3D FACE RECOGNITION

Project submitted in partial fulfillment of requirements

For the Degree of

BACHELOR OF ENGINEERING

BY

ABHIJIT BHANDARI

PRAVIN AGRAWAL

ROHIT JADHAV

SUNIL GOLHAR

Under the guidance of

Internal Guide

Prof. K. T. TALELE

Department of Electronics Engineering

Sardar Patel Institute of Technology

University of Mumbai

2009-2010

BHARTIYA VIDYA BHAVAN'S

SARDAR PATEL INSTITUTE OF TECHNOLOGY

MUNSHI NAGAR, ANDHERI (W),

MUMBAI - 400 058.

2009-10

CERTIFICATE OF APPROVAL

This is to certify that the following students

ABHIJIT BHANDARI

PRAVIN AGRAWAL

ROHIT JADHAV

SUNIL GOLHAR

have satisfactorily carried out the project work entitled

"3D FACE RECOGNITION"

towards the fulfillment of Bachelor of Engineering course in Electronics of the Mumbai University

Prof. K. T. Talele
INTERNAL GUIDE

Prof. Dr. S. T. Gandhe
H.O.DELECTRONICS

Dr. Prachi Gharpure
PRINCIPAL

UNIVERSITY OF MUMBAI

2009-2010

PROJECT APPROVAL CERTIFICATE

This is to certify that the project entitled "**3D FACE RECOGNITION"** has been duly completed by the following students

ABHIJIT BHANDARI

PRAVIN AGRAWAL

ROHIT JADHAV

SUNIL GOLHAR

Under the guidance of Prof. K. T. Talele in the partial fulfillment of the requirement for the award of degree of Bachelors of Engineering (Electronics) of the university of Mumbai.

INTERNAL EXAMINER **EXTERNAL EXAMINER**

DEPARTMENT OF ELECTRONICS ENGINEERING

SARDAR PATEL INSTITUTE OF TECHNOLOGY

MUNSHI NAGAR, ANDHERI (W), MUMBAI-400058

ACKNOWLEDGEMENTS

We feel privileged to thank our project guide, Prof. K. T. Talele for providing all the support needed for successful completion of the project.

We would like to express our gratitude towards their constant encouragement, support and guidance throughout the development of the project.

We are very thankful to Dr. S. T. Gandhe Head of Electronics Engineering Department for his moral support and encouragement.

We would like to thank our fellow colleagues for their support throughout the project.

ABSTRACT

The main objective of the project is to develop a security system based on 3D human face recognition. The proposed system must recognize a 3D face model converted to a 2D image after its comparison with set of 2D images in the database.

We have used 3D Normalization and Template Matching for Recognition in our project, in which we compare the contour of the image (test image) with the set of images (training images), in the database.

The proposed system performs two important tasks i.e.”3D Normalization” and “Recognition”. 3D Normalization is rotation of a human face model along the three axis x-y-z in order to obtain frontal view of face model. This helps us to compare the input image with those in the database. This in turn increases the recognition rate. Recognition is performed to reveal the identity of the person whether he is present in the database or no. Such systems compare the portrait of a tested person with photos of people who have access permissions to object.

Table of Contents

Acknowledgement iii
Abstract iv

1. Introduction

1.1. Face Recognition: An Interesting Concept 1
1.2. 2D Face Recognition 2
1.3. Need for 3D Face Recognition 2

2. Literature Survey

2.1. Database 4
2.2. Normalization 4
2.3. 3D to 2D Conversion 5
2.4. Face Recognition 5
2.5. Template Matching 6

3. Proposed System

3.1. Problem Statement 8
3.1.1. Broad Problem Statement 8
3.1.2. Specific Problem Statement 8
3.2. Methodology 9
3.2.1. For Face Recognition 9
3.2.1.1 FRAV3D Database 9
3.2.2. For User Authentication System 10

4. Requirement Specification

4.1. Software Platform 11
4.2. Hardware Platform 11
4.3. Validation Criteria 11

5. Algorithms

5.1. Reading of Input Image 12
5.2. Normalization of 3D Input Image 12
5.2.1. Normalization along X-Axis 12
5.2.2. Normalization along Y-Axis 12
5.2.3. Normalization along Z-Axis 13
5.3. 3D to 2D Conversion 13
5.4. Algorithm for Face Recognition using Contour Matching 15
5.4.1. Variables 15
5.4.2. Input Image 15
5.5. Algorithm for User Authentication System 18

6. Results

6.1 Using Image Rotated Along X Axis 19
6.2. Using Image Rotated Along Y Axis 21
6.3. Using Image Rotated Along Z Axis 23

6.4. Using Image Rotated Along More than Two Axis 25
6.5. Using Image not in Database 27

7. Experimentation And Its Analysis ....
7.1. Performance Parameters 29
7.1.1. Matching Ratio...29
7.1.2. False Acceptance Ratio (FAR)...29
7.1.3. False Rejection Ratio (FRR) 29
7.1.4. Rejection Ratio (RR)..29
7.2. Experimentation...30
7.2.1. Database 30
7.2.2. Experiment 30
8. User Manual
8.1. Face Recognition Manual 32
8.1.1. To Start the Process of Face Recognition on Images..32
8.1.2. To Select an Image...33
8.1.3. To Convert the 3D input Image to 2D and Display it 34
8.1.4. Normalization of image along X-Y-Z Axis 34
8.1.5. To Perform Face Recognition 38
9. Application..39
10. Conclusion 40
11. Limitation 41
12. Future Scope..42
13. References..43

LIST OF FIGURES

Figure 3.1: Block diagram for template matching 9
Figure 3.2: From left to right, top to bottom, the acquisition sequence of a subject is displaced. Both BMP color images and VRML 3D meshes are shown 10
Figure 5.1: Contour of human face 16
Figure 5.2: Image obtained after overlapping of C and R 16
Figure 6.1: Registered Image 19
....
Figure 6.2: Input Image (X Rotated) 19
Figure 6.3: Converting Registered Image from 3D to 2D 19
Figure 6.4: Converting Input Image (X Rotated) from 3D to 2D 19
Figure 6.5: Normalized the Input Image (X Rotated) 19
Figure 6.6: Performing Histogram Equalization Registered Image 20
Figure 6.7: Performing Histogram Equalization on Input Image (X Rotated) 20
Figure 6.8: Performing Gaussian Filtering on Registered Image 20
Figure 6.9: Performing Gaussian Filtering on Input Image (X Rotated) 20
Figure 6.10: Contour of Registered Image 20
Figure 6.11: Contour of Input Image (X Rotated) 20
Figure 6.13: Input Image (Y Rotated) 21
Figure 6.15: Converting Input Image (Y Rotated) from 3D to 2D 21
Figure 6.16: Normalized the Input Image (Y Rotated) 21
Figure 6.18: Performing Histogram Equalization on Input Image (Y Rotated) 22
Figure 6.20: Performing Gaussian Filtering on Input Image (Y Rotated) 22
Figure 6.22: Contour of Input Image (Y Rotated) 22
Figure 6.24: Input Image (Z Rotated) 23
Figure 6.26: Converting Input Image (Z Rotated) from 3D to 2D 23
Figure 6.28: Normalized the Input Image (Z Rotated) 23
Figure 6.29: Performing Histogram Equalization on Input Image (Z Rotated) 24
Figure 6.31: Performing Gaussian Filtering on Input Image (Z Rotated) 24
Figure 6.33: Contour of Input Image (Z Rotated) 24
Figure 6.35: Input Image (Rotated Along More Than Two Axes) 25
Figure 6.37: Converting Input Image (Rotated Along More than Two Axis) from 3D to 2D 25
Figure 6.38: Normalized the Input Image (Rotated Along More than Two Axis) 25
Figure 6.40: Performing Histogram Equalization on Input Image (Rotated Along More Than Two Axes) 26
Figure 6.42: Performing Gaussian Filtering on Input Image (Rotated Along More Than Two Axes) 26
Figure 6.44: Contour of Input Image (Rotated Along More Than Two Axes) 26
Figure 6.46: Input Image (Image not in Database) 27
Figure 6.48: Converting Input Image (Image not in Database) 27
Figure 6.50: Performing Histogram Equalization on Input Image (Image not in Database) 27
Figure 6.52: Performing Gaussian Filtering on Input Image (Image not in Database) . 28
Figure 6.54: Contour of Input Image (Image not in Database) 28
Figure 8.1: GUI for Face Recognition 32
Figure 8.2: Selecting an image 33
Figure 8.3: After selection of image 33

Figure 8.4: Conversion of 3D to 2D. 34
Figure 8.5: Rotated along X-axis..35
Figure 8.6: Normalized along X-Axis 35
Figure 8.7: Rotated along Y-Axis...36
Figure 8.8: Normalized along Y-Axis 36
Figure 8.9: Rotated along Z-Axis 37
Figure 8.10: Normalized along Z-Axis...37
Figure 8.11: Final Output of GUI 38

CHAPTER 1

INTRODUCTION

1.1. Face Recognition: An Interesting Concept

Face recognition is interesting to study because it is an application area where computer vision research is being utilized in both military and commercial products. Much effort has been spent on this problem, yet there is still plenty of work to be done.

Basic research related to this field is currently active. Often, practical applications can grow out of improvements in theoretical understanding and it seems that this problem will continue to demonstrate this growth.

Personally, we are interested in this project because it's a pattern recognition problem in which humans are very adept, whereas it can be quite challenging to teach a machine to do it. The intermediate and final visual results are interesting to observe in order to understand failures and successes of the various approaches.

Face recognition is challenging because it is a real world problem. The human face is a complex, natural object that tends not to have easily (automatically) identified edges and features. Because of this, it is difficult to develop a mathematical model of the face that can be used as prior knowledge when analyzing a particular image.

Applications of face recognition are widespread. Perhaps the most obvious is that of human computer interaction. One could make computers easier to use if when one simply sat down at a computer terminal, the computer could identify the user by name and automatically load personal preferences. This identification could even be useful in enhancing other technologies such as speech recognition, since if the computer can identify the individual who is speaking, the voice patterns being observed can be more accurately classified against the known individual's voice.

Human face recognition technology could also have uses in the security domain. Recognition of the face could be one of several mechanisms employed to identify an individual. Face recognition as a security measure has the advantage that it can be done quickly, perhaps even in real time, and does not require extensive equipment to implement. It also does not pose a particular inconvenience to the subject being identified, as is the case in retinal scans. It has the disadvantage, however, that it is not a foolproof method of authentication, since human face appearance is subject to various sporadic changes on a day-to-day basis (shaving, hair style, acne, etc…), as well gradual changes over time (aging). Because of this, face recognition is perhaps best used as an augmentation for other identification techniques.
A final domain in which face recognition techniques could be useful is search engine technologies. In combination with face detection systems, one could enable users to search for specific people in images. This could be done by either having the user provide an image

of the person to be found, or simply providing the name of the person for well-known individuals. A specific application of this technology is criminal mug shot databases. This environment is perfectly suited for automated face recognition since all poses are standardized and lighting and scale are held constant. Clearly, this type of technology could extend online searches beyond the textual clues that are typically used when indexing information.

1.2. 2D Face Recognition

Every face or image that we can see is in its 2 dimensional form. There are many different technologies available today to uniquely identify a person's identity. Many of which like Password/PIN known as Personal Identification Number systems are the most common in practice today. However these systems have their own intrinsic drawbacks. Passwords can be forgotten and worse if they are lost or stolen, person identity can be misused by somebody else. In order to overcome these problems there has been a considerable interest in "biometrics" identification systems, which use pattern recognition techniques to identify people using their unique characteristics. Some of those methods are fingerprints and retina and iris recognition. But these are obtrusive and expensive.

2D face recognition has a natural place in the present and the future environment because it's unobtrusive and passive in nature. It does not restrict the movements of an individual during recognition.

2D face recognition gives best result when image is frontal and effect of illumination is negligible. If image is in profile view i.e. Image is rotated more than 20(degrees) then the recognition rate falls considerably.2-D face recognition algorithm are also prone to illumination effect and recognition rate below threshold value. Depth information is one of the most important parameter which mostly concern about eye, nose etc detection and recognition. We can not realize depth information and is simply ignored in 2-D view.

1.3. Need for 3D Face Recognition

A newly emerging trend, claimed to achieve previously unseen accuracies, is 3D.This technique uses 3-D sensors to capture information about the shape of a face. This information is then used to identify distinctive features on the surface of a face, such as the contour of the eye sockets, nose, and chin.

One advantage of 3-D facial recognition is that it is not affected by changes in lighting like other techniques. It can also identify a face from a range of viewing angles, including a profile view.

Even a perfect 3D matching technique could be sensitive to expressions. For that goal a group at the Technion applied tools from metric geometry to treat expressions as isometric.

One advantage of 3-D facial recognition is that it is not affected by changes in lighting like other techniques. It can also identify a face from a range of viewing angles, including a profile view.

Our goal at the end of the project is to develop 3D Human Face Recognition software which works efficiently and is simple to understand.

CHAPTER 2

LITERATURE SURVEY

2.1. Database

2.1.1. "Applying of Virtual Reality In MATLAB", Zhang Jiaxiang Luo Xueshan, Fang Lingjiang Mao Quansheng [1]

MATLAB is a mathematics software applied broad in high-powered numerical value calculation and control, while virtual reality technology is a high-powered visual and interactive simulation technology. This paper discusses that how to hang together between them. The Virtual Reality Toolbox is a solution for visualizing and interacting with dynamic systems in a 3-dimensional virtual reality environment. These dynamic systems are described with MATAB and Simulink.

The Virtual Reality Toolbox extends the capabilities of MATLAB and Simulink into the world of virtual reality graphics. Using standard VRML technology, we can create animated 3-dimensional scenes driven from the MATLAB and Simulink environment. We can observe a simulation of your dynamic system over time in a visually realistic 3D model. Most of the Virtual Reality Toolbox features can be implemented with S i m u l i blocks. Once you include these blocks in a Simulink diagram, they allow you to select the virtual world which you connect to Simulink signals. The Virtual Reality Toolbox automatically scans the virtual world for available VRML nodes that can be driven by Simulink.

2.2. Normalization

2.2.1. "MULTIMODAL 2D, 2.5D & 3D FACE VERIFICATION"
Cristina Conde, Angel Serrano and Enrique Cabello [2]

A multimodal face verification process is presented for standard 2D colour images, 2.5D range images and 3D meshes. Normalization in orientation and position is essential for 2.5D and 3D images to obtain a corrected frontal image. This is achieved using the spin images of the nose tip and both eyes, which feed an SVM classifier. First, a traditional Principal Component Analysis followed by an SVM classifier are applied to both 2D and 2.5D images. Second, an Iterative Closest Point algorithm is used to match 3D meshes. In all cases, the equal error rate is computed for different kinds of images in the training and test phases. In general, 2.5D range images show the best results (0.10% EER for frontal images). A special improvement in success rate for turned faces has been obtained for normalized 2.5D and 3D images compared to standard 2D images.

2.3. 3D to 2D Conversion

2.3.1. "3D-2D projective registration of free-form curves and surfaces"
J. Feldmar, N. Ayache and F. Betting [3]

Some medical interventions require knowing the correspondence between an MRI/CT pre - operative image and the actual position of the patient. Examples occur in neurosurgery, radiotherapy, interventional radiology, but also in video surgery (laparoscopy). The paper presents three new techniques for performing the task without artificial markers. The paper finds the 3D-2D projective transformation (composition of a rigid displacement and a perspective projection) which maps a 3D object onto a 2 0 image of this object. Depending on the object model (curve OT surface), and on the 2D image acquisition system (X-Ray, video), the techniques are different but the framework is common. It does not depend on the initial relative positions of the objects and deals with the occlusions and the outliers. Results are presented on real medical data to demonstrate the validity of OUT approach.

2.3.2. "Automatic 3D-2D image registration using partial digitally reconstructed radiographs along projected anatomic contours"
Xin Chen, Martin R. Varley, Lik-Kwan Shark, Glyn S. Shentall, Mike C. Kirby [4]

The paper presents a new intensity-based 3D-2D image registration algorithm for automatic pre-treatment validation in radiotherapy. The novel aspects of the algorithm includes a hybrid cost function developed based on partial digitally reconstructed radiographs (DRRs) generated along projected anatomic contours and level set for similarity measurement, and a fast search method developed based on parabola fitting and sensitivity based search order. Using CT and orthogonal X-ray images from a skull phantom, the proposed algorithm is compared with the conventional ray-casting full DRR based registration method. Not only is the algorithm shown to be computationally more efficient with registration time being reduced by a factor of 8, but also the algorithm is shown to offer 50% higher capture range allowing the initial patient displacement up to 15 mm (measured by mean target registration error) and high registration accuracy with average errors of 0.53 ± 0.12 mm for translation and 0.61° ±0.29° for rotation within the capture range.

2.4. Face Recognition

2.4.1. "Face Recognition Using Contour Matching"
S. T. Gandhe, K. T. Talele, and A.G.Keskar [5]

In this paper a contour matching based face recognition system is proposed, which uses "contour" for identification of faces. The feasibility of using contour matching for human face identification is presented through experimental investigation. The advantage of using contour matching is that the structure of the face is strongly represented in its description along with its algorithmic and computational simplicity that makes it suitable for hardware implementation. The input contour is matched with registered contour using simple matching algorithms. The proposed system of face recognition may be applied in identification systems.

Step involved in Face recognition system using Contour Matching is as follows:

1) Image Processing and Normalization.
2) Contour Generation.
3) Matching Algorithm.

2.4.2. Face Recognition: Features versus Templates
(Roberto Brunelli and Tomaso Poggio) [6]

The purpose of this paper is to compare two simple but general strategies on a common database. To develop and implement this, two new algorithms; the first one is based on the computation of a set of geometrical features, such as nose width and length, mouth position, and chin shape, and the second one is based on almost-grey-level template matching. The results obtained on the testing sets (about 90% correct recognition using geometrical features and perfect recognition using template matching) favour implementation of the template-matching approach.

2.5. Template Matching

Template matching is a technique in Digital image processing for finding small parts of an image which match a template image. It can be used in manufacturing as a part of quality control, a way to navigate a mobile robot, or as a way to detect edges in images. There are different approaches to accomplishing template matching. Some are more performing than others, and some find better matches.

The basic method of template matching is to loop through all the pixels in the search image and compare them to the pattern. While this method is simple to implement and understand, it is one of the slowest methods.
This method is normally implemented by firstly creating a subimage (the template), we will call this subimage 'w(x,y)' where x and y represent the coordinates of each pixel. We then simply move the center of this subimage w over each (x,y) point in the a candidate image, which we will call 'o(x,y)' and calculate the sum of products between the coefficients in o and the corresponding neighbourhood pixels in the area spanned by the filter mask. This method is sometimes referred to as 'Linear Spatial Filtering'.

This type of spatial filtering is normally only used in dedicated hardware solutions because of the computational complexity of the operation, however we can lessen this complexity by filtering it in the frequency domain of the image, referred to as 'frequency domain filtering,' this is done through the use of the convolution theorem.

Another way to make the matching faster is to reduce the image into smaller images, and then search the smaller subimages for a match to the template, these smaller images are often referred to as an image pyramid (i.e. a 128 x 128px image can have a pyramid of smaller images which are 64x64 32x32, 16x16 etc).
After finding matches in the smaller images, that information is used in the larger image as a centre location. The larger image is then searched in a small window to find the best location of the pattern.

CHAPTER 3

PROPOSED SYSTEM

3.1. Problem Statement

3.1.1. Broad problem statement

Be it for purposes of security or human–computer interaction, there is wide application to 3D face recognition. Recent developments in computer technology and the call for better security applications have turned face-recognition into focus. Face recognition systems have become the subject of increased interest.

Chances of unauthorised access to critical information have become a major concern. So a system must be implemented so that the security system becomes more robust and probability of unauthorized access to the system becomes minimum.

Given a training images from FRAV 3D database of human face models, we intend to design a system that would recognize the identity of a person from a test model of the person. The system is expected to have a high probability of detecting the correct person and minimal probability of false recognition. We are using a set of training models which are pre-processed (in scale, in alignment) for the human face recognition software.

3.1.2. Specific problem statement

Given a training database of pre-processed face models, normalise the image, convert that into 2D image, and train the human face recognition system to recognize the identity of a person from a test image of the person using contour matching technique.

3.2.Methodology

We implement 3D Face Recognition using Counter Matching

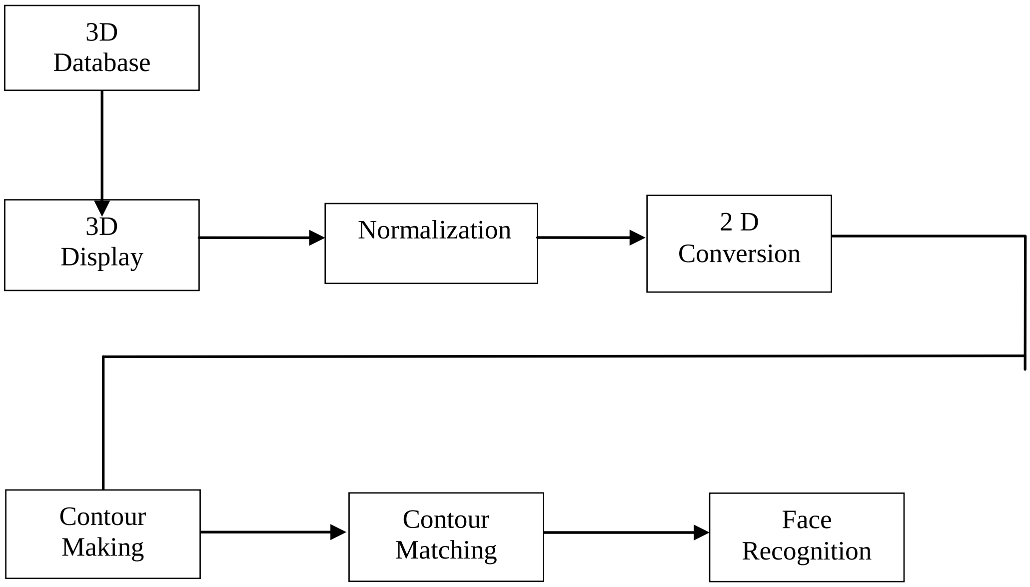


Figure 3.1: Block diagram for template matching

3.2.1 For Face Recognition

The entire process is divided into three major steps:

- Image processing and normalization
- Converting the Model into 2D image
- Face Contour Matching and Recognition.

The image of the person i.e. the model of the person is taken from the database created by a Face Recognition and Artificial Vision Group also known as the FRAV3D database.

3.2.1.1 FRAV3D Database

We have used a multimodal database, the so-called FRAV3D. It has been acquired during ten months with 105 volunteers. The totality of the subjects are young adults (18 - 35 years old), Caucasian, with a certain bias towards men (81 males/24 women). A scanner MINOLTA VIVID-700 red laser light-stripe triangulation range finder was used under controlled indoor conditions. As a result, both a 3D mesh with up to 4000 points and 7500 triangles and a classical 2D colour image were produced. The subjects were asked to sit opposite the scanner, with a dark plain background behind them. No hats, scarves or glasses were allowed. For security reasons, all the participants kept their eyes closed during the acquisition. All scans were acquired using a strict protocol for standardizing reasons. Each shot differed from the previous one in only one acquisition parameter, which included turns, presence or absence of gestures and changes in illumination.

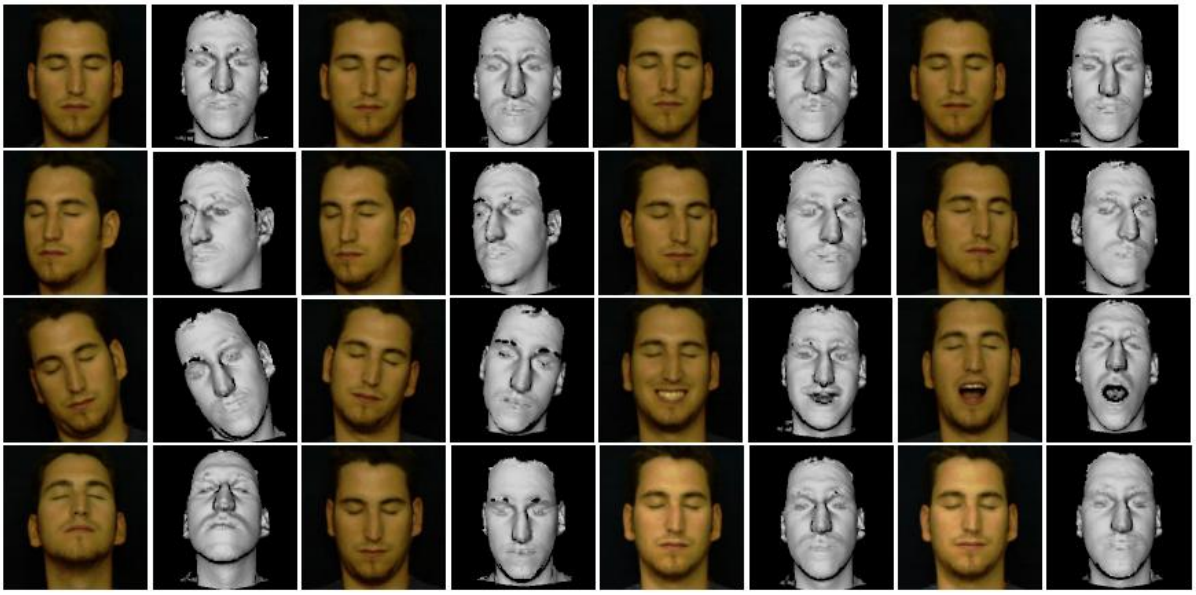


Figure 3.2: From left to right, top to bottom, the acquisition sequence of a subject is displaced. Both BMP color images and VRML 3D meshes are shown.

All the images provided to us in FRAV3D database are VRML (Virtual Reality Modelling Language) 3D Meshes of '.wrl' format. Version of VRML is v1.0.

Then the image is normalised. The face area is then analysed for contours. These contours are then compared with the contours of the people in the database. This comparison is done using Template matching technique. A threshold value is set and the value of matching ratio is compared with the threshold value. If the value exceeds the threshold limit then it is said to be matched and face is recognized.

3.2.2 For User Authentication System

The above described process is only used with a few modifications. For user authentication we assume that all images of a single person are in the database. The system then calculates the adaptive threshold of those images. Therefore, when user authentication is performed the matching ratio of the input image with each image in the database is found and the highest matching ratio is compared with the calculated threshold and the image is accordingly accepted or rejected.

CHAPTER 4

REQUIREMENTS SPECIFICATION

This Software and Hardware Requirements Specification provides a complete description of all the specifications of Face Recognition software.

4.1.Software Platform

- Compatible Operating Systems

    - ✓ Windows XP Professional
    - ✓ Windows 2000 Professional
    - ✓ Windows NT/ME
    - ✓ Windows 98

- Versions of MATLAB

    - ✓ MATLAB version 7.0 or 8.0 installed.

4.2.Hardware Platform

The minimum required configuration:

- 1Ghz or higher processor
- 512 MB of RAM (1 GB Recommended)
- 500 MB of hard disk space

4.3.Validation Criteria

The images to be selected must be of the extension .wrl and the time for recognition is to be tolerated by the user

CHAPTER 5

ALGORITHMS

5.1. Reading of Input Image

The input image which is in .wrl format is written in VRML language.

1. Identify the markers 'point' followed by '[' .
2. Read data(coordinates) which is followed by markers.
3. Stop reading data when ']' marker is identified.
4. Store the data read in an array.
5. Plot the coordinates using 'plot3' function in Matlab.

5.2. Normalization of 3D input image.

Normalization of image is done by using Transformation Matrix.

5.2.1. Normalization along X- Axis

1. Get the angle Θ.
2. Give Θ as input to the Transformation Matrix.
3. Transformation Matrix is given by:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix}$$

4. Display the output.

5.2.2. *Normalization along Y-Axis*

1. Get the angle Θ.
2. Give Θ as input to the Transformation Matrix.
3. Transformation Matrix is given by:

$$\begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}$$

4. Display the output.

5.2.3. Normalization along Z-Axis

1. Get the angle Θ.

2. Give Θ as input to the Transformation Matrix.
3. Transformation Matrix is given by:

$$\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

4. Display the output.

5.3. 3D to 2D Conversion

The perspective projection requires greater definition. A conceptual aid to understanding the mechanics of this projection involves treating the 2D projection as being viewed through a camera viewfinder. The camera's position, orientation, and field of view control the behaviour of the projection transformation. The following variables are defined to describe this transformation:

- $\mathbf{a}_{x,y,z}$ - the point in 3D space that is to be projected.
- $\mathbf{c}_{x,y,z}$ - the location of the camera.
- $\theta_{x,y,z}$ - The rotation of the camera. When $\mathbf{c}_{x,y,z}$=<0,0,0>, and $\theta_{x,y,z}$=<0,0,0>, the 3D vector <1,2,0> is projected to the 2D vector <1,2>.
- $\mathbf{e}_{x,y,z}$ - the viewer's position relative to the display surface.

Which results in:

- $\mathbf{b}_{x,y}$ - the 2D projection of $\mathbf{a}$.

First, we define a point $\mathbf{d}_{x,y,z}$ as a translation of point $\mathbf{a}$ into a co-ordinate system defined by$\mathbf{c}$. This is achieved by subtracting $\mathbf{c}$ from $\mathbf{a}$ and then applying a vector rotation matrix using $-\theta$ to the result. This transformation is often called a camera transform (note that these calculations assume a left handed system of axes):

$$\begin{bmatrix} \mathbf{d}_x \\ \mathbf{d}_y \\ \mathbf{d}_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_x & -\sin\theta_x \\ 0 & \sin\theta_x & \cos\theta_x \end{bmatrix} \begin{bmatrix} \cos\theta_y & 0 & \sin\theta_y \\ 0 & 1 & 0 \\ -\sin\theta_y & 0 & \cos\theta_y \end{bmatrix} \begin{bmatrix} \cos\theta_z & -\sin\theta_z & 0 \\ \sin\theta_z & \cos\theta_z & 0 \\ 0 & 0 & 1 \end{bmatrix} \left(\begin{bmatrix} \mathbf{a}_x \\ \mathbf{a}_y \\ \mathbf{a}_z \end{bmatrix} - \begin{bmatrix} \mathbf{c}_x \\ \mathbf{c}_y \\ \mathbf{c}_z \end{bmatrix} \right)$$

Or, for those less comfortable with matrix multiplication. Signs of angles are inconsistent with matrix form:

The transformed point can then beprojected onto the 2D plane

$$
\begin{aligned}
\mathbf{b}_x &= (\mathbf{d}_x - \mathbf{e}_x)(\mathbf{e}_z/\mathbf{d}_z) \\
\mathbf{b}_y &= (\mathbf{d}_y - \mathbf{e}_y)(\mathbf{e}_z/\mathbf{d}_z)
\end{aligned}.
$$

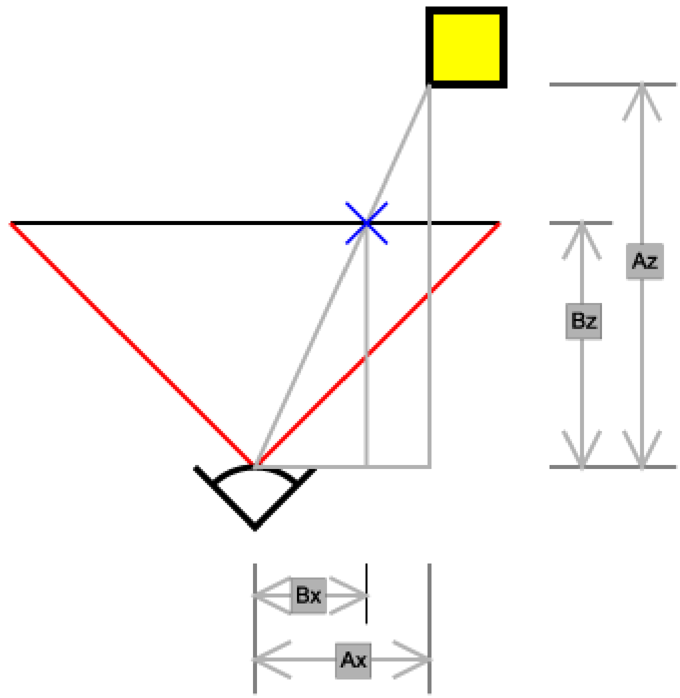
$$
\begin{bmatrix} \mathbf{f}_x \\ \mathbf{f}_y \\ \mathbf{f}_z \\ \mathbf{f}_w \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -\mathbf{e}_x \\ 0 & 1 & 0 & -\mathbf{e}_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/\mathbf{e}_z & 0 \end{bmatrix} \begin{bmatrix} \mathbf{d}_x \\ \mathbf{d}_y \\ \mathbf{d}_z \\ 1 \end{bmatrix}
$$

in conjunction with an argument using similar triangles, leads to division by the homogeneous coordinate, giving

$$
\begin{aligned}
\mathbf{b}_x &= \mathbf{f}_x/\mathbf{f}_w \\
\mathbf{b}_y &= \mathbf{f}_y/\mathbf{f}_w
\end{aligned}.
$$

The distance of the viewer from the display surface, $\mathbf{e}_z$, directly relates to the field of view, where $\alpha = 2 \cdot \tan^{-1}(1/\mathbf{e}_z)$ is the viewed angle. (Note: This assumes that you map the points (-1,-1) and (1,1) to the corners of your viewing surface)

Subsequent clipping and scaling operations may be necessary to map the 2D plane onto any particular display media.



To determine which screen x coordinate corresponds to a point at Ax,Az multiply the point coordinates by:

$$\text{screen x coordinate}(Bx) = \text{model x coordinate}(Ax) \times \frac{\text{distance from eye to screen}(Bz)}{\text{distance from eye to point}(Az)}$$

the same works for the screen y coordinate:

$$\text{screen y coordinate}(By) = \text{model y coordinate}(Ay) \times \frac{\text{distance from eye to screen}(Bz)}{\text{distance from eye to point}(Az)}$$

5.4. Algorithms for Face Recognition using Contour Matching

5.4.1. Variables:

- ✓ fnumber: The number of stored image contours in the database.
- ✓ R: The contour of stored images.

5.4.2. Input Image

Step 1: For obtaining Contour of an image

- ✓ Perform Histogram Equalization on the image
- ✓ Perform Gaussian Blurring on the equalized image
- ✓ Perform Pixel Value Normalization on the blurred image
- ✓ Obtain contour (level 1) of this image

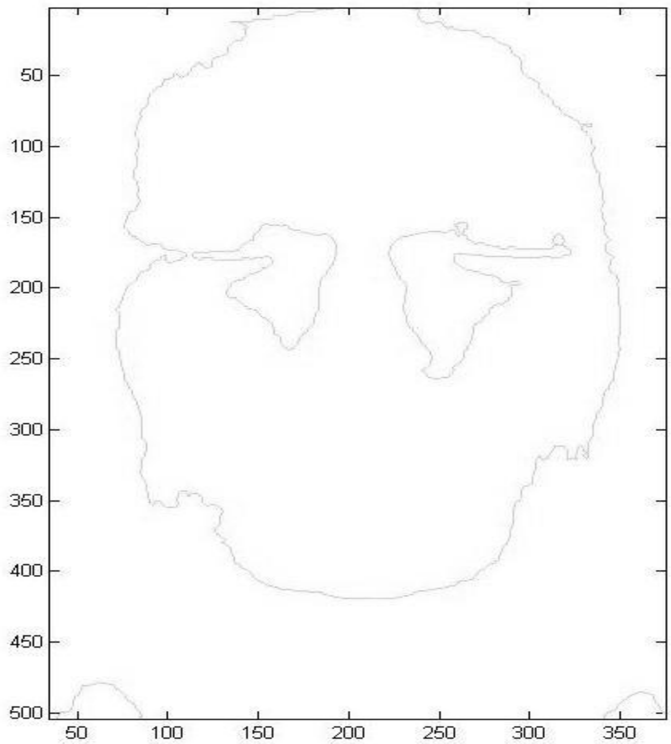


Figure 5.1: Contour of human face

Step 2: For Face Recognition

- ✓ Obtain Contour of the input image and store it in C
- ✓ Select an image from the database
- ✓ Obtain the contour of the image and store it in R
- ✓ C is slid across R in top to bottom and left to right fashion
- ✓ Let f(i,j) and g(k,l) be the pixels at position (i, j) and (k, l) in C and R respectively. (Alpha and Beta measure the horizontal and vertical displacement between the two pixels)
- ✓ To match different images of same person a neighbourhood of 21x21 is considered with f(i,j) at center. For every black pixel at f(i,j), if there exists a black pixel in the neighbourhood of 21x21 pixel, it is considered a match

$$h^{(t)}_{i,j}(\hat{\alpha}, \hat{\beta}) \triangleq \max_{\alpha,\beta} \sum_{i,j} h^{(t)}_{i,j}(\alpha, \beta)$$

$$h^{(t)}_{i,j}(\alpha, \beta) = \Phi\left[\sum_{x=-2}^{2} \sum_{y=-2}^{2} (f^{(t)}_{i,j} \cdot g^{(t)}_{k+x,l+y})\right]$$

$$k = i \pm \alpha,\ l = j \pm \beta,$$
$$\alpha = 0, 1, 2, \ldots 256,\ \beta = 0, 1, 2, \ldots 240$$

$$\Phi[x] = \begin{cases} 1 & \text{for } x \geq 1 \\ 0 & \text{for } x = 0. \end{cases}$$

Figure 5.2: Image obtained after overlapping of C and R

- ✓ The above two steps are repeated for all pixels in C and an overlapping contour is obtained and stored in H.
- ✓ For different images of same person long segments of overlapping contour are obtained in H. But for images of different person's sometimes small segments are obtained in H by chance, which may result in high matching ratio. These small segments are therefore removed and matching ratio is calculated as follows,

$$H(f, g) = \sum_{t=1}^{8} H_{f,g}(t) \cdot W^{(t)}_{i,j}$$

$$H_{f,g}(t) = \frac{2}{F^{(t)} + G^{(t)}} \cdot \sum_{i,j} h^{(t)}_{i,j}(\hat{\alpha}, \hat{\beta})$$

$$h^{(t)}_{i,j}(\hat{\alpha}, \hat{\beta}) = 0 \quad \text{if} \quad h^{(t)}_{i,j}(\hat{\alpha}, \hat{\beta}) \in S^{(n)}(t; l < \theta_t)$$

- ✓ Repeat above procedure till all images in database are selected
- ✓ Obtain the image matched with input image with maximum matching ratio.

- ✓ If this matching ratio exceeds the threshold the two images are said to be of same person.

5.5. Algorithm for User Authentication System

Step1: Database

- ✓ The frontal image of authenticated person is loaded in the database
- ✓ The first image in database is matched with every other image in database (using the algorithm explained above) and matching ratios are obtained
- ✓ Mean and median of range of matching ratios obtained is calculated
- ✓ Maximum of mean and median is considered as threshold for this authenticated person

Step 2: To check for Authentication

- ✓ Select the image to be authenticated
- ✓ Compare the image with every image in database and obtain matching ratio.
- ✓ For any match, if the matching ratio exceeds the threshold obtained as above the person is consider to be authorized
- ✓ For all matches, if none of the matching ratio exceeds the threshold the person is considered to be an intruder

CHAPTER 6
RESULTS

6.1. For Acceptance: Input Image Rotated Along X Axis

Input image taken here is an image rotated along x-axis. Here we normalize the image along x-axis pre-process it and give it as input for recognition.

Figure 6.1: Registered Image

Figure 6.2: Input Image (X Rotated)

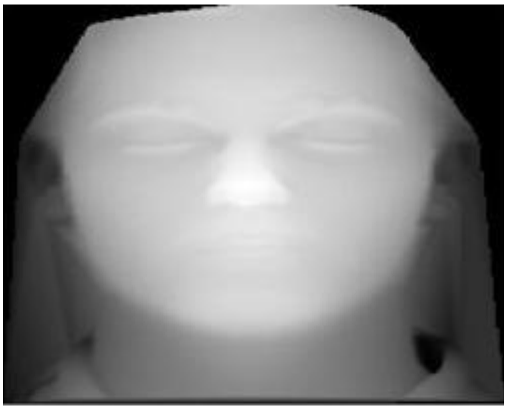


Figure 6.3: Converting Registered Image from 3D to 2D

Figure 6.4: Converting Input Image from 3D to 2D

Figure 6.5: Normalised the Input Image

Figure 6.6: Performing Histogram Equalization Registered Image

Figure 6.7: Performing Histogram Equalization on on Input Image

Figure 6.8: Performing Gaussian Filtering on Registered Image

Figure 6.9: Performing Gaussian Filtering on Input Image

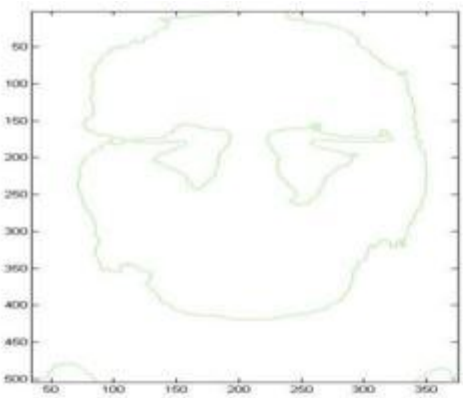


Figure 6.10: Contour of Registered Image

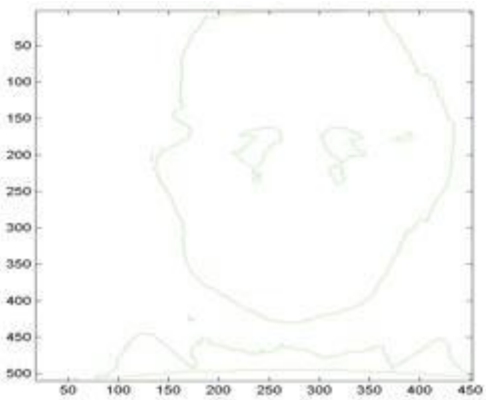


Figure 6.11: Contour of Input Image

Matching Ratio=0.4462 i.e. Accepted

6.2. For Acceptance: Input Image Rotated Along Y Axis

Input image taken here is an image rotated along y-axis. Here we normalize the image along y-axis pre-process it and give it as input for recognition.

Figure 6.12: Registered Image

Figure 6.13: Input Image

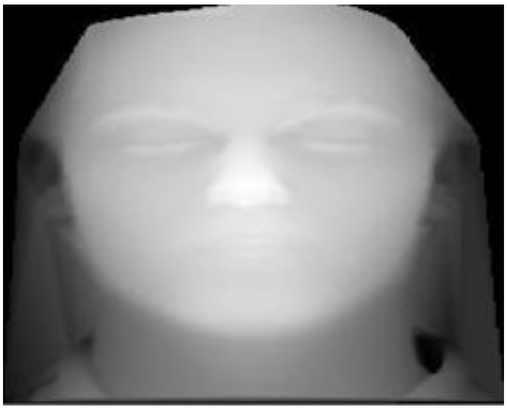


Figure 6.14: Converting Registered Image from 3D to 2D

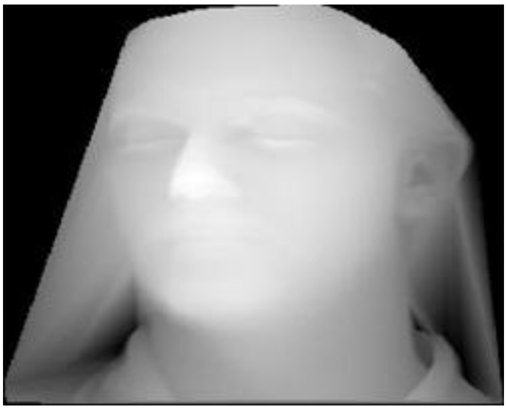


Figure 6.15: Converting Input Image from 3D to 2D

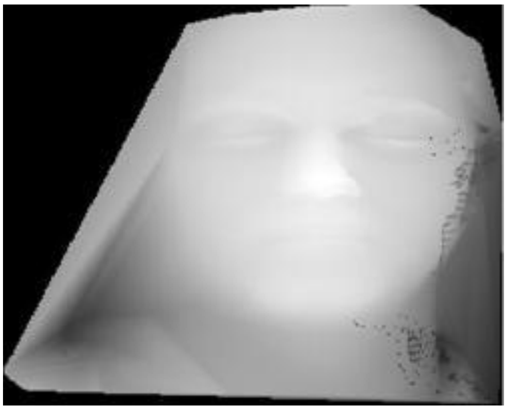


Figure 6.16: Normalised the Input Image

Figure 6.17: Performing Histogram Equalization on Registered Image

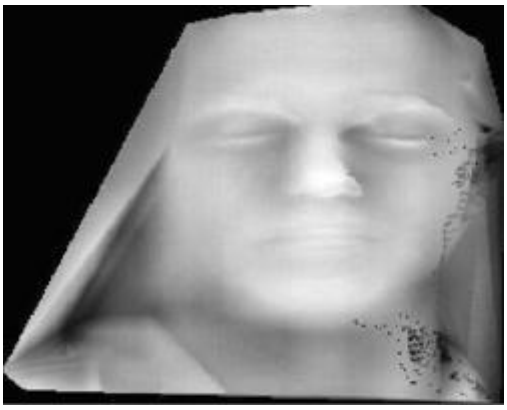


Figure 6.18: Performing Histogram Equalization on Input Image

Figure 6.19: Performing Gaussian Filtering on Registered Image

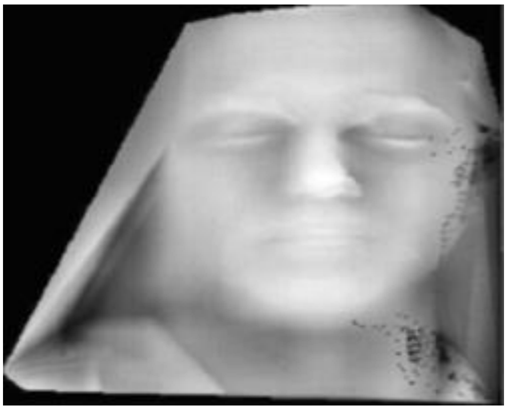


Figure 6.20: Performing Gaussian Filtering on Input Image

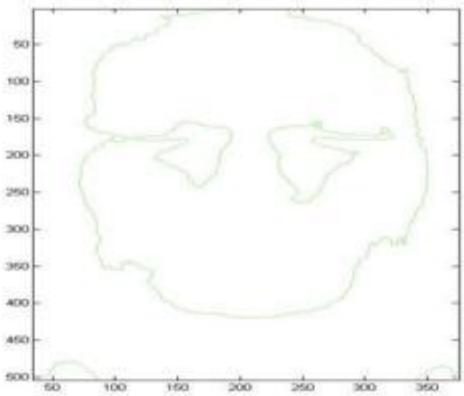


Figure 6.21: Contour of Registered Image

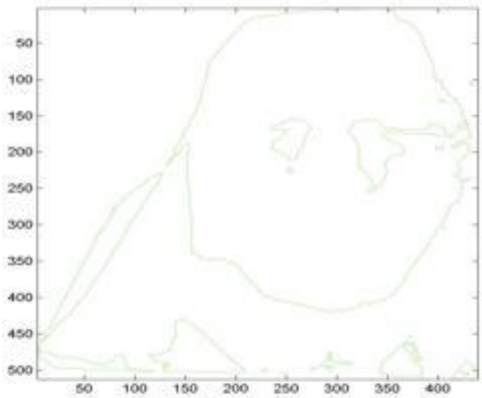


Figure 6.22: Contour of Input Image

Matching Ratio=0.4173 i.e. Accepted

6.3. For Acceptance: Input Image Rotated Along Z Axis

Input image taken here is an image rotated along z-axis. Here we normalize the image along z-axis pre-process it and give it as input for recognition.

Figure 6.23: Registered Image

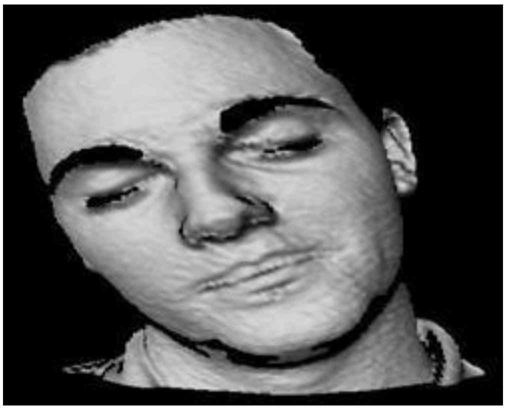


Figure 6.24: Input Image

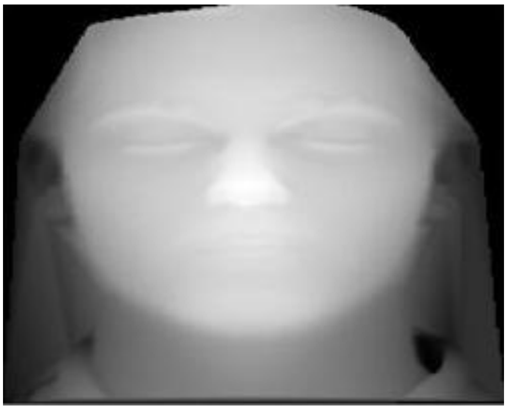


Figure 6.25: Converting Registered Image from 3D to 2D

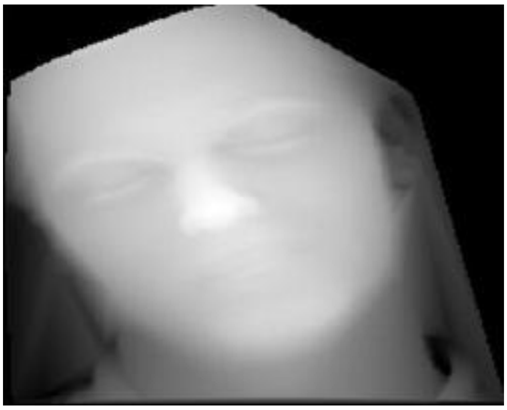


Figure 6.26: Converting Input Image from 3D to 2D

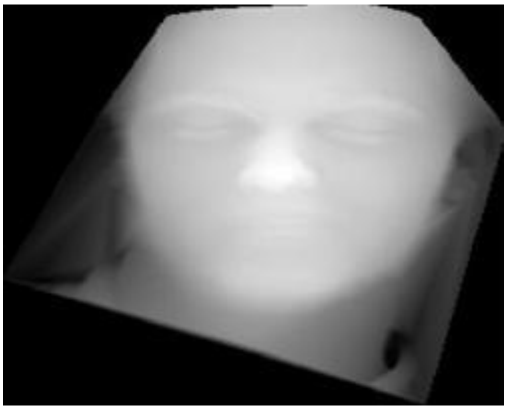


Figure 6.27: Normalised the Input Image

Figure 6.28: Performing Histogram Equalization on Registered Image

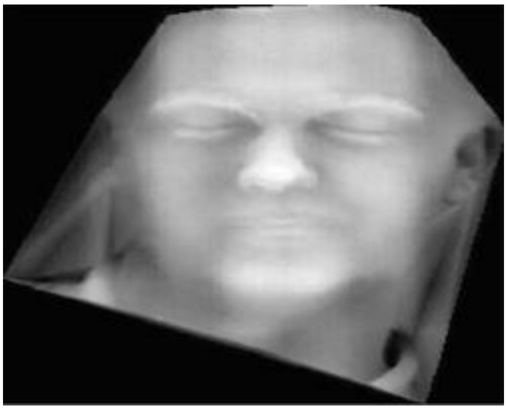


Figure 6.29: Performing Histogram Equalization on Input Image

Figure 6.30: Performing Gaussian Filtering on Registered Image

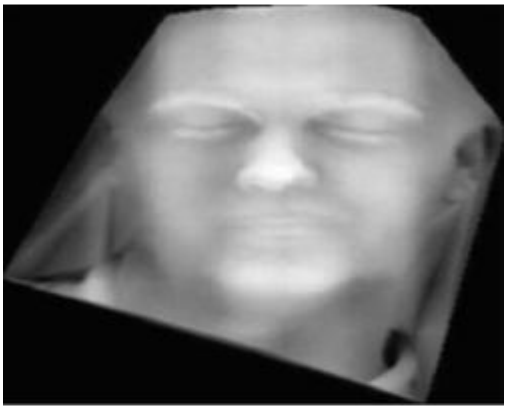


Figure 6.31: Performing Gaussian Filtering on Input Image

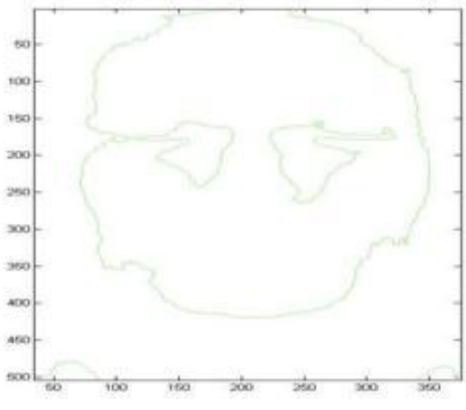


Figure 6.32: Contour of Registered Image

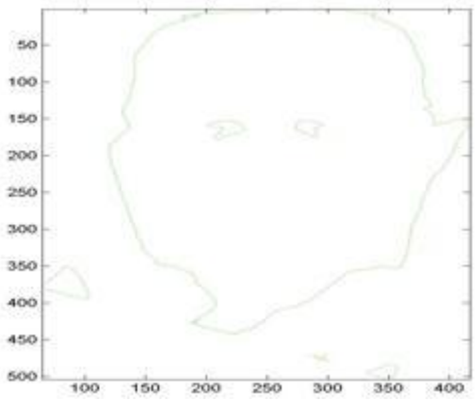


Figure 6.33: Contour of Input Image

Matching Ratio=0.3985 i.e. Accepted

6.4. For Acceptance: Input Image Rotated Along More than Two Axis

Input image taken here is an image rotated along y and x-axis. Here we normalize the image along x-axis pre-process it and give it as input for recognition.

Figure 6.34: Registered Image

Figure 6.35: Input Image

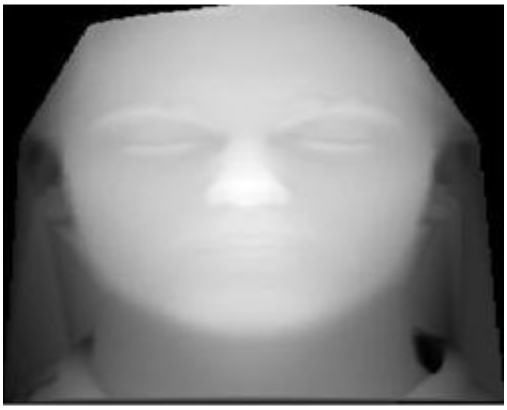


Figure 6.36: Converting Registered Image from 3D to 2D

Figure 6.37: Converting Input Image from 3D to 2D

Figure 6.38: Normalised the Input Image

Figure 6.39: Performing Histogram Equalization on Registered Image

Figure 6.40: Performing Histogram Equalization on Input Image

Figure 6.41: Performing Gaussian Filtering on Registered Image

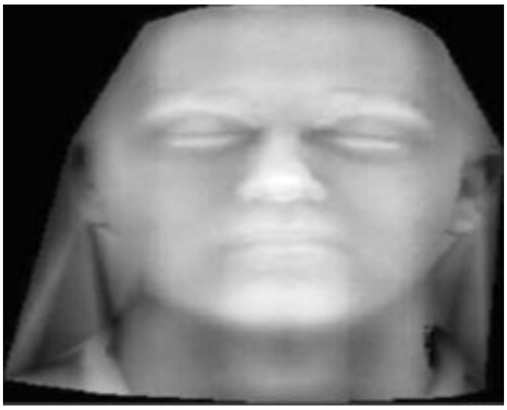


Figure 6.42: Performing Gaussian Filtering on Input Image

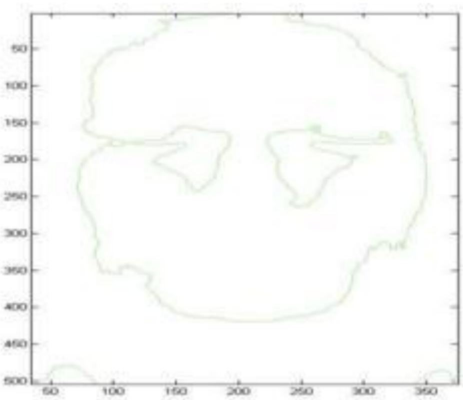


Figure 6.43: Contour of Registered Image

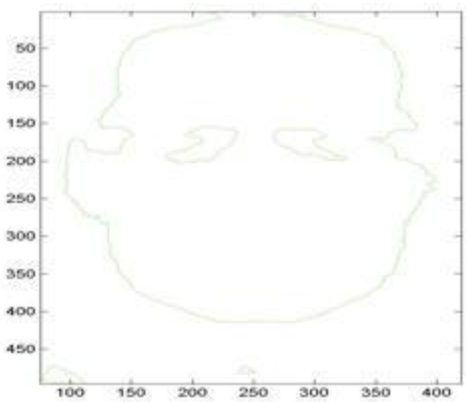


Figure 6.44: Contour of Input Image

Matching Ratio=0.4254 i.e. Accepted

6.5. For Rejection: Input Image compared with Image not in Database.

Figure 6.45: Registered Image

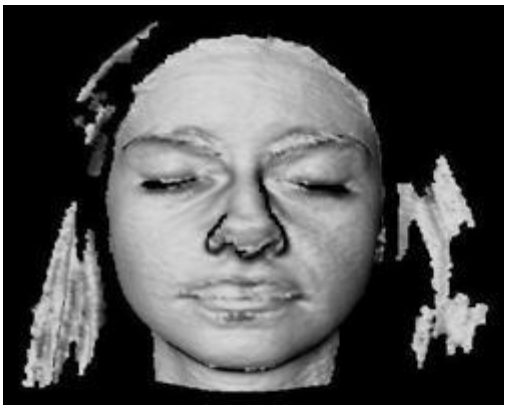


Figure 6.46: Input Image

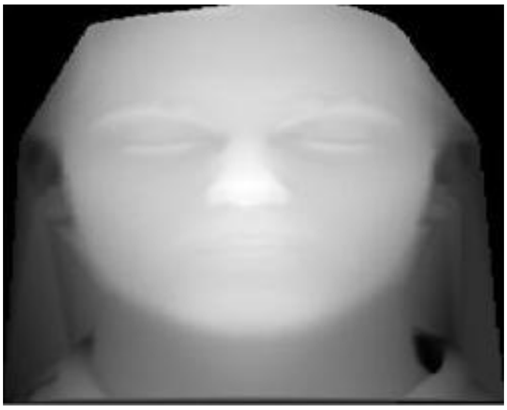


Figure 6.47: Converting Registered Image from 3D to 2D

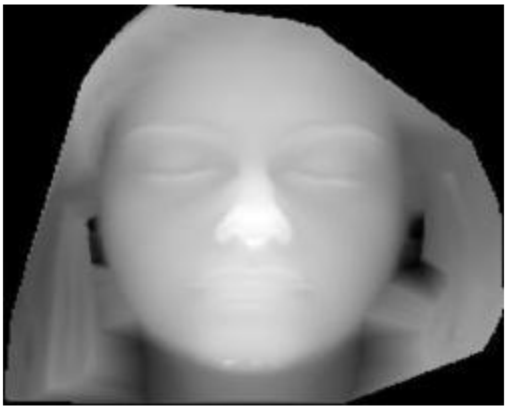


Figure 6.48: Converting Input Image from 3D to 2D

Figure 6.49: Performing Histogram Equalization on Registered Image

Figure 6.50: Performing Histogram Equalization on Input Image

Figure 6.51: Performing Gaussian Filtering on Registered Image

Figure 6.52: Performing Gaussian Filtering on Input Image

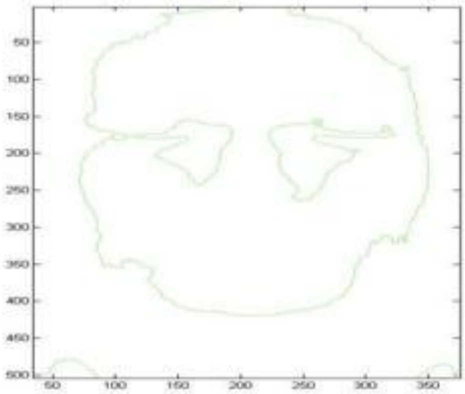


Figure 6.53: Contour of Registered Image

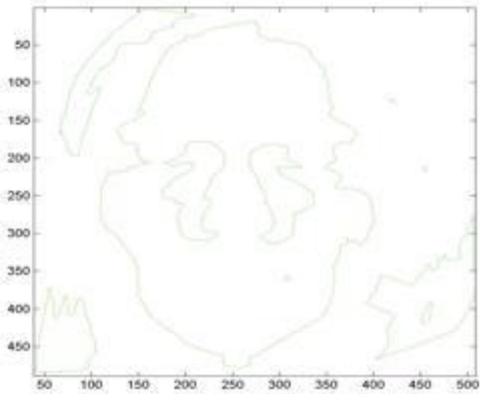


Figure 6.54: Contour of Input Image

Matching Ratio=0.2933 i.e. Rejected

CHAPTER 7

EXPERIMENTATION AND ITS ANALYSIS

7.1.Performance Parameters

7.1.1. Matching Ratio

Matching Ratio can be defined as the measure of similarity between two images where matching ratio 1 denotes an exactly similar image.

7.1.2. False Acceptance Rate (FAR)

In biometrics, the instance of a security system incorrectly verifying or identifying an unauthorized person is known as false acceptance. Also referred as a *type II error*, a false acceptance typically is considered the most serious of biometric security errors as it gives unauthorized users access to systems that expressly are trying to keep them out.

The *false acceptance rate*, or FAR, is the measure of the likelihood that the biometric security system will incorrectly accept an access attempt by an unauthorized user.

$$FAR = \frac{Number\ of\ false\ acceptances}{Number\ of\ identification\ attempts}$$

7.1.3. False Rejection Rate (FRR)

In biometrics, the instance of a security system failing to verify or identify an unauthorized person is known as false rejection. Also referred as a *type I error*, a false rejection does not necessarily indicate a flaw in the biometric system for example, in face recognition base system a blurred image can cause a false rejection of the authorized user.

The *false rejection rate,* or FRR, is the measure of the likelihood that the biometric security system will incorrectly reject an access attempt by an unauthorized user.

$$FAR = \frac{Number\ of\ false\ rejection}{Number\ of\ identification\ attempts}$$

7.1.4. Recognition Rate (RR)

The recognition rate, or RR, is the measure that the biometric security system will correctly accept an access attempt by an authorized user.

$$RR = \frac{Number\ of\ correct\ recognitions}{Number\ of\ identification\ attempts}$$

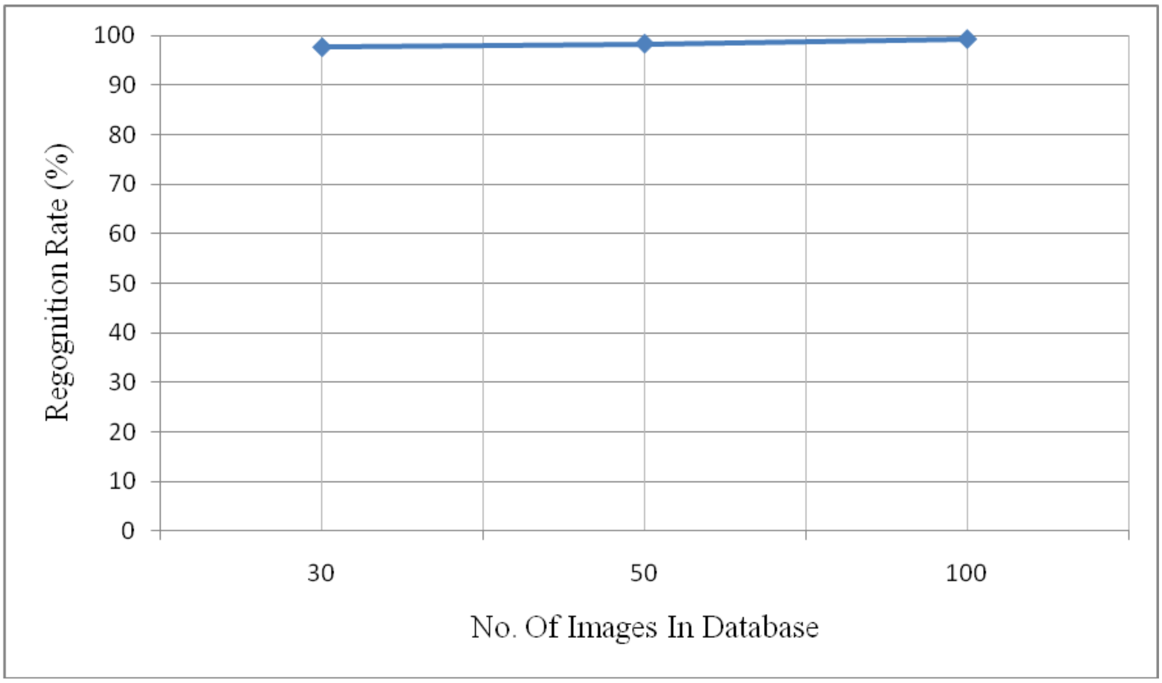
7.2.Experimentation

7.2.1. Database

Our database contains a total of 160 images, 16 each of 10 individuals. The images are in '.wrl' format. The threshold value for the recognition is set to 0.35 for the experimentation purpose.

7.2.2. Experiment

- Set 1: In this set 30 images are loaded in database. Remaining 130 images are used for testing.
- Set 2: In this set 50 images are loaded in database. Remaining 110 images are used for testing.
- Set 3: In this set 100 images are loaded in database. Remaining 60 images are used for testing.
- Graphs



Graph 1: Recognition Rate v/s No. Of Images In Database

Graph 2: False Acceptance Rate v/s No. Of Images In Database

Graph 3: False Rejection Rate v/s No. Of Images In Database

- Result Analysis

  - ✓ From the graphs shown above we see that the efficiency increases as we increase the no. of images present in the database.
  - ✓ The efficiency for 30 images in database is 97.6 %.
  - ✓ The efficiency for 50 images in database is 98.3 %.
  - ✓ The efficiency for 100 images in database is 99.2 %.

CHAPTER 8
USER MANUAL

8.1. Face Recognition Manual

8.1.1. To start the process of Face Recognition on Images

- Install MATLAB R2008 on your PC and then run MATLAB program
- Open the initial2.m in the folder 'Final'
- Run the code
- This will show the 'Face Recognition' GUI on your screen

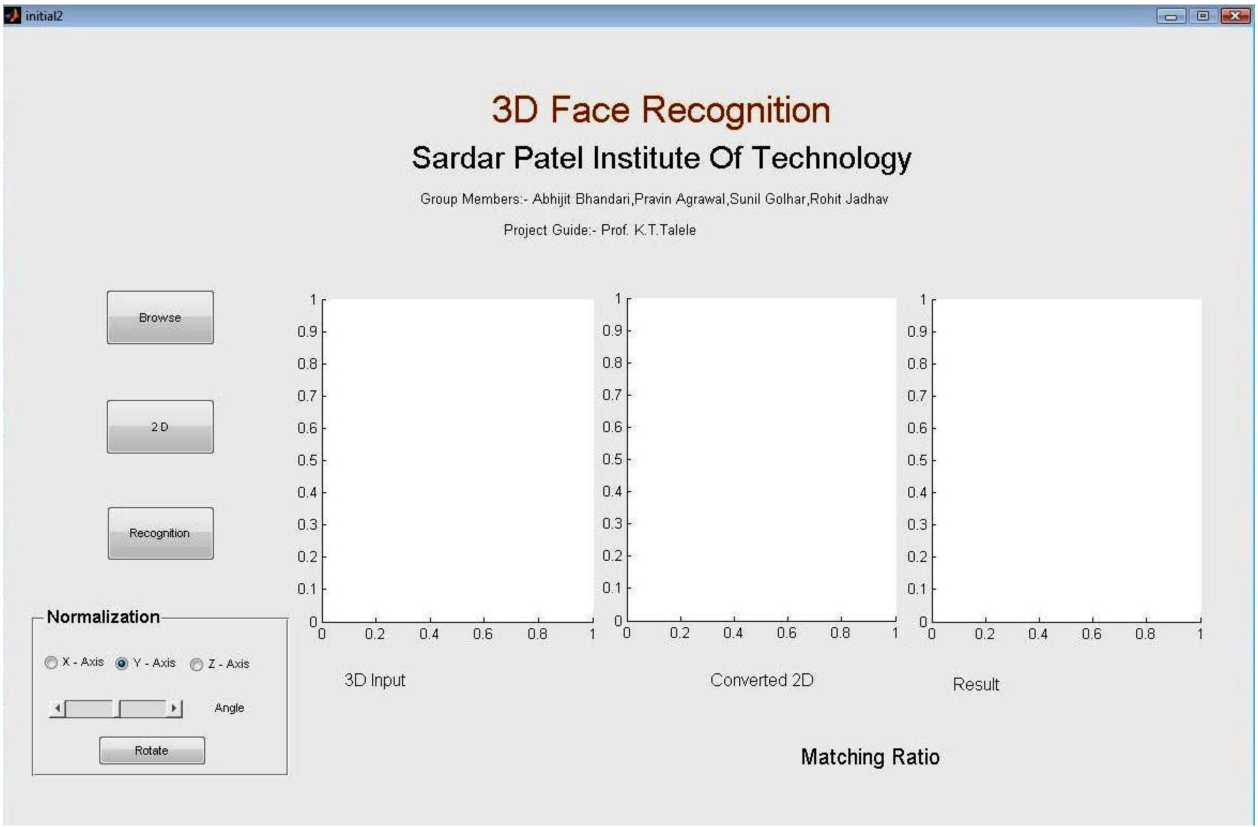


Figure 8.1: GUI for Face Recognition

8.1.2. *To Select An Image*

- You can select an image (for face recognition) by clicking on the pushbutton **Browse**
- A browser window appears on the screen to select an image with the extension .wrl

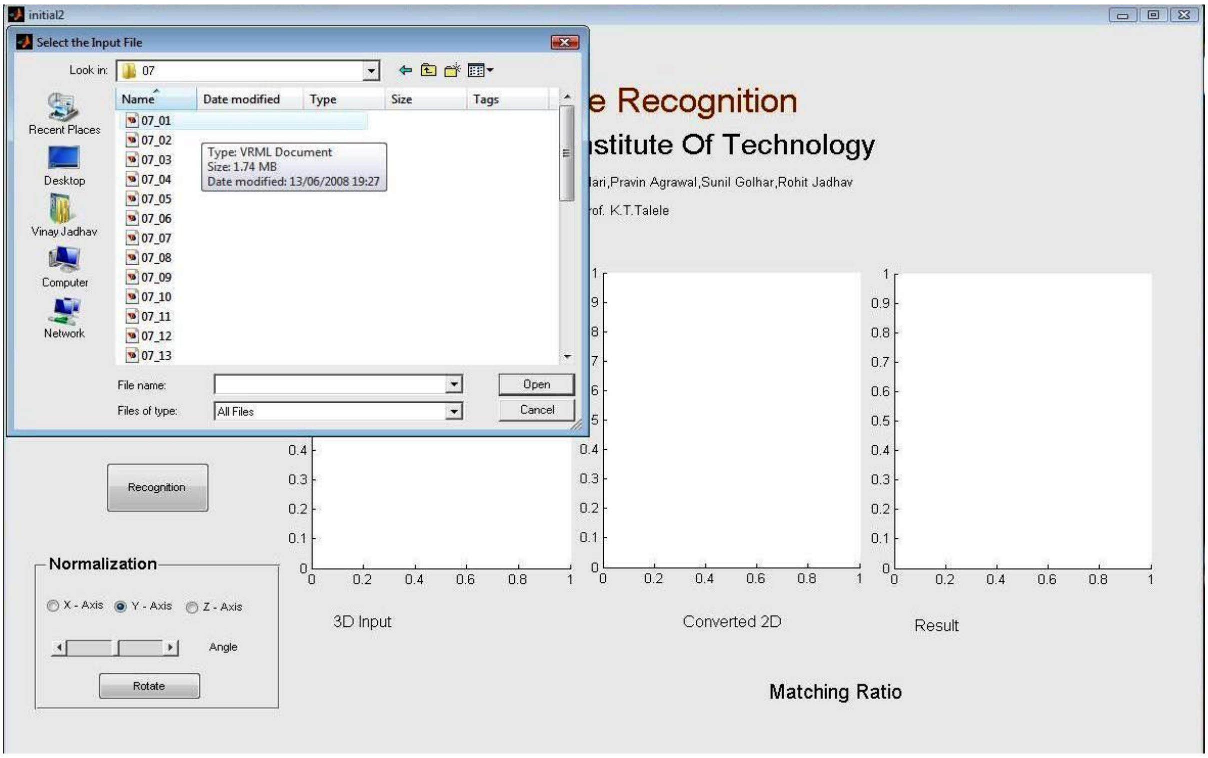


Figure 8.2: Selecting an image

- Select the required image. The image appears on the GUI as shown below

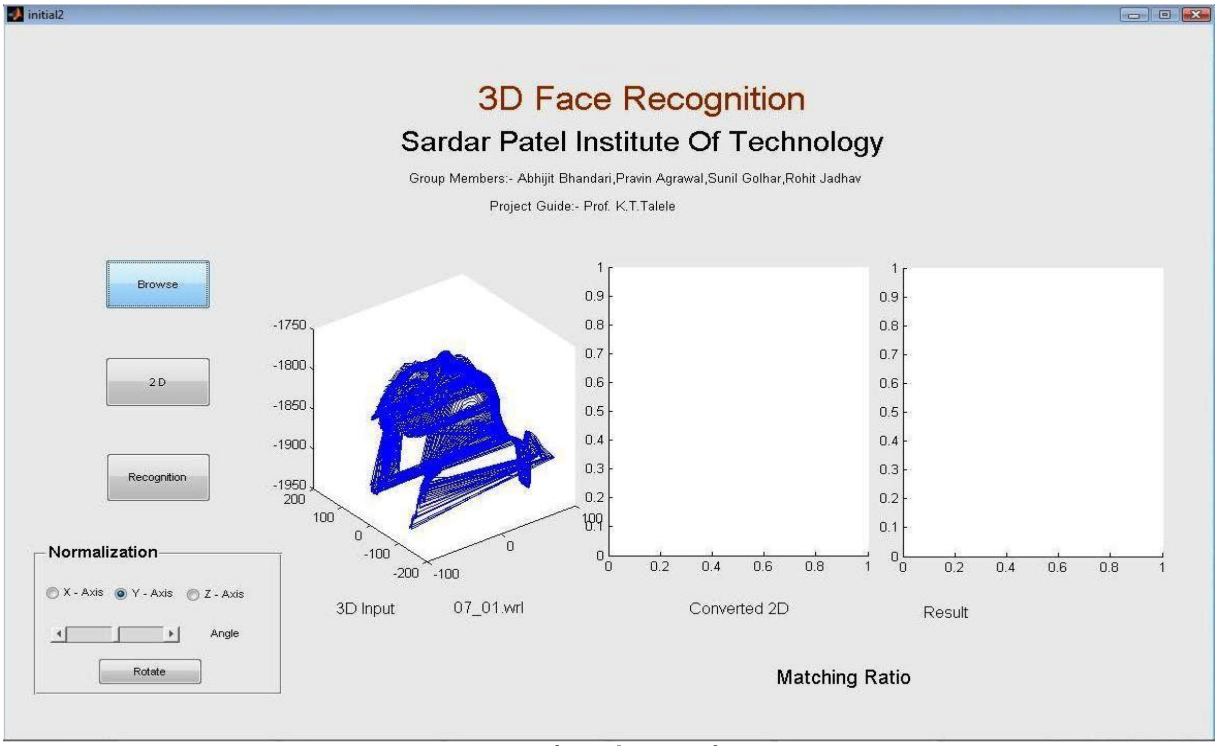


Figure 8.3: After selection of image

8.1.3. *To convert the 3D input Image into 2D and Display it*

- Now after selecting an image, the next step is to convert the input image into its 2D equivalent. In order to do so click on 2D button.
- After clicking the 2D button 'to2d.m' file will run.
- It will convert the 3D input model into a 2D image.
- Also the white space occurring after conversion into 2D is removed by cropping the output image and then displaying it.
- The output of this stage is as shown below.

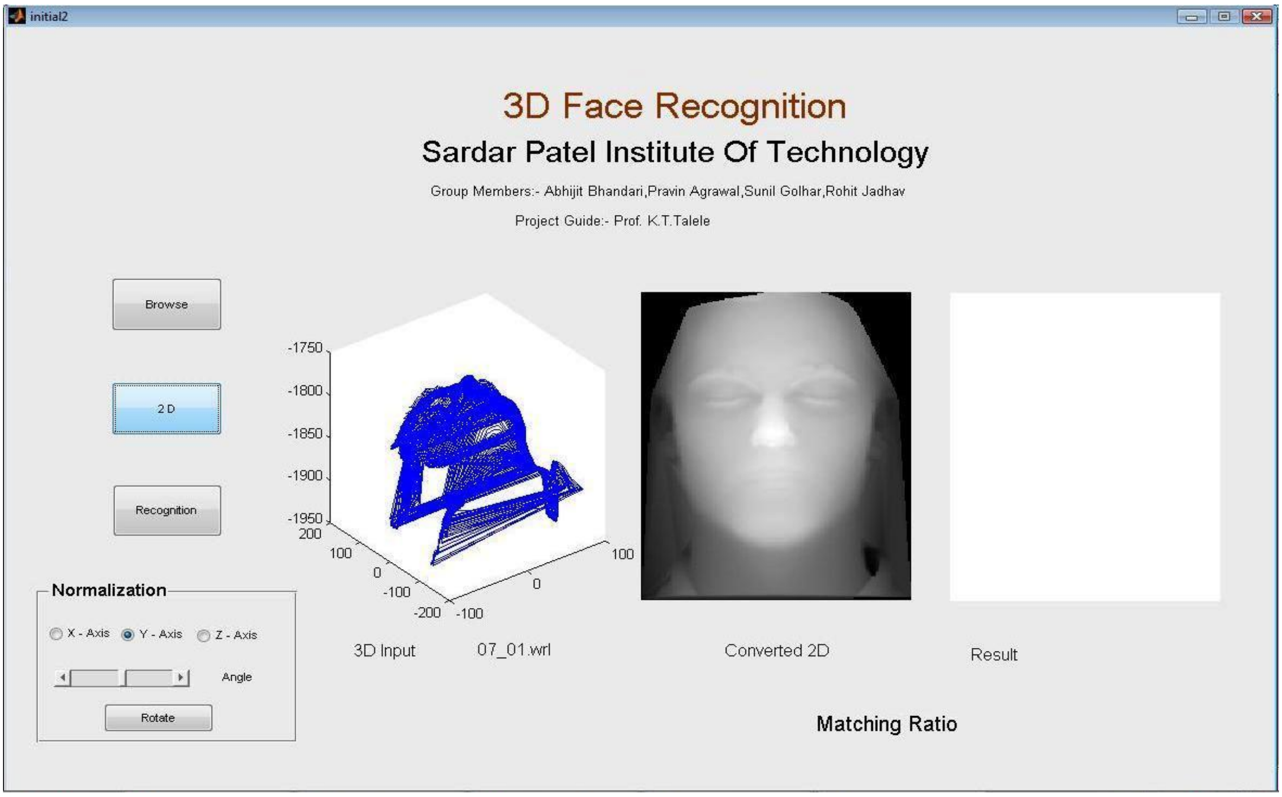


Figure 8.4: Conversion of 3D to 2D.

8.1.4. *Normalization of image along X-Y-Z Axis*

- Here we are using manual normalisation of image.
- In order to rotate an image along particular axis just select the respective radio button on the GUI and move the slider to a particular angle by which you wish to rotate the image.
- After setting up an angle press the Rotate Button.
- The 3D image gets rotated along that axis by the angle set and its 2D converted image is displayed upon the previous 2D Image.

a. Normalization along X-Axis

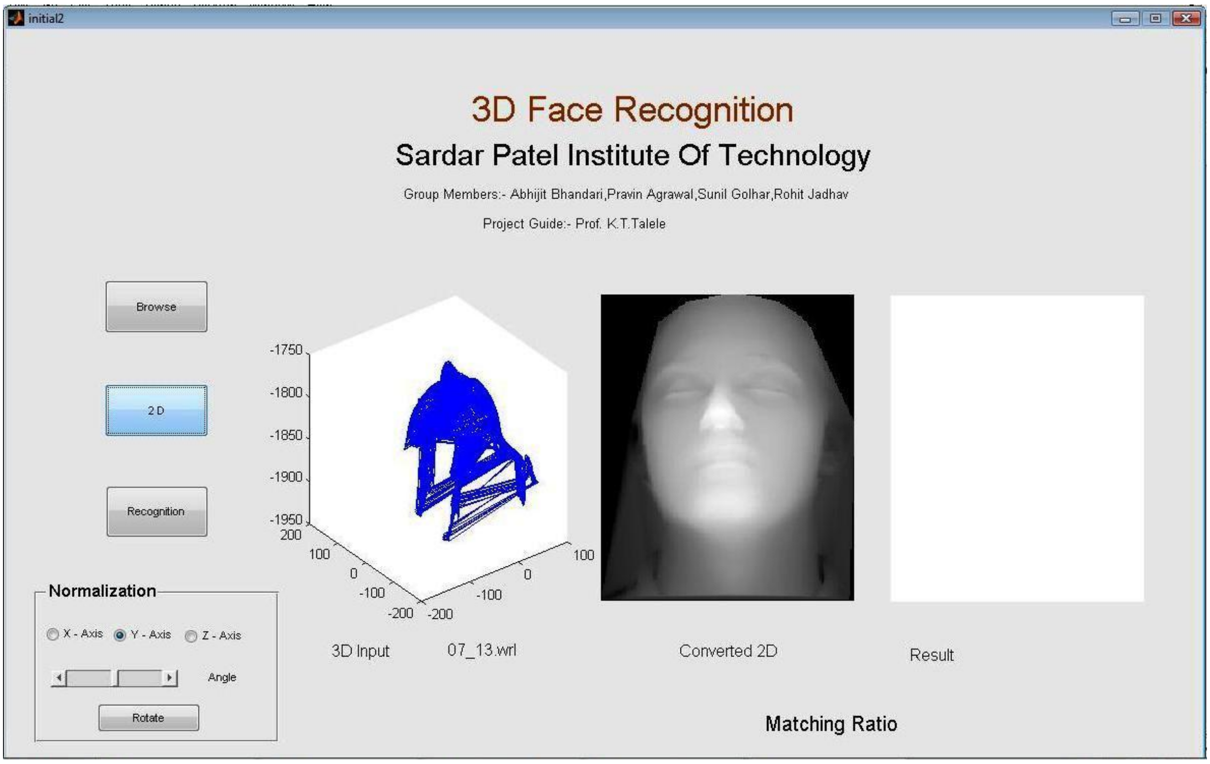


Figure 8.5: Rotated along X-axis

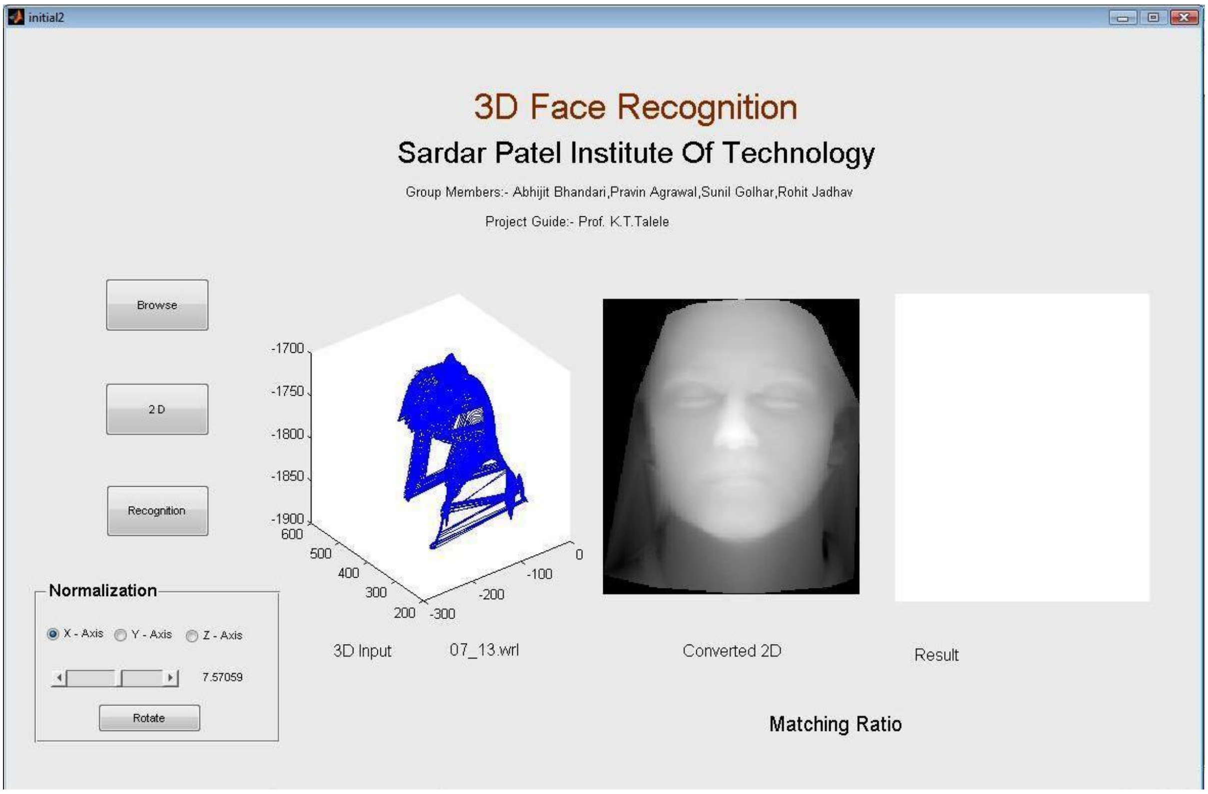


Figure 8.6: Normalised along X-Axis

b. Normalization along Y-Axis

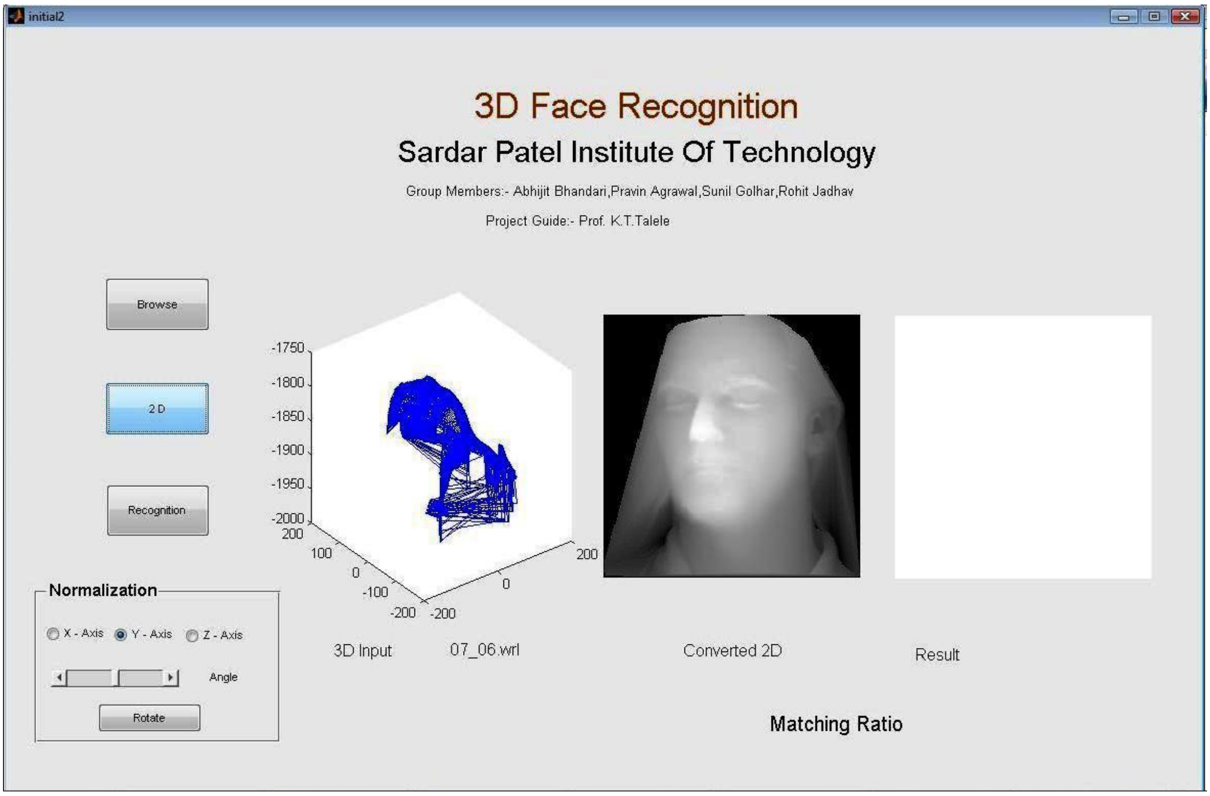


Figure 8.7: Rotated along Y-Axis

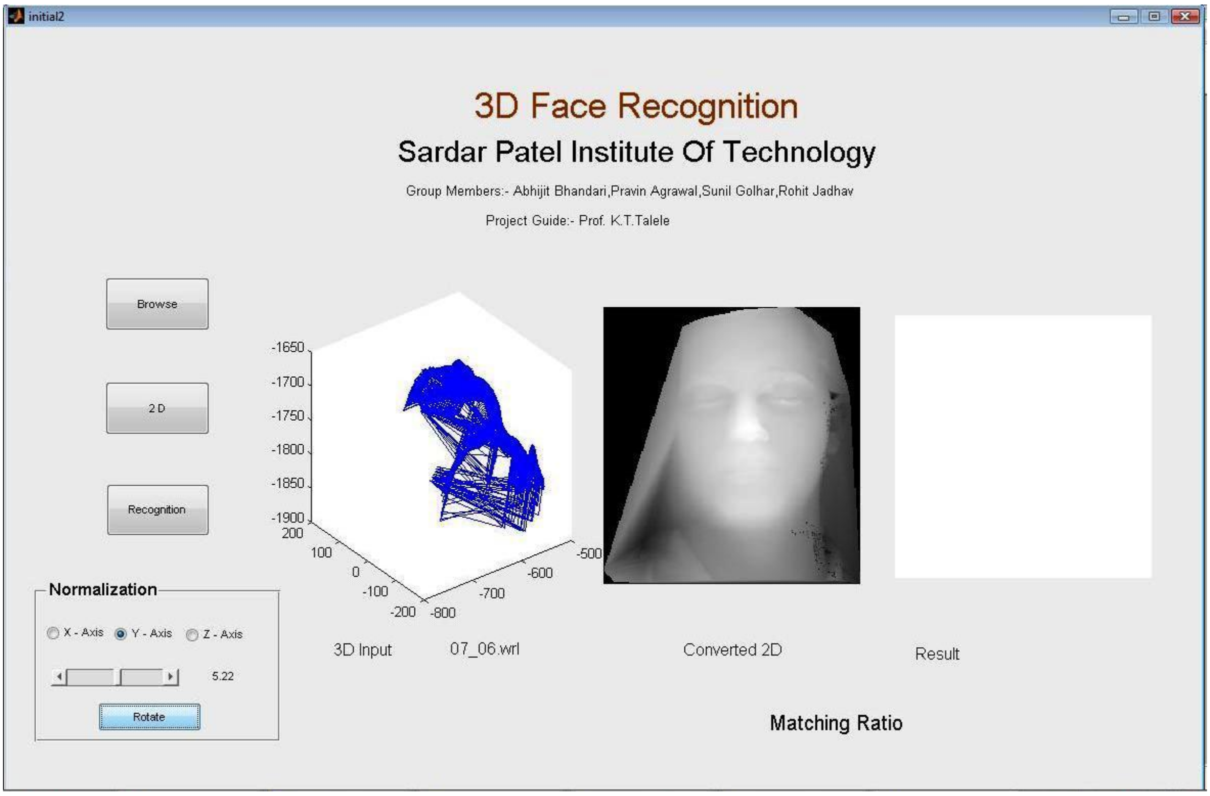


Figure 8.8: Normalized along Y-Axis

c. Normalization along Z-Axis

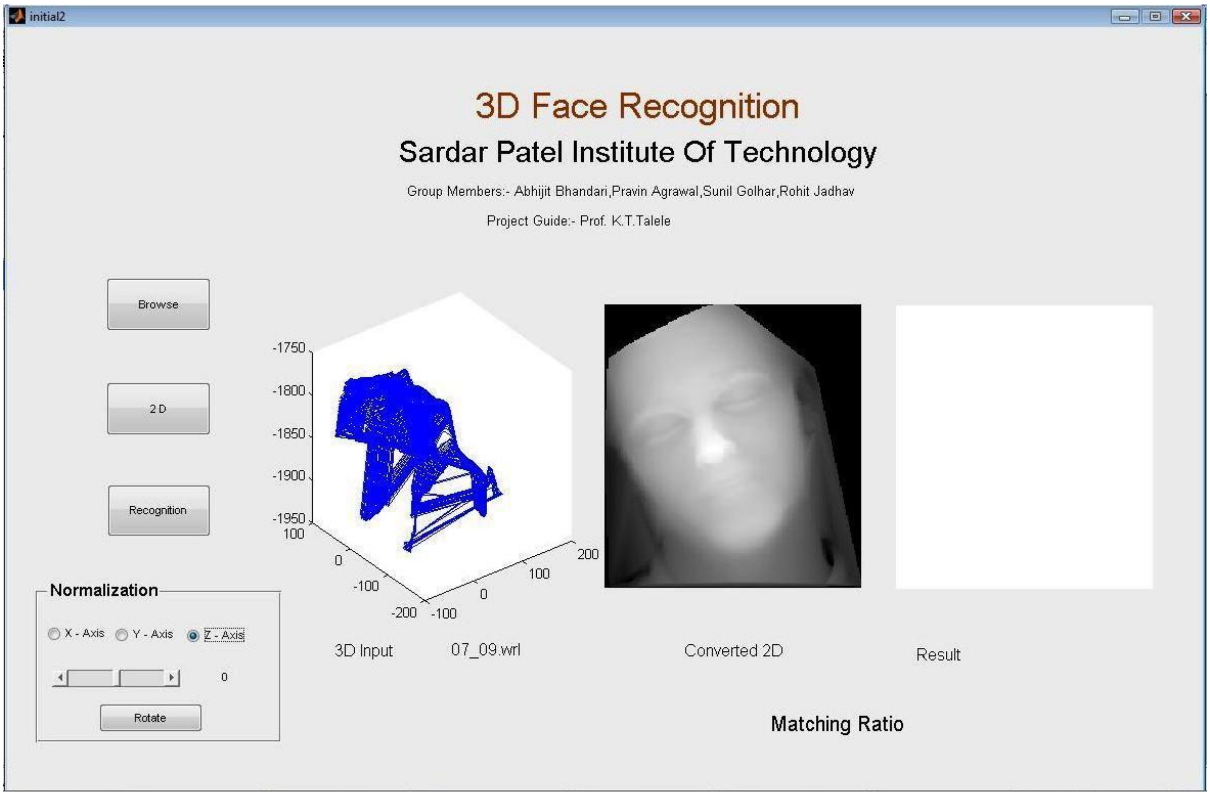


Figure 8.9: Rotated along Z-Axis

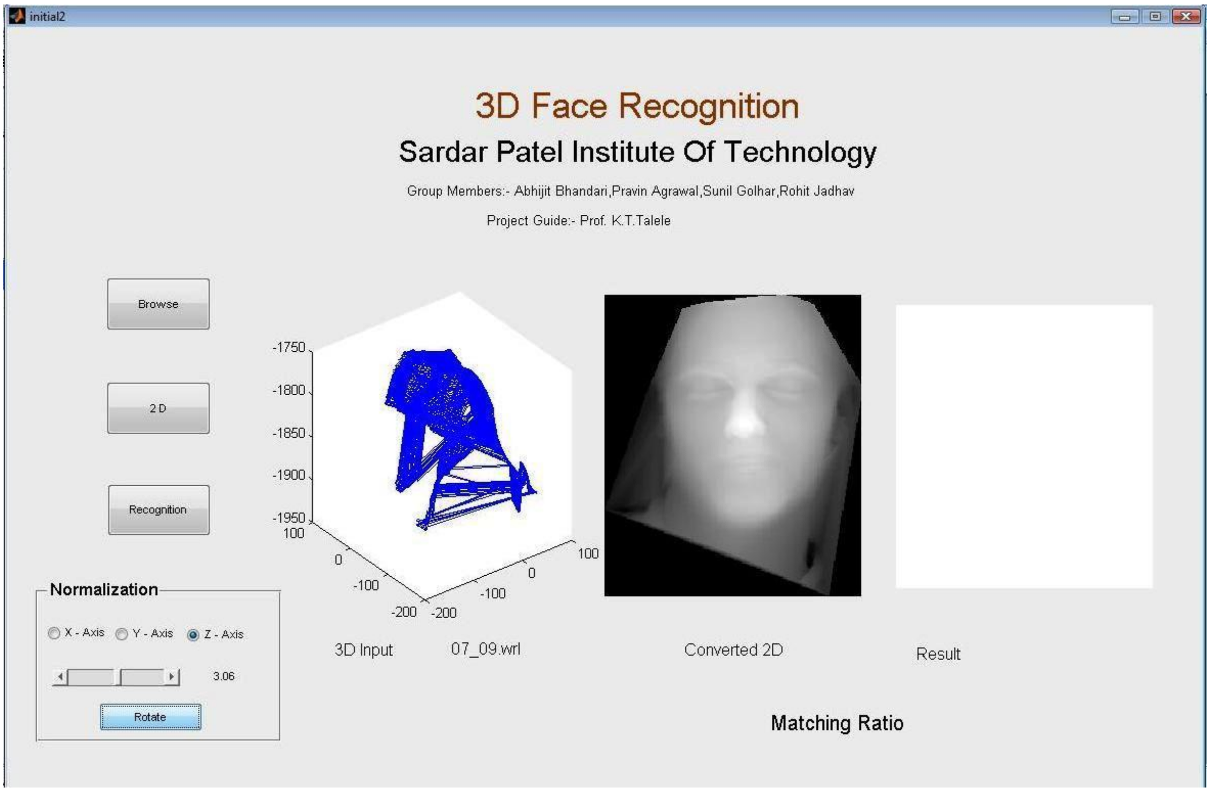


Figure 8.10: Normalized along Z-Axis

8.1.5. To Perform Face Recognition

- After the image is normalised the Face Recognition function is implemented with the help of the pushbutton **Recognition.**

- If the matching ratio of the input image is above the threshold, the result is considered a match.
- The highest matching ratio is displayed as shown along with the matched database image.

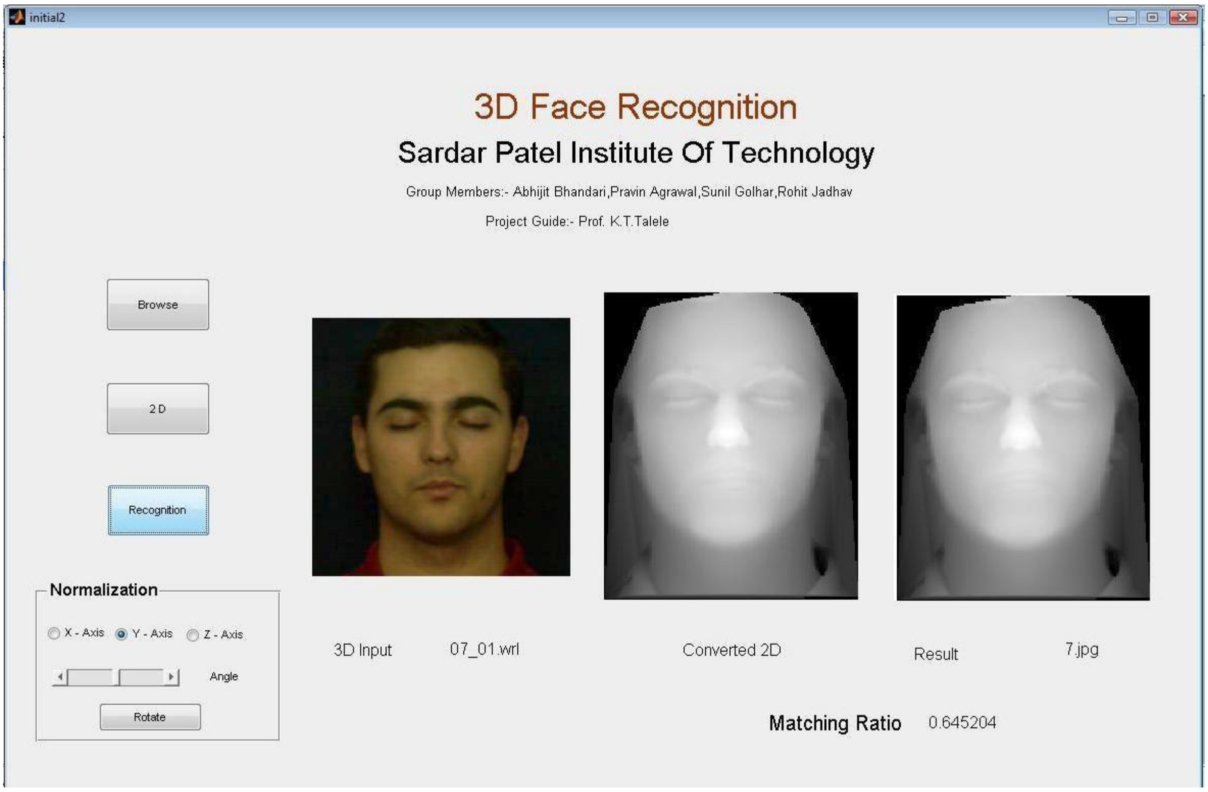


Figure 8.11: Final Output Of GUI

CHAPTER 9

APPLICATIONS

- There are several practical reasons for favouring facial recognition over other biometrics (iris scanning, fingerprinting, signature authentication etc) for the purposes of identification.

- Since the biometric data can be captured at a distance, it does not require active participation on the part of the subject - the individual need not pose, push a button or click a mouse to activate a system, stare into a lens or press an ink pad.

- Facial Recognition is unobtrusive and discrete. The infrastructure for its implementation is already widespread and inexpensive.

- At present the demand for Facial Recognition technologies are fueled by the needs of Homeland Security.

- Applications for Facial Recognition are varied and vast:

  - ✓ Document Control e.g. **digital chip in passports, drivers' licenses**
  - ✓ Computer security e.g. **user access verification**
  - ✓ Physical access control e.g. **smart doors**
  - ✓ Voter registration e.g. **election accuracy**
  - ✓ Time and Attendance e.g. **entry and exit verification**
  - ✓ Computer games e.g. **a virtual "you" plays against virtual opponents**
  - ✓ Border Controls/Airports

- With improvements in reliability, future applications will come increasingly from the commercial sector.

CHAPTER 10

CONCLUSION

The aim of the proposed project was to develop an efficient 3D face recognition system that would provide a high probability of appropriate face recognition and a minimum probability for false acceptance and false rejection. The 3D face recognition system developed has been successful in achieving its set objectives. The efficiency of the project is subject to the size of database.

By the realization of the proposed system we have learnt many aspects of human face recognition system.

The project can be revised further in terms of additional functionalities and features that can be appended to the developed system in the future.

CHAPTER 11

LIMITATIONS

Although the idea of the whole process sounds convincing, there are certain limitations in implementing the process which are stated as follows:

- Images which are tilted wit h a large rotation angle show false detection.
- Images with high illumination and with expression are not detected.
- The system works only for images with .wrl extension of 3D models.
- The system works only for VRML v1.0 models.
- Realtime implementation of this system is very costly.

Another notable point is that as the size of the database increases, the required processing power along with memory requirement also increases.

CHAPTER 12

FUTURE SCOPE

Image-based face recognition is still a very challenging topic after decades of exploration.

Real time face recognition such as capturing the 3D model with 3D camera and recognizing it at the same time is a challenging task. In order to perform this task we need special 3D Cameras which can capture image from two angles in order to create 3D face model. Since the device is very expensive it is not feasible to use this for general purpose.
Normalization is the area where we have huge scope to do it automatically.
Face recognition in video is yet another challenging area of exploration and a feasible technique to implement and use.
Also face recognition using coloured images and models with face mask is a topic to be explored.
So the scope for future development of our project is enormous. The field of face recognition incorporating coloured images is still new field with huge scope of exploration.

References

[1] Applying of Virtual Reality In MATLAB, Zhang Jiaxiang Luo Xueshan, Fang Lingjiang Mao Quansheng

[2] Cristina Conde, Angel Serrano and Enrique Cabello, Multimodal 2d, 2.5d & 3d Face Verification

[3] J.Feldmar, N.Ayache and F. Betting, 3D-2D projective registration of free-form curves and surfaces

[4] Xin Chen, Martin R. Varley, Lik-Kwan Shark, Glyn S. Shentall, Mike C. Kirby, Automatic 3D-2D image registration using partial digitally reconstructed radiographs along projected anatomic contours

[5] S. T. Gandhe, K. T. Talele and A. G. Keskar, Face Recognition Using Contour Matching

[6] Roberto Brunelli and Tomaso Poggio, Face Recognition: Features versus Templates

[7] Prof. Vijayakumar Bhagavatula, Face Recognition using Correlation Filters

[8] Cheng Zhong, Zhenan Sun, Tieniu Tan and Zhaofeng He, Robust 3D Face Recognition in Uncontrolled Environments

[9] Xiaoguang Lu and Anil K. Jain, Automatic Feature Extraction for Multiview 3D Face Recognition

[10] Rafael C. Gonzalez, Digital Image Processing

[11] http://mathworks.com