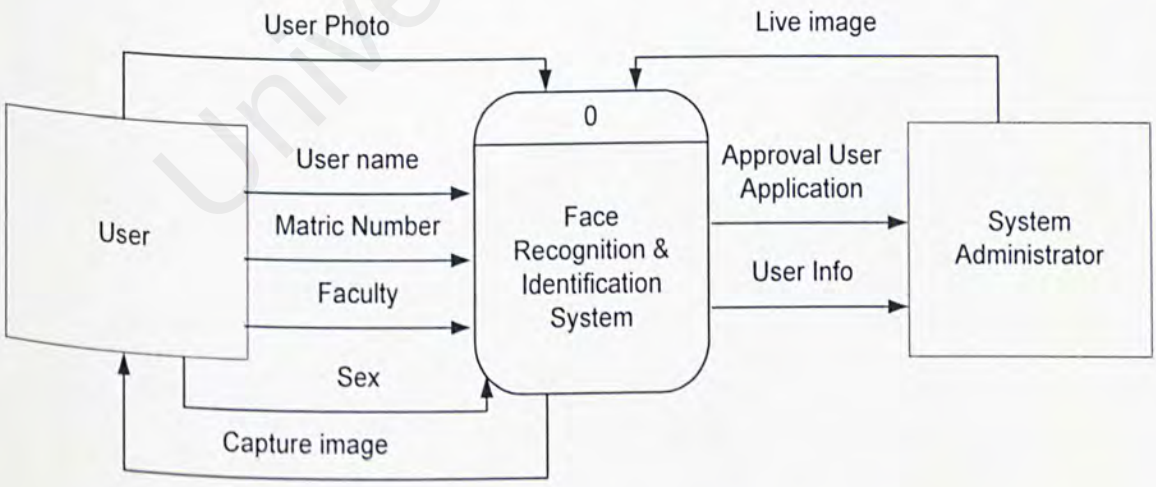


Figure 5.5 Context Level Diagram of FaceRec

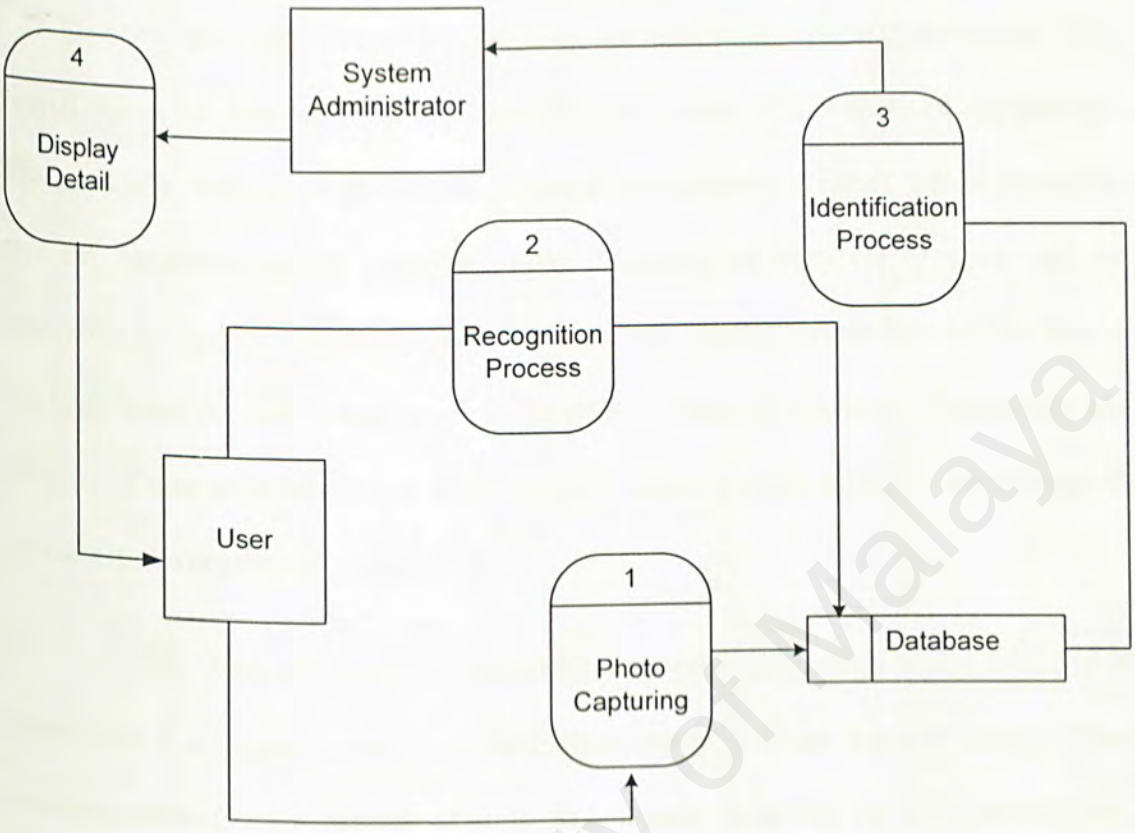


Figure 5.6 Diagram 0 of FaceRec

5.3 Database Design

Data storage is considered by some to be the heart of an information system. It is a central source of data meant to be shared by many users for a variety of applications. The heart of a database is the DBMS (database management system), which allows the creation, modification and updating of the database; the retrieval of data; and the generation of reports. The main objective of database design is to make sure that data is available when the user wants to use it. Apart from that, the accuracy, consistency and integrity of data must be assured from time to time, to provide efficient data storage as well as efficient updating and retrieval.

In 1976, Peter Chen had introduced the use of the entity-relationship model (E-R Model). An E-R diagram contains many entities, many different types of relations, and numerous attributes. The benefits of Entity Relationship modeling are mentioned below:

i. Databases need to be designed and entity relationship (ER) modeling is an aid to design.

ii. An ER model is a graphical representation of the system and is a high-level conceptual data model.

iii. Supports a user's perception of data and is independent of the particular DBMS and hardware platform.

5.3.1 Data Dictionary

Data dictionary or metadata can be defined as descriptions of the database structure and contents. Data dictionary defines the field, field type and descriptions of each table.

In FaceRec, one database had been defined namely FaceRec and contained 1 tables, which are FaceRec_person.

Database Name : **FaceRec**

Table name : **FaceRec_person (D1)**

Field Name	Data Type	Length	Note
person_id	number	10	ID of person
person_name	varchar2	255	Name of the person
person_department	varchar2	255	Department of the person
person_sex	varchar2	255	

Table 5.2 FaceRec Data Dictionary

5.4 User Interface Design

The interface stands as the representation of the system and by reflection, the developer competence as a system analyst. A good user interface will allow people to understand the domain problem to work together with the application without needing to read the manuals or receiving any training. At worst, users will simply refuse to use the software system irrespective of its functionality. If information is presented in a confusing or misleading way, users may misunderstand the meaning of information. They may initiate a sequence of actions that corrupt data or even cause catastrophic system failure.

Although text-based interfaces are still widely used, especially in legacy systems, computer users now expect application systems to have some form of graphical user interface (GUI). Interfaces are designed with following objectives in mind.

- Increase the speed of data entry and reduce errors.
- Relatively easy to learn and use.
- Provide appropriate feedback to users from the system.
- Allow users to access the system in a way that is congruent with their individual needs.
- Fast, full screen interaction is possible with immediate access to anywhere on the screen.
- Reduce cost of supporting, training and interactive.

5.4.1 User Interface for FaceRec

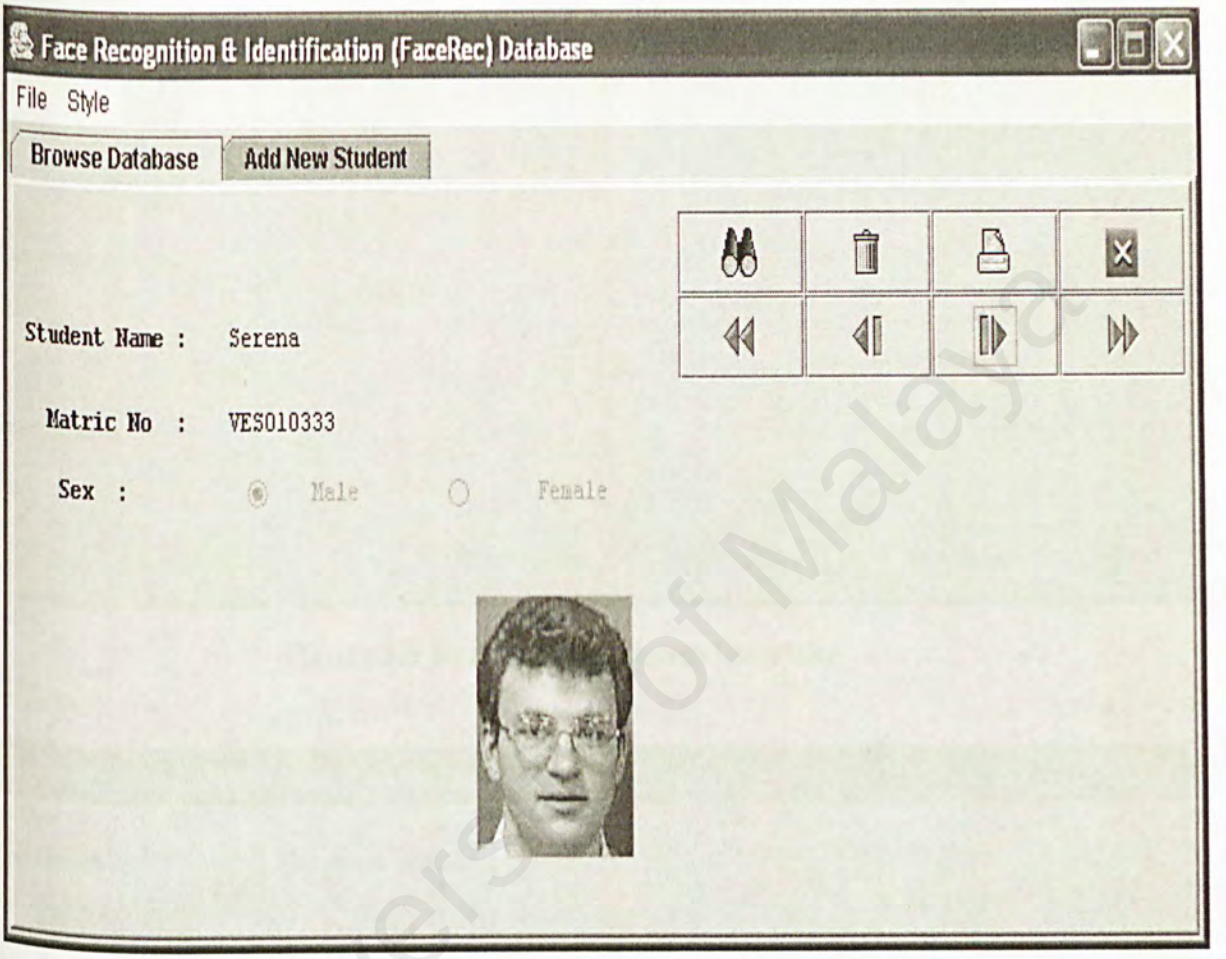


Figure 5.7 FaceRec Student Database Interface

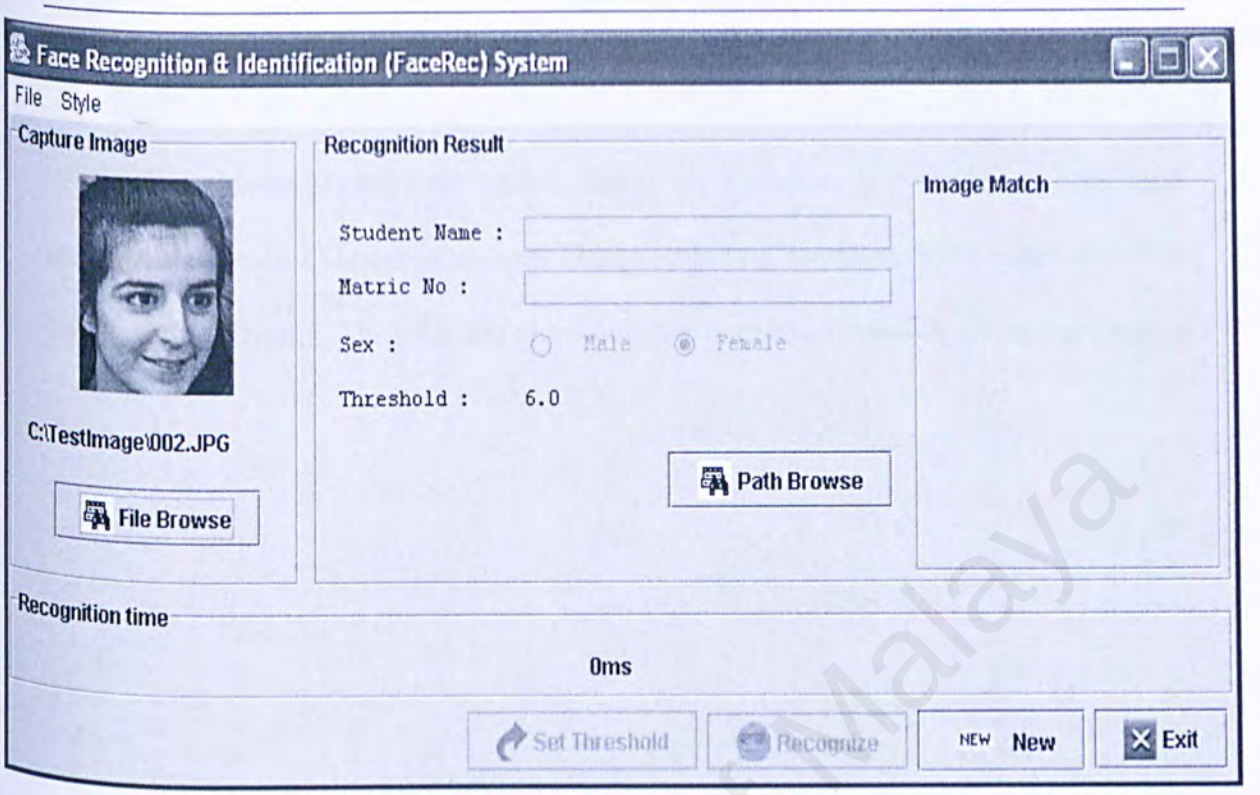


Figure 5.8 Recognition Process Interface

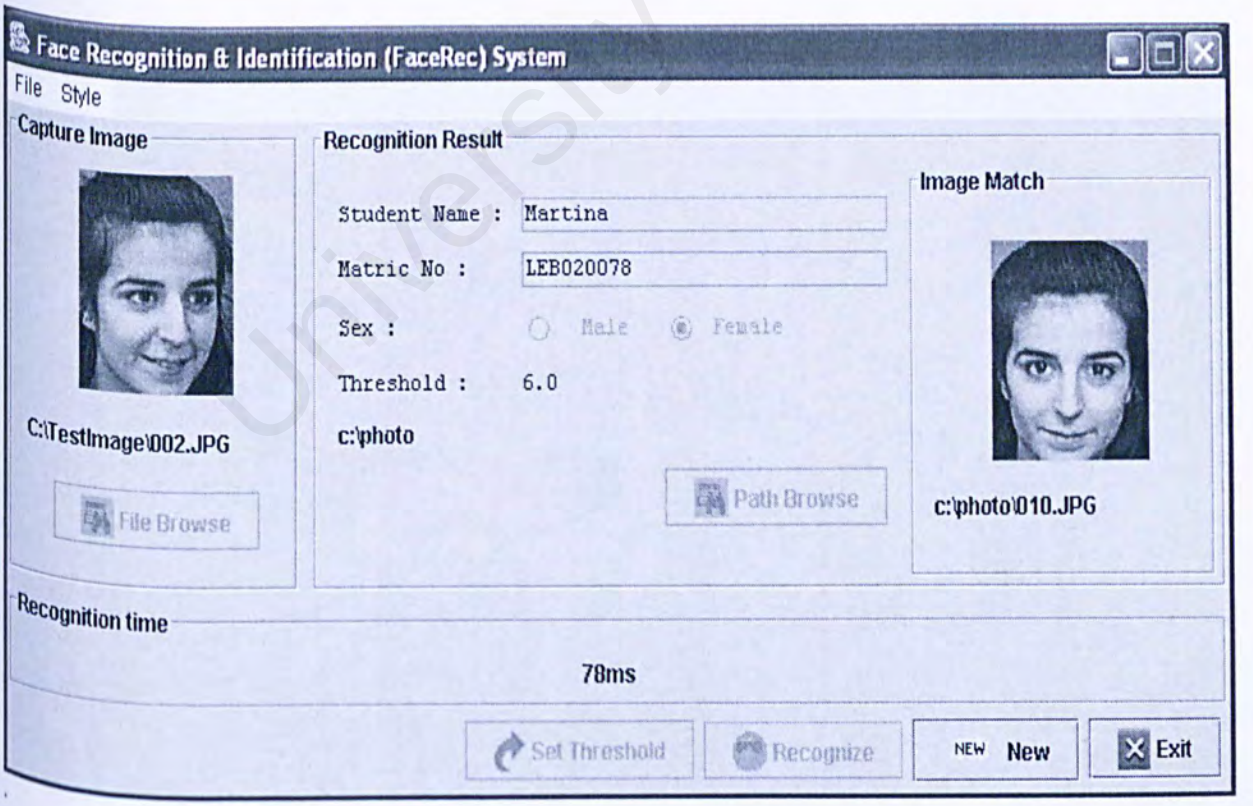


Figure 5.9 Recognition Complete Interface

5.5 Chapter Summary

This chapter presents the system design for FaceRec. It given the overview of System functionality Design, Database Design, Module Functionality Design and also User Interface Design. The GUI will show how the FaceRec system expected outcome.

Chapter 6: System Implementation

CHAPTER 6: SYSTEM IMPLEMENTATION

6.1 Introduction

System implementation is a phase integrating the designed modules or functions to develop a system based on the given requirements. It is the process takes place after the system design phase. This phase describes how the initial and revised process design put into the real work. Therefore, system development, coding methodology and development tools are included in this phase.

6.2 Development Environment

The development environment of FaceRec consists of software and hardware configuration. Using the suitable hardware and software will help in speed up the system development. The hardware and software tools that used to develop and document the system will be discuss as below.

6.2.1 Hardware Requirement

The hardware used to develop the system are as listed:

- Pentium II 233 Mhz Processor
- 128MB RAM
- 10 GB hard disk
- 52X CD-ROM drive
- Other standard computer peripherals

6.2.2 Software configuration

The software tools that have been used to develop FaceRec are:

Software	Usage	Description
Microsoft Windows 2000 Professional	System Requirement	Operating System
Java	System Development	System Design and Coding
Microsoft Access 2000	System Development	System Database
Microsoft Word	System Development	Project Documentation

Table 6.1 Software Tools

6.3 Module Implementation

The development of FaceRec was implemented module-by-module. A prototype was created and tested for each of this module. Testing and improving the prototype is the repeated until a precise and functional prototype is created. Lastly all of the prototypes will be combined and integrated to form the final application.

6.3.1 Student Database Module (Database.java)

The first module to implement is the Student Database Module. This database module is to add, delete, search and also update student details including name, matric number and gender.

Sample coding:

```
// Accessing the Student Database

void accessDB(String sql,int indexYouWant)
{
        try
        {
                Class.forName(driver);
                Connection connection=DriverManager.getConnection(url);
                Statement statement = connection.createStatement();
```

```
                System.out.println("Current SQL is : "+sql);
                boolean hasResults = statement.execute(sql);
                if(hasResults)
                {
                        ResultSet result = statement.getResultSet();
                        if(result!=null)
                        {
                                displayResults(result,indexYouWant);
                                connection.close();
                        }
                }
                catch(Exception ex){}
        }


// Display the Student Record

void displayResults(ResultSet r,int indexYouWant) throws SQLException
{
        ResultSetMetaData rmeta = r.getMetaData();
        int foundrec = 0;
        int numColumns=rmeta.getColumnCount();
        int SlNo=0;
        String SlNo_txt="";
        String StudentName="";
        String Matric="";
        int Sex=0;
        String imagename="";
        int count=0;

        while(r.next() && count!=indexYouWant)
        try{
                count++;
                SlNo=r.getInt(2);
                SlNo_txt=Integer.toString(SlNo);
                StudentName=r.getString(3);
                Matric=r.getString(4);
                Sex=r.getInt(5);
                imagename=r.getString(6);
            }
        catch(Exception e)
            {
                e.printStackTrace();
            }

        if(count!=indexYouWant)
         {
                JOptionPane.showMessageDialog(this,"No record(s)found.","Warning",
                JOptionPane.WARNING_MESSAGE);
         }
         else
        {
                jlabel5.setText(SlNo_txt);
                jlabel3.setText(StudentName);
                jlabel7.setText(Matric);
```

```
                labelim.setIcon(new ImageIcon(imagename));
        }
        if(Sex==1)
        {
                radio1.setSelected(true);
        }
        else if(Sex==2)
        {
                radio2.setSelected(true);
        }
}
```

6.3.2 Main Module (Face.java)

The main module is to create a function of recognition by display the image and details of the person being compared successful.

Sample coding:

```
// Recognition process

if(ev.getSource()==recognizeButton)
{
        if(pathName!=null && pathName.length()>0 && fileName!=null &&   fileName.length()>0)
        {
                try {
                        TimeUpdate time=new TimeUpdate(System.currentTimeMillis());
                        time.start();
                        EigenFaceCreator creator = new EigenFaceCreator();
                        creator.THRESHOLD=THRESHOLD;

                        creator.readFaceBundles(pathName);
                        String result =  creator.checkAgainst(fileName);
                         String result1 = pathName +"\\" + creator.checkAgainst(fileName);
                        time.setStop();
                        System.out.println("Most closly reseambling is: "+result+" with
                        "+creator.DISTANCE+" distance.");

// display result

                JOptionPane.showMessageDialog(null, "Constructing face-spaces from
                "+pathName+" ...\nComparing "+fileName+" ...\nMost closly reseambling: "+result+"
                with "+creator.DISTANCE+" distance.", "FaceRec Result",
                JOptionPane.INFORMATION_MESSAGE);

                 if(result==null)
                JOptionPane.showMessageDialog(null, "Matching Fail...Image Not  Found","Error
```

```
Matching", JOptionPane.INFORMATION_MESSAGE);

 else
{
String sql = "SELECT * FROM studentrecords WHERE picture= ('"+result1+"')";
int count=0;

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection connection=DriverManager.getConnection("jdbc:odbc:student");
Statement statement = connection.createStatement();
boolean hasResults = statement.execute(sql);
if(hasResults)
{
        ResultSet resultSet = statement.getResultSet();
        if(resultSet!=null)
        {
                while(resultSet.next())
                {
                int Sex=resultSet.getInt("Sex");
                text_StudentName.setText(resultSet.getString("StudentName")
                text_MatricNo.setText(resultSet.getString("Matric"));
                if(Sex==1)
                {
                        RB_Male.setSelected(true);
                }
                else if(Sex==2)
                {
                        RB_Female.setSelected(true);
                }
                LB_PathImageMatch.setText(pathName+"\\"+result);
                LB_ImageMatch.setIcon(new ImageIcon(pathName+"\\"+result));
                count++;
                }
        }
}
connection.close();
```

6.3.3 Camera Module (Camera.java)

The camera module is to connect the system with the web camera for the purpose to captured photo of student by implement JMF.

Sample coding:

```
// Captured image using Web Camera

public Camera() {
                addWindowListener(new WindowListener());
                setTitle("JMF FaceRec Camera ");
                try {
                        vision1 = new FrameGrabber();
                        vision1.start();
                }
                catch(FrameGrabberException fge) {
                        System.out.println(fge.getMessage());
                }
                image1 = vision1.getBufferedImage();
                setSize(image1.getWidth(this) +20 , image1.getHeight(this) +40 );
                setVisible(true);
                setResizable(false);
                setBackground(Color.black);

                while(true) {
                        image1 = vision1.getBufferedImage();
                        image2 = proces(image1);
                        repaint();
                        try {
                                Thread.sleep(1);
                        } catch(Exception e) {
                                System.out.println("Error");
                        }
                }
        }
```

6.3.4 Recognition Module (EigenFaceComputation.java)

The recognition module is to implement the eigenfaces algorithm into the system and also doing computation and try to find out the match face image.

Sample coding:

```
// Compute Average Face

for ( pix = 0; pix < length; pix++)
{
        temp = 0;
        for ( image = 0; image < nrfaces; image++)
        {
        temp +=  face_v[image][pix];
        }
        avgF[pix] = temp / nrfaces;
}
```

```
// Compute difference and covariance matrix

for ( image = 0; image < nrfaces; image++)
{
        for ( pix = 0; pix < length; pix++)
        {
        face_v[image][pix] = face_v[image][pix] - avgF[pix];
        }
}
System.arraycopy(face_v,0,faces,0,face_v.length);

Matrix faceM = new Matrix(face_v, nrfaces,length);
Matrix faceM_transpose = faceM.transpose();

 Matrix covarM = faceM.times(faceM_transpose);


// Compute Eigenvalue and eigenvector

EigenvalueDecomposition E = covarM.eig();

double[] eigValue = diag(E.getD().getArray());
double[][] eigVector = E.getV().getArray();
```

6.4 Functionality of the Final System

As was mentioned before, the system can be divided into two main modules. Therefore, it can be said that the system has two main functions: the implementation of the database and the recognition and identification process. The system also provides additional functions to the user. Users of the system could choose which functions that they would like to perform at any given time. In this section, these functions would be explained in detail, together with the explanation of the use of the face recognition and identification system.

6.4.1 Using the system – Main Interface

Every user of the system would first come across the main interface of the system, that shows the clear division of the system's two main functions, as seen in Figure 6.1. Users of the system could choose any one of these two main functions of the system at any given time, depending on the existence of the training set of face images.

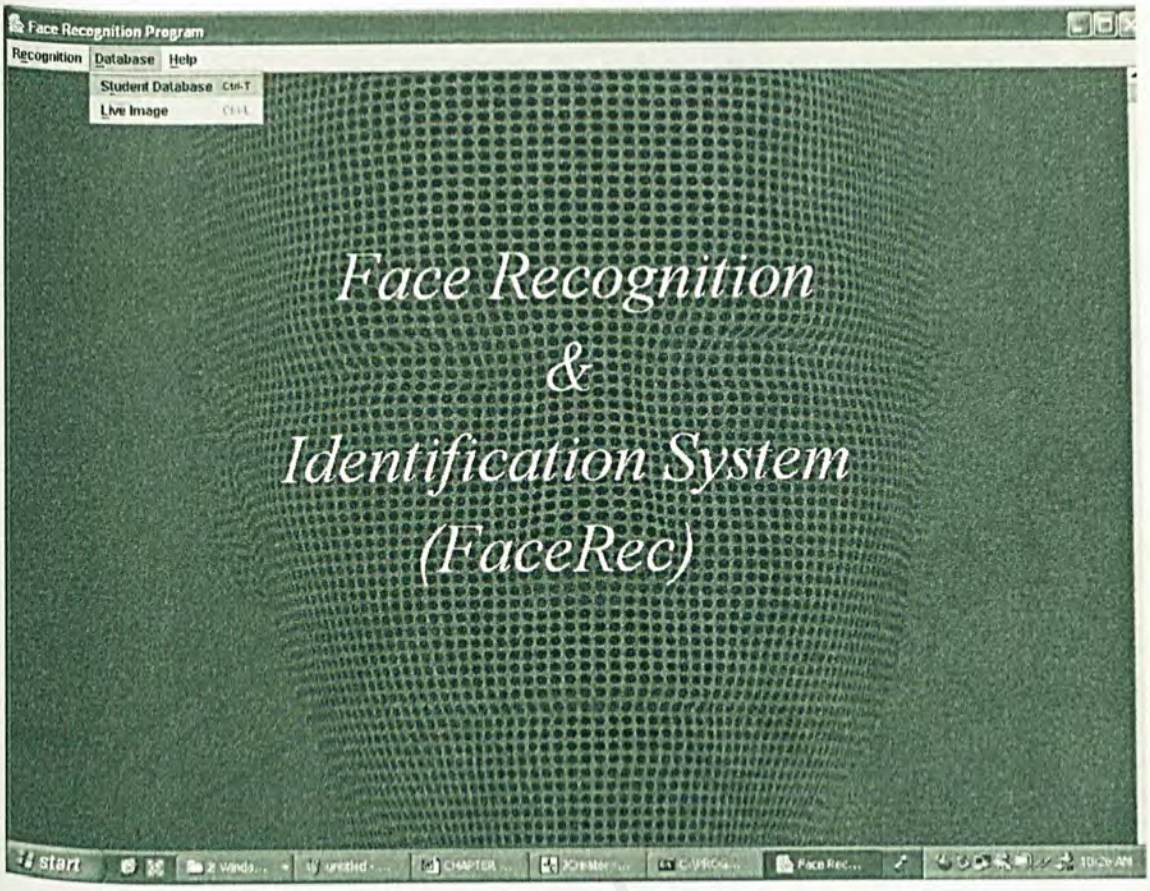


Figure 6.1 Main Interface

When a user first starts using the system, the user needs to first input the training set of face images. Therefore, the user must first choose the Database manipulation function before proceeding to the Recognition and Identification function of the system. But if the training set of face images have already been input into the database, then a user can choose to begin with the recognition and identification process. Let it be known thought, that the user must first ensure that the image resolution and size of the selected test image to be recognized must be the same as those of the training set of face images.

6.4.2 Using the system – Database manipulation

Here, the user is allow for the process of manipulating the database. This function is important as it prepares the training set of face images and its relevant data to be used in the recognition and identification process. Under this function, the user could

choose Student Database to input a new record, or search an existing record, or delete an existing record.

➢ *Search or Delete an existing record*

The user of the system could search an existing record to display the student record. To search or delete the record, the user needs to select the 'Student Database' command of the 'Database' function, as shown in Figure 6.1. Once this is done, a new dialog box would be displayed.

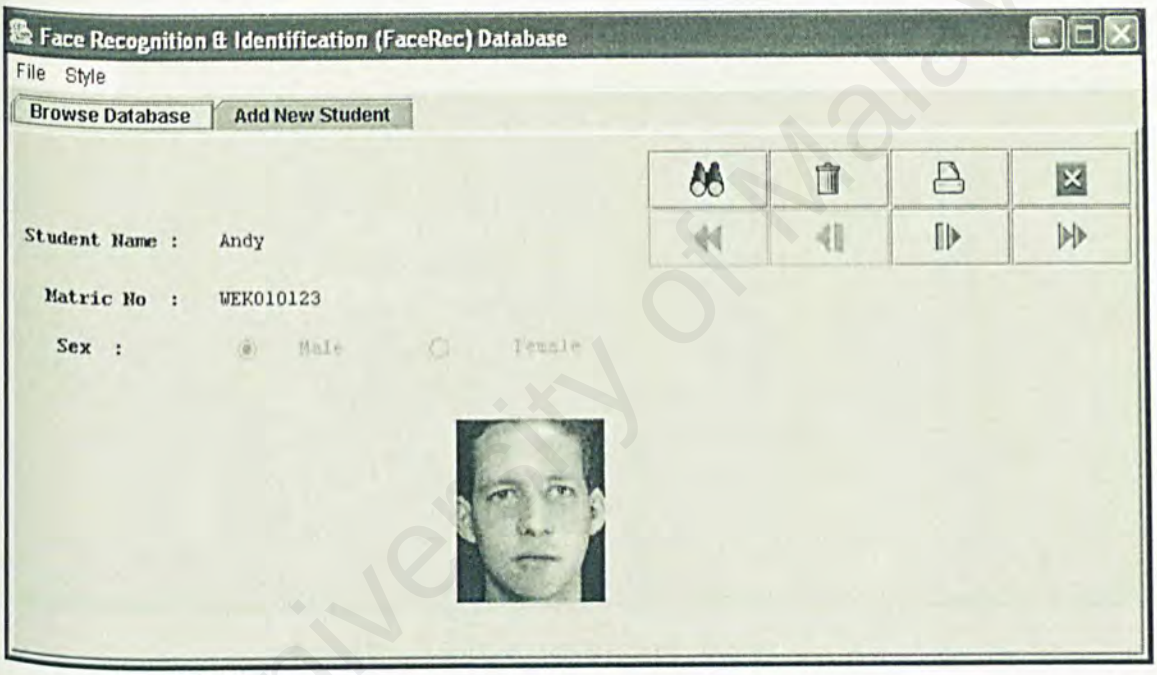


Figure 6.2 Search or Delete the existing system

Once the dialog is shown, the user can either choose Search or Delete button. Once this button is selected, a dialog box would appear requesting the user to input the Matric Number of the person whose record needs to be searched or deleted. Once the user has entered the required information, the system would check whether the record exists in the database. If it does, the record would be display for search function else an error message would be displayed, informing the user of the non-existence of the required record. For the delete function, a confirm dialog box will be prompt. If OK button is click, the record will be permanently deleted.

➢ *Input a new record*

To input a new record into the database, the user needs to select the 'Student Database' command of the 'Database' function, as shown in Figure 6.1. Once this is done, a new dialog box would be displayed. The user need to click the 'Add New Student' function in order to enter all the required data needed to recognize and identify a person, as shown in Figure 6.3. Once the user has entered all the required data, the user can click the 'Save Data' button. This will automatically save the new record.

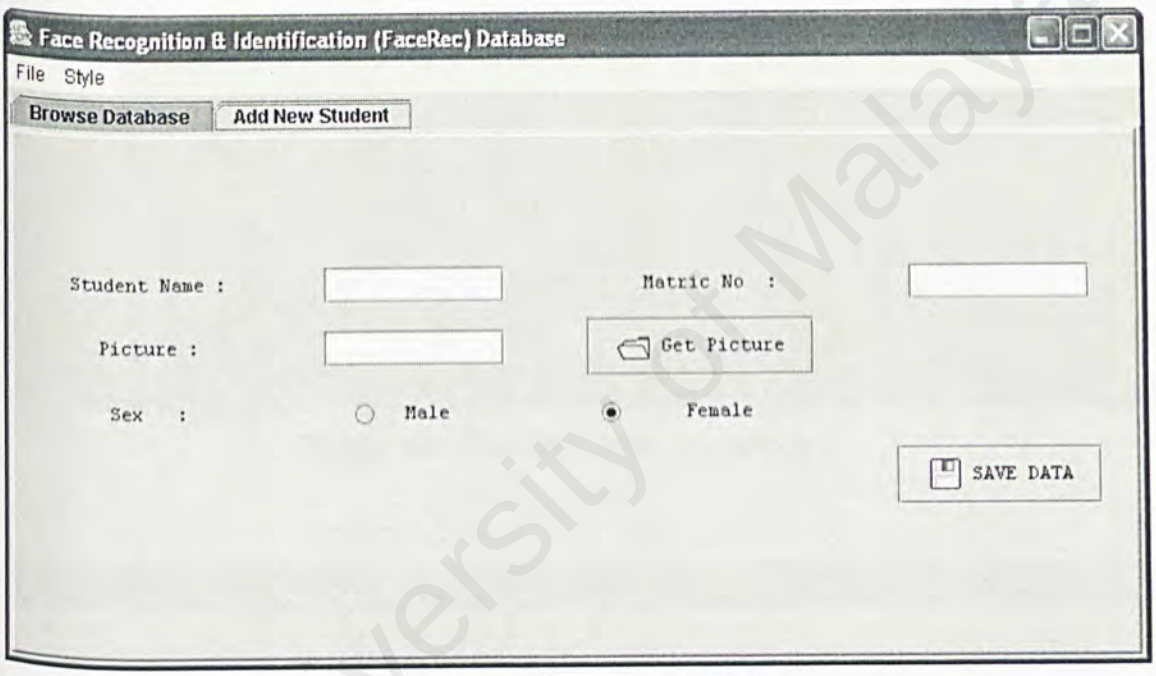


Figure 6.3 Add a new record dialog

6.4.3 Using the system – Face Recognition and Identification

Once the user has entered the training set of face images, the recognition and identification process can now begin. A user can choose to perform this process by selecting the 'Recognition' function from the Main Interface and select the 'Start Recognition' command as shown in Figure 6.4. A new dialog box would be displayed as shown in Figure 6.5.

Figure 6.4 Choosing Start Recognition

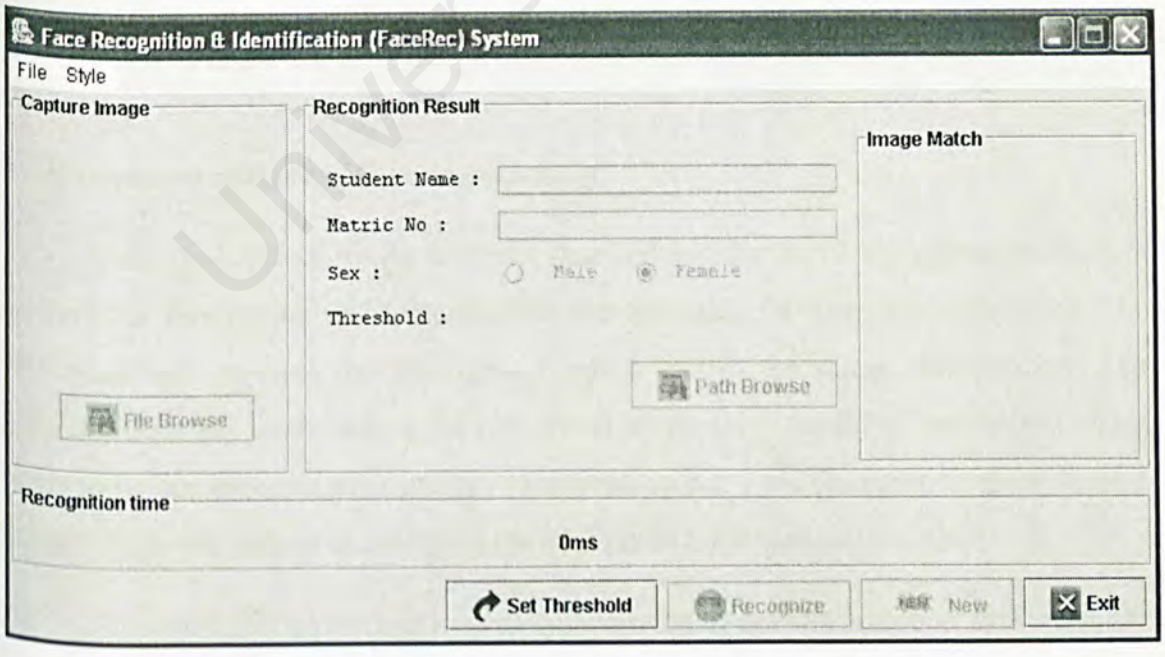


Figure 6.5 Face Recognition and Identification System

➢ *Acquiring the test face image*

The user can begin the process of recognition and identification by selecting the 'Start Recognition' command of the 'Recognition' function. Selecting this command will automatically display a standard dialog box so that the user could select the required JPEG file that contains the face image would be displayed and ready for recognition and identification, as shown in Figure 6.6.

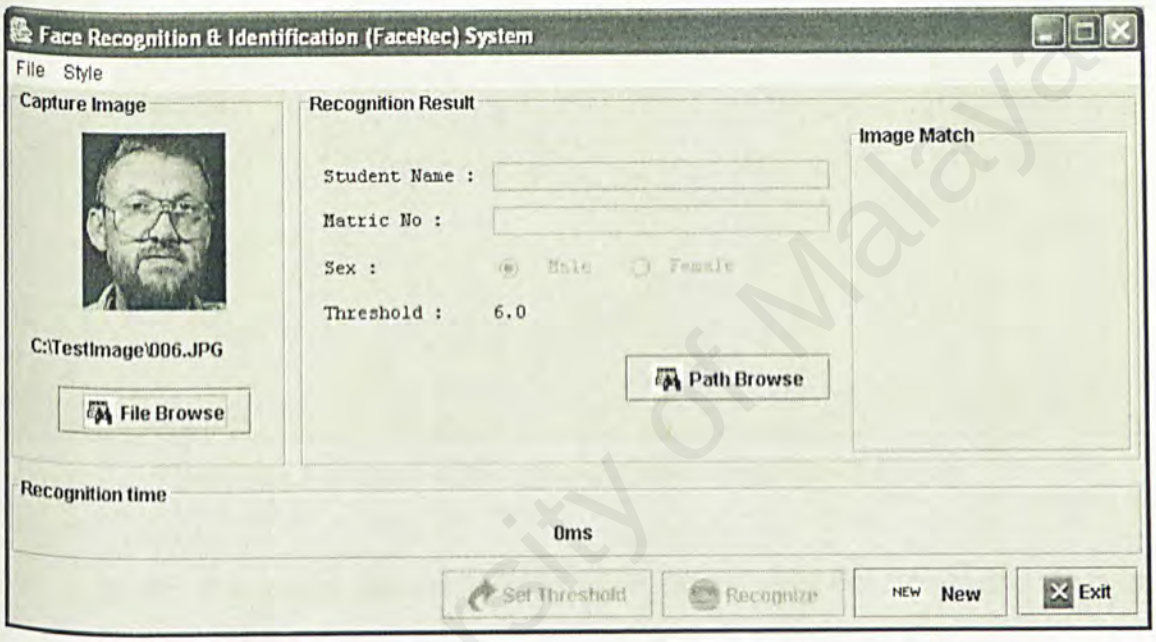


Figure 6.6 Test image displayed and ready for recognition

➢ *Recognition and Identification process*

Once the test face image has been acquired and displayed, the system is ready to perform the recognition and identification process using the eigenfaces approach. The user needs only to click the 'Recognize' button to start the recognition process. This process would take some time as the eigenfaces needs to be calculated and the test image needs to be compared to each image in the training set. The 'Recognition time' located at the bottom part aids in determining the time taken for the whole process.

When the recognition process is completed and the test image is identified, the result would be displayed on the screen, as shown in Figure 6.7. The result that is displayed contains the relevant information that would identify the face in the test image

along with the face image that matches it. This is to allow the user to compare both face images in order to determine the accuracy of the face recognition and identification system developed.

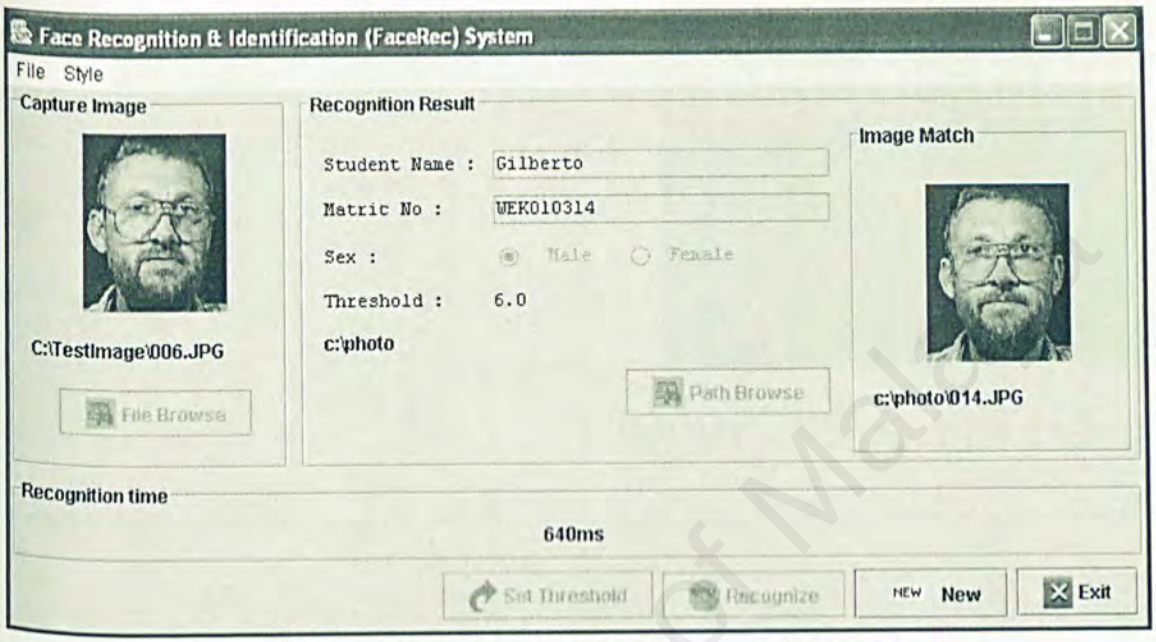


Figure 6.7 Example of the result of the recognition process

If the test image cannot be identified or exceeds the acceptable distance threshold, then a message would be display to the user that no image is found.

6.4.4 Using the system – Live Image

One of the modules of the system is to implement live capturing of human faces using web camera. The user of the system could grab the image of the student before save it into the database. To use the web camera, the user needs to select the 'Live Image' command of the 'Database' function, as shown in Figure 6.1. If the camera is detected, the process of capturing image can be done in very easy way. Java Media Framework API (JMF 2.1.1.e) must be install in order to implement this module.

Chapter 7: System Testing

CHAPTER 7: SYSTEM TESTING

7.1 Introduction

System testing is a significant and critical phase that ensures the system fulfills the user's requirement s and assures the quality of the delivered system. Testing provides a method to discover logical error and to test the system reliability. It is done throughout system development, not just at the end. This is because system that is failed after installation will result a waste in cost, time and effort. However successful testing will result in quality software with less errors and work according to specification.

Several testing stages that involve during the development of the system are:

- Unit testing
- Integration testing
- System testing

7.2 Unit Testing

In unit testing stage, testing will be concentrated on the smallest component o the system for testing. In his phase, individual components are tested to ensure that stand-alone program fixes the bug without side effects. After new component is developed, it is tested independently without other system components. This is to assure that the component is able to work accurately and persistently. All function on each button is examine to ensure it perform the entitles output such as hyperlink to the right page, call

the right function to execute, display the correct message according to the error and eliminates all the syntax faults occurred.

In FaceRec, those units that were tested independently are:

- Installation
- Add student record
- Delete student record
- Search student record
- Recognition testing
- Button testing

7.3 Integration Testing

After all components have been unit-tested, the next step is ensuring that the interfaces among the components are defined and handled properly. This step is called **integration testing**, also known as module testing, which verifies that the all the components work together as described in the module or system design specifications.

During the integration testing, two or more units in which either unit that use output data from or provide input data for another unit were tested in collection.

The order in which components are tested affects our choice of test cases and tools. The system is viewed as a hierarchy of components, where each component belongs to a layer of the design. In this system, the Bottom-up Integration approach is used where testing begins from the bottom and works the way up. The process is continued until all the modules are tested.

7.3.1 Bottom-up Integration

Bottom-up testing is one popular approach for merging components to test the larger system. In this method, each component at the lowest level of the system hierarchy is tested individually first. Then the next components to be tested are those that cal the previously tested ones. This approach is followed repeatedly until all components are included in the testing. The bottom-up method is useful when many of the low-level components are general-purpose utility routines that are invoked often by others, when the design is object-oriented or when the system is integrating a large number of stand-alone reused components. Testing which had been carried out throughout FaceRec is based on Bottom-up Integration approach as shown in the following figure.

7.4 System Testing

System testing is the last testing procedure. It is differences from unit testing and integration testing. System testing performed to uncover its limitations, measure it capabilities and make certain that the entire system works according to user's specifications. Developers will join the users to perform this stage of testing where the system is checked against the user's requirements description.

System tests study all the concern issues and behaviors that can only be exposed by testing the entire integrated system or major part of it. The testing process is also concerned with validating the system meets it functional and non-functional requirements. Under system testing, the whole process was simulated and followed through until the end.

System modification will be implemented if there is a need to change or do not met the user's requirements specifications. If the users are satisfied with the system's characteristics, the system is ready to be deployed for use. The testing result will show whether or not the entire system specifications and objectives are achieved.

7.4.1 System Test Considerations

In system testing, the behavior of the individual functions and functional tests also involved:

- **The Event List**

All the possible triggers are exercised and the expected results compared with the actual result. Every function is tested by one or more events in the event lists.

- **Error Message Testing**

The error message, which can be generated by the system during invalid data entry are checked for spelling, appropriateness and consistence. Acknowledgement messages will also implement the same list. It is the message that informs the user about the state of a user request process.

- **Documentation Testing**

All examples used in the user's manual is tested for correctness and for whether or not the manual gives the exact answers users will obtain when they run the examples.

7.5 Fundamental Tests (Product Verification Testing)

There are also tests fundamental to all software. Certain of these are difficult to measure accurately. Four of these fundamental tests are:

- **Usability**

The usability should be based in building user interfaces that have patterns already familiar to the typical user. The user then learns to use the software through pattern matching and paradigm shifts, exactly as they do in mastering any product.

- **Install Ability**

How easy is it for a novice to install the software correctly and easy without resource to an expert?

- **Performance**

Performances tests are conducted to ensure that the system response time meet user expectations and do not exceed the specified performance criteria under heavy stress or volume. During these tests, response time and the transaction rate are measured, the purpose of performance tests is to test-run the performance of various functions of the software within a specified hardware configuration. The performance tests can couple this test with stress testing.

- **Reliability**

Reliability tests are conducted, according to mathematical models of software reliability, to ensure that the system can be probability of some function of the system failing within a specified time. Reliability testing is monitoring the mean time between

failures. Reliability and consistency testing go hand in hand where the system behavior (inputs, outputs, response time) is measured for consistency.

In conclusion, the testing steps are shown as below.

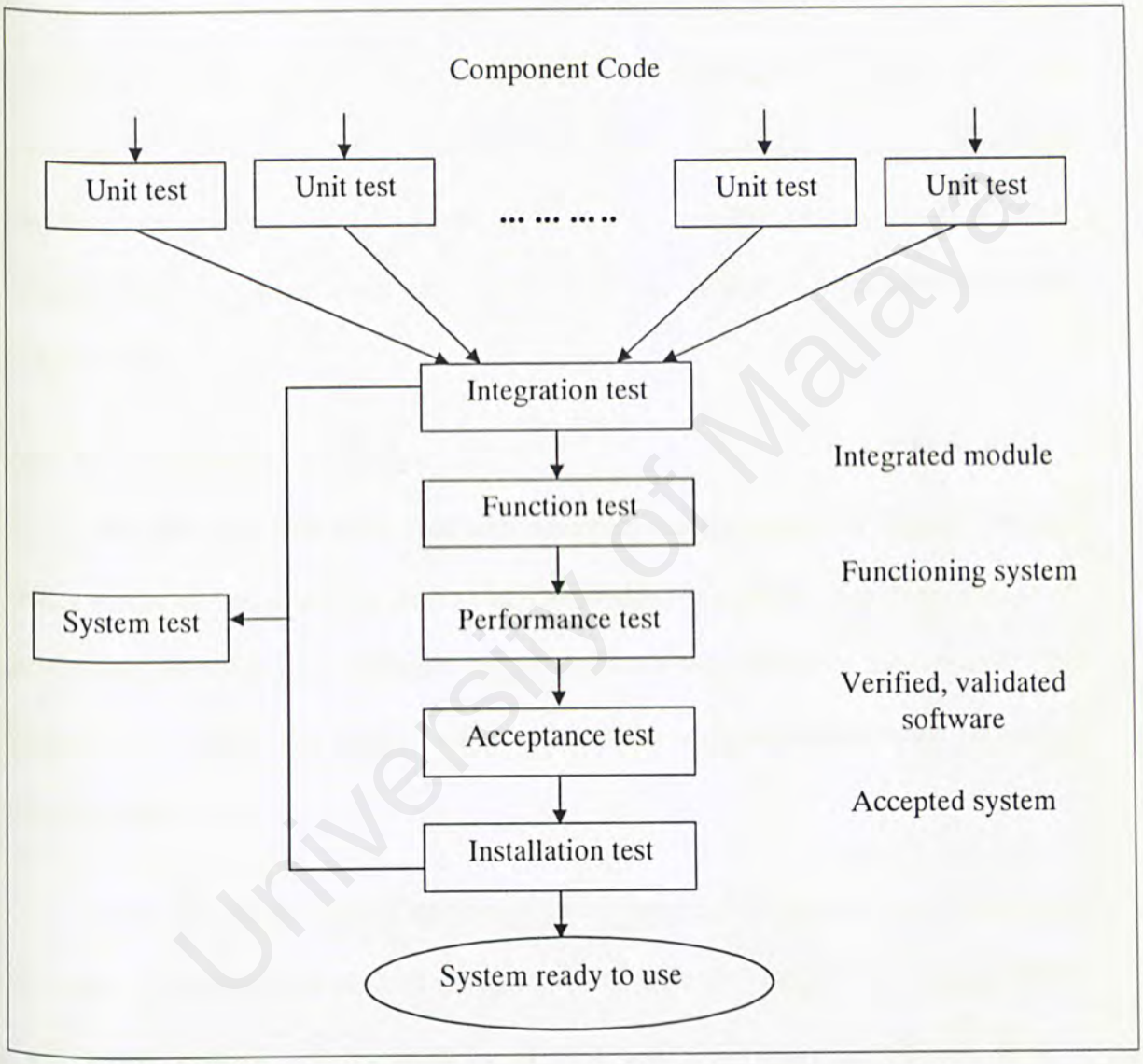


Figure 7.1 Testing Steps

7.6 Performance

Like all other systems, it is important to perform a number of tests or experiments on the system in order to determine its performance. The system was first tested using a database of face images that share the same attributes, such as approximately similar head size and minimal light variations between images, in order to determine the performance of the system under certain constraint condition. Once this has been performed, several experiments were done on the system involving light variation, head orientation and size variation of the face image in performing the recognition process.

7.6.1 The database of face images

The face image database used in this project was taken from of Surrey, Olivetti-Oracle Research Lab (AT&T Laboratories Cambridge). It consists of the face images of 20 persons, each taken in different angels and each with different expressions. The pictures are greyscale face images, taken with a black background and under controlled lighting conditions.

Constraint by the limited number of face images in the database, only one image was taken for each person as the training set (that of the full frontal face images) while the rest were used for the test samples. Therefore, the training set consists of 20 full-frontal face images, as shown in Figure 7.2. As for the test samples, a combination of 18 pictures were taken, each with different head orientation, light variation and face sizes.

Figure 7.2 Example of the training set of face images

7.6.2 Testing of the system

At the beginning, the test sample that was used consists of 20 images, each taken from the face images in the training set. It was discovered that all the faces in the test sample were accuracy matched and identified; no faces were rejected as unknown faces. The result gave 100% correct recognition ad identification.

7.6.3 Experiment 1: Head orientation

When different head orientation and different expressions of the face images were used as the testing sample as shown in Figure 7.3, it was discovered that the accuracy of the recognition and identification process drops significantly. When 6 thresholds were used, the system achieved an accuracy of around 50% of correct identification.

Figure 7.3 Example of the test image for different head orientation and expression.

7.6.4 Experiment 2: Lighting conditions change

When there was a small change in the lighting condition of the face images used as the test images, it was discovered that the system achieved more than 50% correct identification of the sample test image. But when there was a significant change of lighting condition of the face image used as the test sample as shown in Figure 7.4, the recognition rate drops significantly to only 20%.

Figure 7.4 Example of the test image for different light source.

7.6.5 Experiment 3: Face size change

When the size of the face images changes dramatically, where much larger face images were used as shown in Figure 7.5, it was discovered that the process of recognition and identification were only 35% correctly identified.

Figure 7.5 Example of the test image with different head size and scale

The above experiments were done mainly to show the main characteristic of the eigenfaces approach under different sets of conditions. It must be noted, however, that is more face images for each person is included in the training set, the error rate is likely to be decreased.

7.7 System Maintenance

Usually in system develop, maintenance will be conducted once the system is finished or delivered. The maintenance services will be make sure the system function properly, modify some application or add new functions in this system. Unfortunately, in this Face Recognition and Identification System (FaceRec), there will be no longer

maintenance services. The reason is because FaceRec is not deploying in any organization except this system will be using by organization. Then only the maintenance services will be implemented.

7.8 Chapter Summary

In summary, the performance of the eigenfaces approach in face recognition is depending on several attributes or factors, such as:-

- **Light variation**

The light variation between images in the training set and within a face image plays an important part in the efficiency of the face recognition algorithm using eigenfaces. As explained previously, the larger this variation is, the higher the error rate when performing recognition. Therefore, to achieve the optimal performance in face recognition, the lighting variation between images in the training set should be small as to not affect the efficiency of the algorithm.

- **Head orientation**

If the head orientation of the face images in the training set is approximately similar to one another and to that of the test image, then the error rate of the eigenfaces approach is minimal.

- **Head size**

The head size of the face images in the training set must be approximately similar to one another and to that of the test image in order to achieve optimal efficiency in the recognition process.

Chapter 8: System Evaluation

CHAPTER 8: SYSTEM EVALUATION

8.1 Introduction

Evaluation is a process that occurs continuously at all phases of the system development. Evaluation phase was to determine the extent to which the system the expected outcomes have been realized, and the prescriptive value of the process where extraneous factors were taken consideration. Lastly, conclusion will be making for this system.

8.2 Problems Encountered and Recommended Solutions

In every project, the problems always occur during system development. Throughout this project, many problems have kept unfolding one another as development work progresses due to many reasons.

8.2.1 During Analysis Phase

- **Determining Scope of the System**

Since there is less experience in developing system, it was hard to determine to which extent to define the scope of the system so that it can be completed within the given time frame. However, this was overcome by analyzing and studying all of the capabilities provide by Java, Java SDK, Windows 2000 and other technologies before determining the scope of the system.

8.2.2 During Design Phase

- **Time Constraint**

There was not enough time to study, learn and produce the best solution of design in Semester 1. Mainly, this was cause by inexperience and insufficient knowledge of designing a system. Furthermore, time is needed to study and explore Java language and also eigenfaces algorithm, before knowing how to apply these technologies and languages in the process of developing and solving problems. Implementing face recognition algorithm is still new and there is not much source to be found. Thus the best way is to study as many approaches used by others researcher and also their documentation.

8.2.3 During Implementation Phase

- **No prior experience in the chosen programming languages**

There was a learning curve in understanding how the Java works since inexperience in Java. Scripting in a new environment such as JAMA and implementing eigenfaces algorithm using Java requires some knowledge of how the matrix class is used in order to build the required functionality of the system application. The best way of learning java is refer to some of the sample examples available in the Java reference books and Internet.

- **Problem on Installation**

During implementation phase, there were a lot of problems on installing and configuring Windows 2000, JCreator, Java SDK, and other tools before starting the

coding. Some of the needed software and tools were successfully installed only after a few times of formatting and reinstallation. From this, I learn to know that, it is essential to know the sequence of products installations. This is to ensure smooth execution without system errors.

8.3 System Strength

This system although does not have powerful features to some extent, but still has some strength of its own when compare to others face recognition system.

8.3.1 Advantages of the system

In this section, the advantages of developing the system using the approaches that was mentioned previously are discussed, along with the advantages of using the system as a whole.

- **The use of Microsoft Access 2000 database**

As was mention before, the purpose of the system is to not only recognize faces, but also to identify them. Therefore, along with face images, other relevant information that could identify a person needs to be stored as well.

By using Ms Access 2000 database, the information needed to recognize and identify a person could be stored in a systematic and organized way. Ms Access allows the design of the database to be transparent to the user, thus eliminating the need for the user to know how the data are being stored. Also, accessing the required data could be done in more efficiently by incorporating Ms Access into the system.

- **The use of JPEG files**

Most of the current recognition systems that are available take as input face images in the form of raw format. This will make it difficult for the user to view or manipulate these images, as normal drawing software does not support this type of file.

In the case of the face recognition and identification system that was developed for this project, it takes as input face images in the form of JPEG files. This will make the viewing an manipulation of the images easier as mostly all existing drawing software supports this format. Therefore, images that are taken as input could be pre-processed before the recognition and identification process takes place.

- **Easy to access performance of system**

Because the system that was developed in this project not only recognizes, but also identifies a person, the result of running the system would be displayed in such a way as to allow for the comparison between the face images that were to be recognized with those that were identified. By following both pictures to be displayed in the same screen, one next to the other, the accuracy or correctness of the recognition process could be evaluated. Thus, this will allow users to evaluate the performance of the system instantaneously.

- **Performs identification**

Because the system also incorporates identification of a person whereby each person in the database is given a unique identifier, the system could be used with other system that uses databases that deal with the storing of data of a person. The unique

identifier could be used to identifying a person in both databases. For example, this system could be used along with a "Student Record System" whereby this system is used to identify a student from a given picture, and the other system is used to get relevant information on the student, such as his/her name, address, telephone and etc.

- **User-friendly Interface**

By following standard windows application design such as the used of the menu selection, it makes the use of the system less complicated and easier to use. With the use of enable/disable menu selection commands and buttons, along with the help function, it can guide the users in using the developed system.

Because of the user-friendly interface, the user can perform both the recognition & identification function and the database manipulation function in a straight forward manner.

8.4 System Constraints

8.4.1 Limitation of the system

As with other systems, the face recognition and identification system that was developed for this project has its own set of limitations. In this section, the problems and limitations of the developed system would be explained in detail.

- **Normalizations of face images.**

All of the experiments mention in the previous chapter were performed using face images that have been pre-processed so that

By using Ms Access 2000 database, the information needed to recognize and identify a person could be stored in a systematic and organized way. Ms Access allows the design of the database to be transparent to the user, thus eliminating the need for the user to know how the data are being stored. Also, accessing the required data could be done in more efficiently by incorporating Ms Access into the system.

- **The limitation on the number of face images per person**

As was mention before, the purpose of the system is to not only recognize faces, but also to identify them. Therefore, along with face images, other relevant information that could identify a person needs to be stored as well.

By using Ms Access 2000 database, the information needed to recognize and identify a person could be stored in a systematic and organized way. Ms Access allows the design of the database to be transparent to the user, thus eliminating the need for the

user to know how the data are being stored. Also, accessing the required data could be done in more efficiently by incorporating Ms Access into the system.

- **The use of one database**

The system that was developed in the project uses only one copy of a database in all of its execution. Therefore, if a user wants to use several different data stored in different databases (all with same format), the user needs to first delete the data in the existing database by using the "Delete Record" function, and then enter the required new data one-by-one. This process is not only time-consuming but is quiet tedious as well. Another way to overcome this limitation is to move the currently used database into another folder and move the required database into the necessary folder to be accessed by the system.

8.5 Future Enhancement

In order to overcome the limitations of the current system, there are many ideas that could be looked into for enhancing the performance of the face recognition and identification system that was developed in this project. Some of the enhancements that could be considered are:-

- **Normalization of face images**

As was mention in the previous section, the normalization of the face images used both in the training set and test samples are important in decreasing the error rate in the face recognition and identification process. It would also benefit the users of the

system tremendously if the process could be done automatically, without having to perform it manually.

One way of performing the normalization of face images is by incorporating the Active Shape Modules. This model is used to locate a large number of characteristics points on the face that can be used as reference points the variables are constrained by capturing faces in a consistent manner.

Usually, the eyes are used as the reference point, whereby the distance between the centers of the eye socket is fixed for all the face images in the training set. Some of the methods that could be used to normalize a face images are template constructors and template descriptors.

- **Using Gaussian Smoothing**

As was seen in the results of the experiment done, the presence of background images does affect the performance on the recognition and identification process. Therefore, in order to decrease the error rate of the system and increase its performance, Gaussian Smoothing could be incorporated into the system in order to minimize the background effect.

- **Options for users to input number of images**

In order to overcome the limitation, the users could be given the option of entering the number of face images that he/she wants for each person in the training set. Therefore, it would be worthwhile if the system could increase or decrease the number of face images for each person as required by the users. This could be done by

dynamically adding or deleting the columns in the database. Therefore, adding this extra feature in the database manipulation module of the system could increase the flexibility of the end-system.

8.6 Conclusion

The face recognition and identification system that was developed has met its objective. Both the eigenfaces algorithm and the database technology were able to be implemented using Java and Microsoft Access 2000 respectively.

The developed system is not performs the recognition of the face images, but it also performs the identification of face images along with database manipulation. By incorporating Microsoft Access 2000 into the program, the relevant data needed to identify a person along with her/his respective face images could be stored systematically. And because Java supports the use of database in Microsoft Access 2000, the data that are stored in the database could be easily accessed and manipulated.

It can also be concluded that using JPEG file as the format for the face images helped in the pre-processing of images that needs to be perform before the recognition and identification process can take place.

BIBLIOGRAPHY

1. AT&T Laboratories Cambridge. (1999). *ORL face database*. Available from: http://www.uk.research.att.com:pub/data/att_faces.zip [Accessed 14 August 2003]

2. Eigenfaces Group. (1999). *Eigenfaces Face Recognition program*. Available from: http://www.owlnet.rice.edu/~elec301/Projects99/faces/ [Accessed 19 July 2003]

3. Garfinkel, S. (1996, March). *Hot zones*. Wired, 159.

4. Gordon, G. (1995). *Face recognition*. Available from: http://www.tasc.com/news/prism/0595/articles/face.html. [Accessed 22 July 2003]

5. Kelly,M.D.,1970. "*Visual Identification of people by computer*", Tech.Rep.AI-130, Standard AI Project.

6. Lau Technologies. (1999). *Facial recognition technology*. Available from: http://www.lautechnologies.com/Pages/FacialRecognition.htm. [Accessed 22 June 2003]

7. Mandel, T. (1997) *The Elements of User Interface Design*, Wiley.

8. Microsoft Corporation. (2003). *Top 10 reasons to move to Windows 2000.* Available from: http://www.microsoft.com/windows2000/professional/evaluation/whyupgrade/default.asp [Accessed 30 July 2003]

9. Miros. (1999a). *What are neural networks*. Available from: http://www.miros.com/Neural_networks_description.htm. [Accessed 28 June 2003]

10. Miros. (1999b). *Evolving approaches to recognizing a friendly face*. Available from: http://www.miros.com/Neural_networks_description.htm. [Accessed 28 June 2003]

11. Mueller.J.P.2002.Visual C++ 6: from the Ground Up. Osborne/McGraw-Hill. 2nd Edition.

12. Page, D. (1998). *Biometrics*: Facing down the identity crisis. Available from: http://www.hightechcareers.com/doc198/biometrics198.html. [Accessed 27 July 2003]

13. Pfleeger,Shari Lawrence.2001. Software Engineering: *Theory and Practice*. New Jersey:Prentice Hall. 2nd Edition.

14. Ponti, R. (1999). *Facial features identification* [online].Available from: http://www.tech.purdue.edu/it/resources/aidc/BioWebPages/Biometrics_Face.html.

15. Pressman, Roger S. (2001) *Software Engineering: a practitioner's approach* – 5th edition. McGraw-Hill.

16. Sommerville I.,2001. *Software Engineering*. 6th ed. United States of America: Addison Wesley.

17. Starner.T (1992). *Eigenfaces Face Recognition program.* Available from: http://www2.cs.cmu.edu/afs/cs.cmu.edu/project/codabumba/eigenfaces/main.c [Accessed 16 July 2003]

18. Technology Recognition Systems. (1999). *Facial thermograms*. Available from: http://www.betac.com/trs/facial.htm. [Accessed 9 August 2003]

19. Turk, M. & Pentland, A. (1991). *Eigenfaces for recognition*. Journal of Cognitive Neuroscience, 3, 71-86.

20. Vismod, (1998). *Photobook/Eigenfaces Demo.* Available from: http://www-white.media.mit.edu/vismod/demos/facerec/basic/html. [Accessed 18 July 2003]

21. w3schools, (2003). *OS statistics*. Available from: http://www.w3schools.com/browsers/browsers_stats.asp [Accessed 29 July 2003]