RFID AND FACE RECOGNITION BASED ENTRY SYSTEM

By

GC Badar Munir

CJUO Shahryar Ahmad

GC Afaq Rafiq

FC Muhammad Faraj

Submitted to the Faculty of Department of Electrical Engineering, Military College of Signals, National University of Sciences and Technology, Islamabad
in partial fulfillment for the requirements of a B.E Degree in Telecom Engineering

May 2015

Abstract

RFID tags which will be used to identify the authorized person. When the person passes its tag through the RFID Reader, the reader will read the tag. Video camera will be interfaced with the host computer. Reader will send the Tag of a particular student to the host computer for the identification of the student. As soon as the Tag is read by the Reader, camera will capture the image of the person. In the database each identification number corresponds to a particular person. Images of the persons are already stored in the database. Face recognition will be used for verification.

Automated face recognition can never surpass human face recognition system. There is, and will remain a space for improvement in terms of optimal recognition results or efficiency. One of the major problems of facial recognition is its computational power and complexity. This dilemma is more common for larger database even using state of the art computers. In this project Discrete Wavelet Transform will be applied prior to face recognition algorithm which will reduce the computational complexity by an element of four with negligible effects on results. The family of Daubechies are going to be investigated and good bases in terms of potency with in the family of Daubechies are going to be projected for Face Recognition.

It is hereby certified that the contents and form of the project report entitled "RFID and Face Recognition Based Entry System", submitted by the syndicate of

1. GC Badar Munir
2. CJUO Shahryar Ahmad
3. GC Afaq Rafiq
4. FC Muhammad Faraj

Have been found satisfactory as per the requirement of the B.E. Degree in Electrical (Telecom) Engineering.

Supervisor:

Col Dr. Imran Touqir

MCS, NUST

____

DECLARATION

We hereby declare that no content of work presented in this thesis has been submitted in support of another award of qualification or degree either in this institution or anywhere else.

DEDICATED TO

Almighty Allah,

Faculty for their help

And our parents for their support

ACKNOWLEDGEMENT

Nothing happens without the will of Allah Almighty. We thank Allah Almighty for giving us knowledge and strength to accomplish this task successfully.

We would like to thank our project supervisor,Col Dr. Imran Touqir, without his support and encouragement; it would not have been possible to complete this project.

We would also like to thank and our colleagues for helping in developing the project and people who have willingly helped us with their abilities.

Last but not least, we are very thankful to our parents, who bore with us in times of difficulty and hardship. Without their consistent support and encouragement we could not have accomplished our targets successfully.

Table of Contents

Contents

List of Abbreviations:

DWT	Discrete Wavelet Transform
FR	Face Recognition
GUI	Graphical User Interface
LSE	Least Square Error
MCS	Military College of Signals
PCA	Principal Component Analysis
PCB	Printed Circuit Board
RFID	Radio Frequency Identification
RGB	Red Green Blue

List of Figures

Chapter 1

1.1 Introduction:

1.1.1 Overview

RFID technology is now commonly used in our daily life. Barcodes are replaced by this technology. This technology provides more security and privacy as compared to the previous. But this technology is also being misused. To avoid this misuse, a module of Facial Recognition is added to the RFID Technology. A person can be identified by RFID chip and verified by face recognition.

Face Recognition is a computationally complex technique and will pose problem that needs to be addressed for its real time applications. A high resolution image enhances the computational complexity and envisage a large feature vector which needs to be processed. Another problem in this area is that high dimensionality inhibits the classification and also decision making process for face recognition.

A number of dimensionality reduction techniques exist for face recognition purpose such as Principal Component Analysis, yet even the best technique that exist in literature does not suffice the dimensionality reduction needs for real time scenario with limited resources. The optimal wavelet basis for face recognition has yet not been explored in the literature and is open for research.

Therefore there is a dire need to use an appropriate transform with near optimal basis coupled with some optimal dimensionality reduction technique prior to feature extraction for face recognition.

1.1.2 Problem Statement

"Preliminary Design and Prototype Development of RFID and Face Recognition Based Entry System for the user using the system."

1.2 Project Description

The Project involves the combination of a RFID Module with the Face Recognition of a person. The technique used for the facial Recognition is Principal Component

Analysis (PCA). Discrete Wavelet Transform is applied before the Principal Component analysis technique, reducing the computational complexity. A user has to scan his RFID card with the RFID reader, which will trigger the control to the person's folder in the stored database. The face recognition will be applied to that very folder. In this way the complexity is reduced further more.

The Facial Recognition portion of the project involves following steps.

i. Acquire input image/frame for face recognition in real time from camera.
ii. Detect and extract face from the image
iii. Normalize the face image.
iv. Extract approximated image by taking DWT.
v. Employ PCA and extract feature vector from the image.
vi. Compare and find least square error (LSE) of the feature vector with already stored database of the feature vectors.
vii. Decide match, no match or even not a face based on minimum of LSE.

If the face of a person is matched, the Microcontroller module gives the access to the person by opening the gate.

1.3 Prospective Application Area

The Project will have its applications in many areas.

i. Using RFID and FR enhances the security anywhere it is applied.
ii. The Security Situation of our country emphasis to enhance the security level.
iii. This project will reduce the manpower needed at any entry point of an organization.

1.4 Scope, Objectives, Deliverables and Specifications

1.4.1 Scope

The basic scope of the project is to achieve these objectives.

1. Face Recognition Software using MATLAB.
2. RFID reader and Microcontroller based Hardware module.

3. An algorithm matching the outputs of both.

1.4.2 Objectives

The objectives of the project include learning of various dimensionality reduction techniques including Multi-resolution Analysis, Eigen spaces, Principal Components Analysis (PCA), discrete transformations (DWT) and feature extraction techniques for facial recognition.

Goals of the project are.

1. A complete Face Recognition system using PCA and DWT.
2. Interfacing of Microcontroller with MATLAB in the host Computer.
3. Communication of Microcontroller and RFID Reader.
4. Matching Algorithm.
5. GUI on MATLAB of the Face Recognition System.

1.4.3 Deliverables:

1. Face Recognition based on PCA and DWT algorithms.
2. RFID Module
3. Serial port Communication
4. Communication via USB cable for the ease
5. LCD interfacing
6. PIC microcontroller
7. PIC 18F26K22 for its UART compatibility

1.4.2 Specifications

1.4.2.1 Hardware specification

Following hardware specifications are required in the project:

1. RDM6300 RFID Reader and RFID Tags.
2. Input: A Logitech Sphere AF web camera through a USB.
3. Recommended hardware: a PC with at least 1.6GHz CPU and 2GB memory.
4. Security Module comprising of PIC 18F26K22 Microcontroller and other components.

1.4.2.2 Software specification:

Following software specifications are required in the project.

1. MATLAB 7.11 2012a is used for the designing of Face Recognition System.
2. MikroC is used for Microcontroller Coding.

Chapter 2

2.1 Literature Review:

2.1.1 Overview of existing literature:

2.1.1.1 Principal Component Analysis

PCA has been used in the past as a major technique for Face Recognition system. Principal component analysis (PCA) may be defined as a mathematical technique that uses orthogonal transformations to convert set of observations of possibly correlated variables into a set of uncorrelated variables called PC (principal components). The number of principal components is less than or equal to the number of original variables. This transformation is done in such a way that the first principal component's variance is as high as possible, and each succeeding component in turn has the highest variance possible under the constraint that it has to be orthogonal to the preceding components. Principal components are independent only if the data set is normally distributed jointly. PCA show sensitiveness to the relative scaling of an original variables. It is also named the discrete Karhunen–Loève transform (KLT), depending on the field of application, proper orthogonal decomposition (POD) or the Hotelling transform [4].

Principal Components Analysis (PCA) is a statistical technique in modern data analysis that has found application in different areas such as face recognition. It is used for data compression and data analysis. The hidden, simplified structures that often underlie a complex dataset can be viewed by applying Principal Component Analysis (PCA). It is a method to identify patterns in data, and express the data in a way to highlight how similar and dissimilar they are. PCA is appropriate when you have obtained measures on a large number of observed variables and wish to develop a few smaller number of artificial variables, called principal components, that accounts for most of the variance in the observed variables. PCA is a variable reduction technique. It is useful when there is some kind of redundancy in variables. This means that some of the variables are correlated with one another. We could possibly reduce the observed variables into a smaller number of artificial variables (principal components) that will account for most of the variance in the observed variables. PCA is a linear transformation that transforms a data to a new coordinate system such that the principal components, the new set of variables, are uncorrelated, are linear functions of the original variables, and the greatest variance by any projection of the data comes to lie on the first coordinate, the second greatest variance on the second coordinate, and so on. Principal Component Analysis compute the covariance matrix for the whole data set. The eigenvalues and eigenvectors of the covariance matrix are computed, and are sorted according to decreasing value of eigenvalue [2].

2.1.1.2 RFID

Over the last few decades, man has tried a large number of methods for his security. Cards having barcodes were used in the past. Over the time these cards were replaced with RFID technology (Radio Frequency Identification). RFID chip is installed on the card. It can only be detected if it is in the range of the RFID detector. It is an automatic identification technology whereby digital data encoded in a RFID tag is captured by a reader using radio waves. RFID used in this system is contactless read/write and passive (Without any power source). It can be detected from 10cm up to 30cm [1].

2.1.1.3 Discrete Wavelet Transform

The discrete wavelet transform (DWT) is a linear transformation technique that operates on a vector of dataset whose length is an integer power of two, which transforms it into vector of the same length which are numerically different. DWT is a tool which separates data into various different frequency components, and then analyze each component with resolution matching to its scale. It is calculated with a cascade of filters followed by sub sampling of factor of 2. Outputs of this filter are given by following equations.

$$a_{j+1}[p] = \sum_{n=-\infty}^{+\infty} l[n-2p]a_j[n]$$

$$d_{j+1}[p] = \sum_{n=-\infty}^{+\infty} h[n-2p]a_j[n]$$

The Elements aj can be used for next step (scale) of the transform and elements dj, called wavelet coefficients, which determine output of the transform. L[n] and h[n] are coefficients of low pass and high-pass filters respectively. On scale j+1 there is only half from number of a and d elements on scale j, which causes that DWT can be done until only two aj elements remain in the analyzing signal these elements are called scaling function coefficients. DWT algo for two-dimensional pictures is also similar. The DWT is performed firstly on all image rows and then on every column. The main feature of DWT is its multi scale representation of a function. By using the wavelets, a function can be checked at various levels of resolution. The DWT is also invertible and can be orthogonal. Wavelets seem to be effective for analysis of textures recorded with different resolution. It is a very important task in imaging, because high-resolution images require long time for its acquisition. This causes an increase of artifacts caused by a patient movements, which should be avoided. There is an expectation that the proposed approach will provide a tool for fast, low resolution. It is a vector, which contains energies of wavelet coefficients calculated in sub bands at successive scales. To compute the wavelet features in the first step Harr wavelet is calculated for whole image. As a result of this transform there are 4 sub band images at each scale.

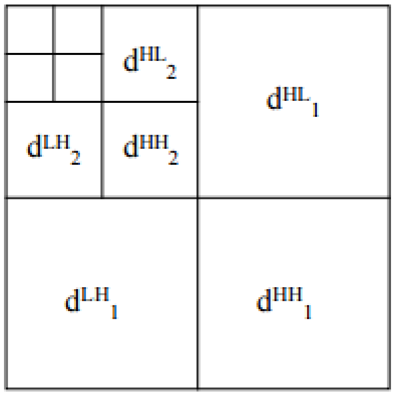


Figure 1: Image Transformation in DWT

Of course, ROIs are reduced in successive scales in order to correspond to sub band image dimensions. In a given scale the energy is calculated only if ROI at this scale contains at least 4 points. Output of this procedure is a vector of features containing energies of wavelet coefficients calculated in sub bands at successive scales[9].

Chapter 3

3.1 Detailed Design:

The project consists of two portions. First, the Face Recognition Algorithm which is implemented on MATLAB. The Second is the Microcontroller module which consists of RDM 6300 RFID Reader and PIC18F26K22 microcontroller. The block diagram of the project is given in figure 2.

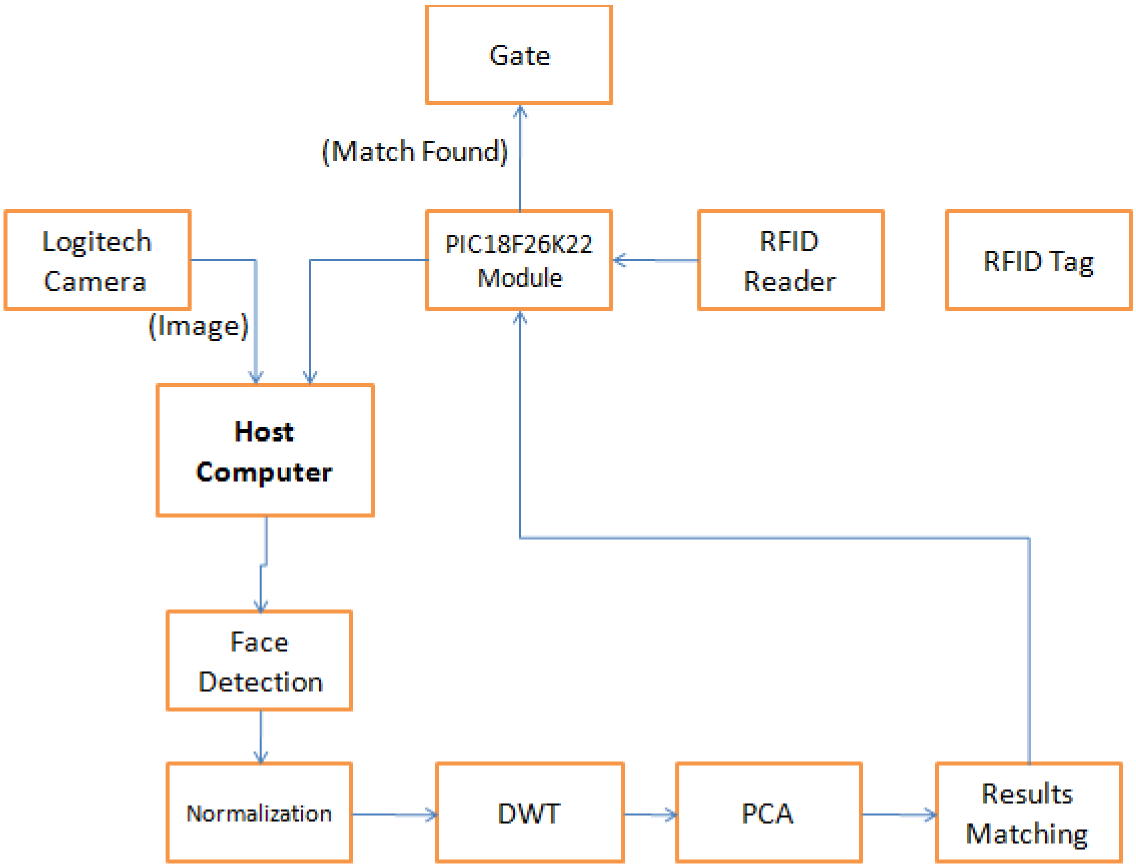


Figure 2: Block Diagram

3.1.1 Modules Description and Design

This project emphasize on how to build a complete Facial Recognition system using PCA and DWT. Useful wavelet basis amongst all the daubechies family will be proposed , based on the efficiency of wavelet basis. The result is to be matched with the RFID input.

Following algorithm will be explored to achieve our objectives:

1. RFID input from the Microcontroller module and giving it to the host PC.
2. Acquiring input image for face recognition in real time from the webcam.

3. Detecting and extracting face from the image (viola jones)
4. Normalization of the face image.
5. Extracting the approximated images by taking DWT.
6. Employing PCA and extract feature vector of the image.
7. Comparing and findong least square error (LSE) of the feature vector with already stored database of the feature vectors.
8. Decide match, no match or even not a face based on minimum of Least Square Error.

Further explanation of each module is shown in the subsequent paragraphs.

3.1.1.1 RFID/Microcontroller Module

This is the hardware part of the project. The module does the job of allowing authenticated personnel to either enter or exit a facility with the help of an automated mechanical barrier. It consists of a PIC18F26K22 microcontroller, Max 232 IC, RDM6300 RFID Reader and ULN2803 IC. As the command from the MATLAB GUI is given, the host computer listens the data transmitted over the serial port. When the person passes his tag through the RFID Reader; the reader will read the tag. Reader will send the Tag of a particular student to the Microcontroller which sends the desired information to the host computer via max 232 IC.

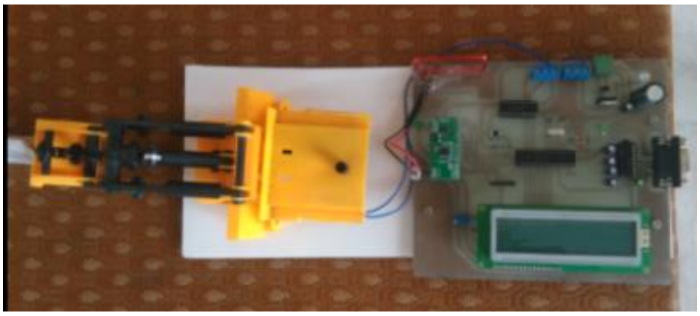


Figure 3: Hardware Module

3.1.1.1.1 PIC 18F26K22 Microcontroller

One of the major components of the project, the brain of our project, is the PIC18F26K22 microcontroller. It performs all the functions of the recognition of an individual, from communicating with the host computer to driving the gate using the DC motor.

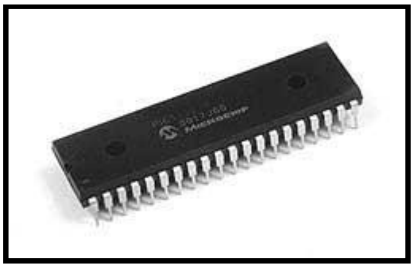


Figure 4: PIC 18F26K22

The microcontroller has been chosen attributable to USB communication capability and also the easy programming of the controller. Additionally the versatile library for the PIC18F simplifies the committal to writing and will increase the potency of the controller. This controller is copiously obtainable at a comparatively low value and it's a range of functions that makes it terribly appropriate for our project

3.1.1.1.2 MAX 232 IC

This IC is used when there is a communication from the Microcontroller to the host computer. This must be used before the serial communication port. The pin configuration of the MAX 232 is shown in the figure.

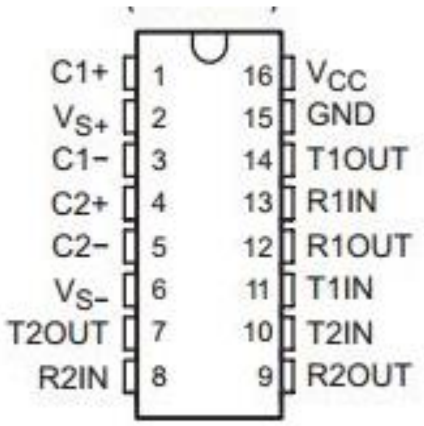


Figure 5: Pin Configuration of MAX 232 IC

3.1.1.1.3 USB Interface

The project hardware module communicates with the host system through the USB interface. We have programmed the PIC18F26K22 microcontroller to allow USB interfacing and with least hardware components we have successfully made a stable

USB interface that allows us to control the Hardware module from a computer system. We have used the USB-232 converter for our ease.

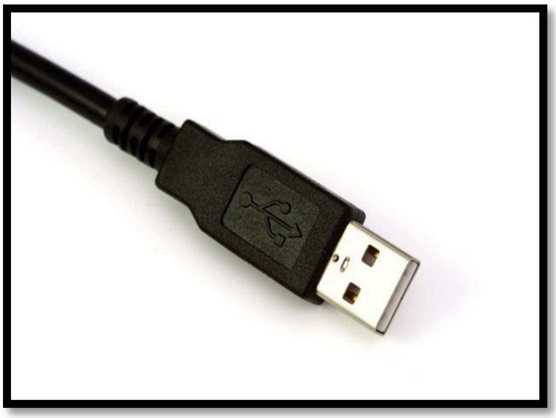


Figure 6: USB Cable

Figure 7: USB-232 Serial Converter

3.1.1.1.4 DC Motor

In order to move the gate, we have employed an ordinary DC motor that is capable of providing sufficient torque.

Figure 8: DC Motor

3.1.1.1.5 RDM6300 RFID Reader

RDM6300 RFID card reader is designed for reading code from 125 kHz card compatible read-only RFID tags. It has following attributes.

- Support external antenna
- Maximum effective distance up to 50mm
- UART Interface
- Smaller design

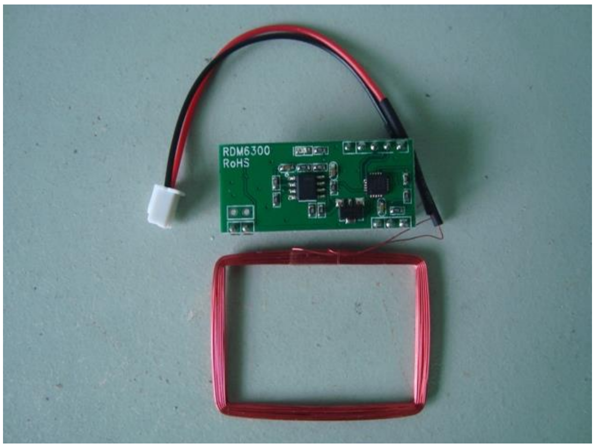


Figure 9: RDM6300 RFID Reader

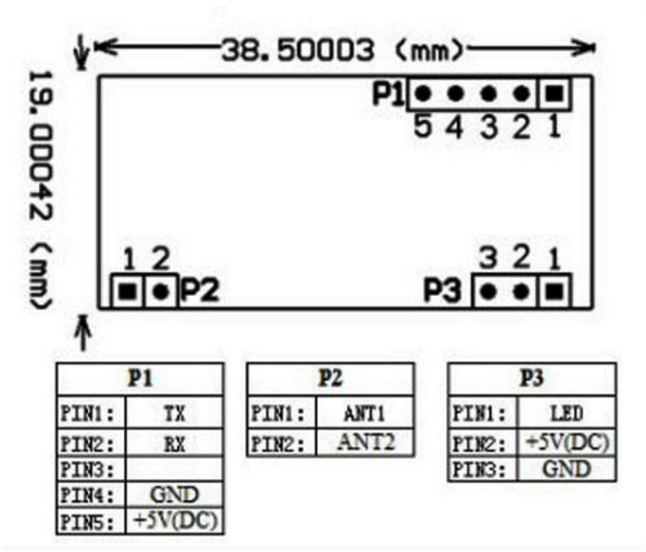


Figure 10: RDM 6300 Pin Configuration

It has a smaller design and is not that expensive so it is economical to deploy in this project.

3.1.1.2 The Image Acquisition Module

This is the first point of the facial recognition system. The individual is asked to present an image of his face to the system. A webcam is used to get this image of the individual.

3.1.1.2.1 Camera selection

Face recognition requires a good camera which should be having Good Pan, zoom, resolution and tilt capacity.

3.1.1.2.2 IP Camera

Image taken from an IP Camera is given to the viola jones face detection algorithm distorts badly and the final image is of a poor quality. Thus inhibiting the system to recognize a person.

Figure 11: IP Camera

3.1.1.2.3 Webcam

The Logitech sphere AF webcam has zoom, pan and tilt facility. This camera gives 102 degrees of tilt and 189-degrees of field of view. It is very easy to interface to the computer. In poor lighting conditions, it is able to adjust intelligently, producing the best possible image for the FR Process. Moreover, it is very economical to deploy in the project as it has been provided by the MCS.

Figure 12: Logitech Sphere AF Webcam

3.1.1.3 Face Detection

Face detection segments the face region from the background. The detected face may have to be half-tracked employing a face tracking element. Face alignment is aimed toward achieving additional correct localization and at normalizing faces thereby whereas face detection provides course estimate of the placement and scale of every detected face. Facial parts like face, nose, eyes and mouth and facial outlines square measure settled supported the placement purpose. The input face image is normalized with relation to geometric properties like size and poses victimization geometric transforms or morphing. The face is typically additional normalized with relation to measuring properties like illumination and grey scale.

One of the classical techniques employed in face detection is Viola Jones algorithm. This method depends on the utilization of straightforward Haar-like options that are evaluated quickly through the utilization of a replacement image illustration. The calculation of the options is expedited with the utilization of an “integral image” that races the system.

3.1.1.4 Image Normalization

Appearance based strategies operate sub windows of fastened size. So resizing is critical. Standardization of pixel intensity helps correct variations in imaging parameters in cameras moreover as changes in illumination conditions. Intensity standardization operation includes mean values, standardization, bar chart leveling and illumination correction. The face image is resized to uniform dimensions of 2p x 2q wherever p and q are fastened integers.

3.1.1.5 Discrete Wavelet Transform

Wavelet transforms are not taking into account altered premise, rather they build up outline parameters. Wavelets are the little minimalistically upheld waves with unit vitality and zero mean. Under suitability obliges their expansion and constriction alongside interpretations constitute the Hilbert space. Mallat created wavelet execution through channel banks that imagines impeccable recreation. Daubechies called the channel coefficients as enchantment numbers that submit to the wavelet configuration compels.

DWT is connected to the picture in even heading took after by vertical bearing. Decimation by two if there should be an occurrence of Haar is connected in every heading to save over all vitality. The picture is decayed into its estimates termed as LL band, level points of interest termed as LH band, vertical subtle elements termed as HL band and corner to corner points of interest termed as HH band where the letter L speaks to low recurrence and H speaks to high recurrence. Figure-13 shows Image disintegration through wavelet channel banks.

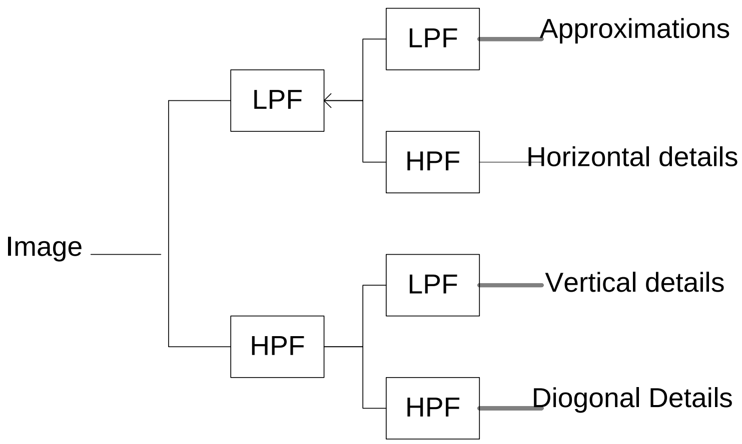


Figure 13: Wavelet decomposition filter bank

DWT, which is in view of sub-band coding, is a strategy found to yield a quick reckoning of Wavelet Transform. It is anything but difficult to actualize and decreases the processing time and assets needed. The computational many-sided quality of DWT is O(N). Wavelet transform can be performed just at products of scale and interpretation interims. A fitting wavelet transform can bring about strong representations with respect to force changes and is fit for catching generous facial components while keeping computational many-sided quality low. Processing rate of the framework is further enhanced through component enhancement by characterizing a limit esteem beneath which transformed qualities are made zero. By disintegrating a picture utilizing DWT, the determination of the sub group's pictures are lessened. Thus, the computational multifaceted nature will be decreased drastically by taking a shot at lower determination picture. The face picture close estimations with determination 16x16 is utilized by utilizing DWT as 16x16 determination is adequate for precise face acknowledgment.

3.1.1.6 Principal Component Analysis

PCA is a broadly utilized method for face recognition process. A discriminating issue in the examination of such multidimensional information is its high dimensionality. PCA is a dimensionality lessening method in view of removing the wanted number of main parts of the multidimensional information. The main chief segment is the straight blend of the first measurements that have the greatest fluctuation; the nth central part is the direct mix with the most noteworthy difference, subject to being orthogonal to the n-1 first primary segments. PCA is taking into account second request measurements of the information picture, which tries to accomplish an ideal representation that minimizes the recreation blunder in a slightest square sense. Despite the fact that PCA has impediments, for example, poor unfair force and vast computational burden, yet it a cross breed game plan is visualized to repay it by DWT dimensionality lessening module. Steps involved in PCA implementation are:

1. Get some data
2. Subtract the mean
3. Calculate the covariance matrix
4. Calculate the eigenvectors and eigenvalues of the covariance matrix
5. Choosing components and forming a feature vector
6. Deriving the new data set

3.1.1.7 Feature Extraction Module

The normalized face image is extended to the feature extraction module in to seek out the key options that are probably to be used for classification. In different words, this module is liable for composing a feature vector that's to an adequate degree to represent the face image.

3.1.1.8 Match Module

Recognition module is last module of the planned face recognition model wherever a check image is bestowed as input. Check image is 1st preprocessed and afterward it is decimated with same value of down multiplier factor as employed in training module. Then image is born-again into a column vector with a dimension of L x 1. In matching method, the geometrician distance of this vector with every column of subspace gathered throughout coaching of model is nonheritable and at last a distinction matrix of dimension L x T is obtained. LSE is calculated and when comparison with that threshold yields the result as recognized face, not recognized face or maybe not a face image.

Chapter 4

4.1 Project Analysis and Evaluation

All the result and simulation related to project are shown below.

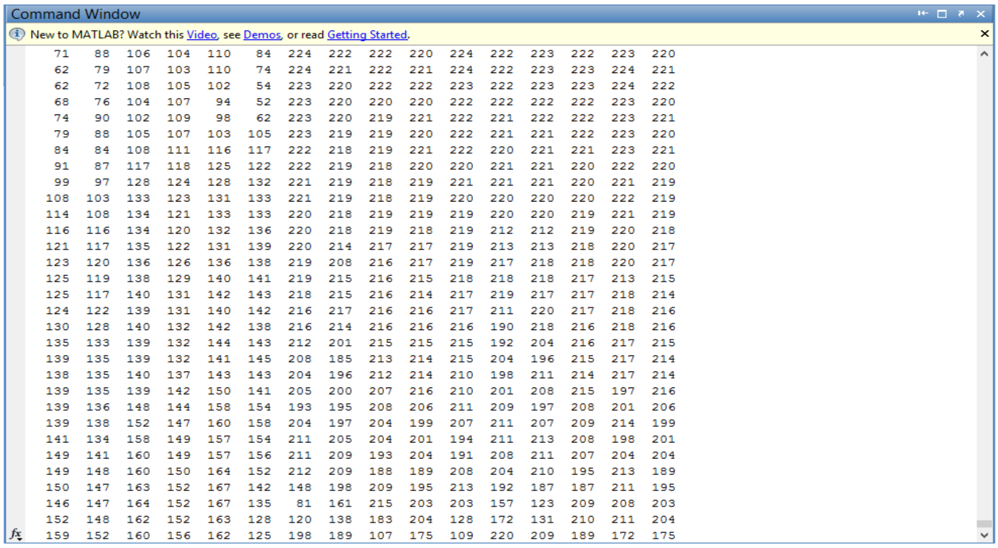
Figure 14: Main Program GUI

The main program GUI consists of the following pushbuttons.

1. Database
2. Webcam
3. New user

4.1.1 Database:

When the database pushbutton is pressed, it loads all the feature vectors of the images stored in the database.



Command Window

New to MATLAB? Watch this Video, see Demos, or read Getting Started.

71	88	106	104	110	84	224	222	222	220	224	222	223	222	223	220
62	79	107	103	110	74	224	221	222	221	224	222	223	223	224	221
62	72	108	105	102	54	223	220	222	222	223	222	223	223	224	222
68	76	104	107	94	52	223	220	220	220	222	222	222	222	223	220
74	90	102	109	98	62	223	220	219	221	222	221	222	222	223	221
79	88	105	107	103	105	223	219	219	220	222	221	221	222	223	220
84	84	108	111	116	117	222	218	219	221	222	220	221	221	223	221
91	87	117	118	125	122	222	219	218	220	220	221	221	220	222	220
99	97	128	124	128	132	221	219	218	219	221	221	221	220	221	219
108	103	133	123	131	133	221	219	218	219	220	220	220	220	222	219
114	108	134	121	133	133	220	218	219	219	219	220	220	219	221	219
116	116	134	120	132	136	220	218	219	218	219	212	212	219	220	218
121	117	135	122	131	139	220	214	217	217	219	213	213	218	220	217
123	120	136	126	136	138	219	208	216	217	219	217	218	218	220	217
125	119	138	129	140	141	219	215	216	215	218	218	218	217	213	215
125	117	140	131	142	143	218	215	216	214	217	219	217	217	218	214
124	122	139	131	140	142	216	217	216	216	217	211	220	217	218	216
130	128	140	132	142	138	216	214	216	216	216	190	218	216	218	216
135	133	139	132	144	143	212	201	215	215	215	192	204	216	217	215
139	135	139	132	141	145	208	185	213	214	215	204	196	215	217	214
138	135	140	137	143	143	204	196	212	214	210	198	211	214	217	214
139	135	139	142	150	141	205	200	207	216	210	201	208	215	197	216
139	136	148	144	158	154	193	195	208	206	211	209	197	208	201	206
139	138	152	147	160	158	204	197	204	199	207	211	207	209	214	199
141	134	158	149	157	154	211	205	204	201	194	211	213	208	198	201
149	141	160	149	157	156	211	209	193	204	191	208	211	207	204	204
149	148	160	150	164	152	212	209	188	189	208	204	210	195	213	189
150	147	163	152	167	142	148	198	209	195	213	192	187	187	211	195
146	147	164	152	167	135	81	161	215	203	203	157	123	209	208	203
152	148	162	152	163	128	120	138	183	204	128	172	131	210	211	204
159	152	160	156	162	125	198	189	107	175	109	220	209	189	172	175

Figure 15: Loading Database

4.1.2 Webcam:

When the Webcam button is pressed, it runs the main program. Firstly it is required that a user must scan his RFID card.

Figure 16: Scanning a RFID Card

After it is read by the microcontroller and finally it is read by the MATLAB. It triggers on the Webcam and user's Image is taken.

Figure 17: Webcam Image

After taking the Image, the PCA is implemented and the image taken is matched with user's images in the database.

If the image is matched with a stored image, the gate present at the hardware module is opened.

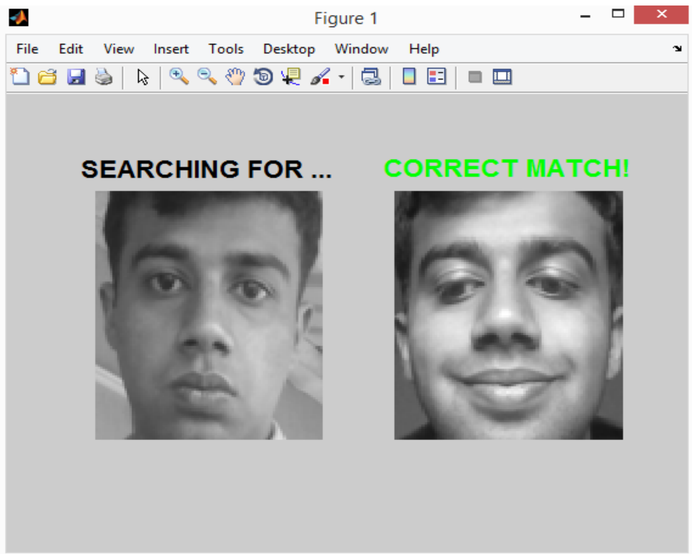


Figure 18: Image is matched

If the image is not found, the gate remains closed.

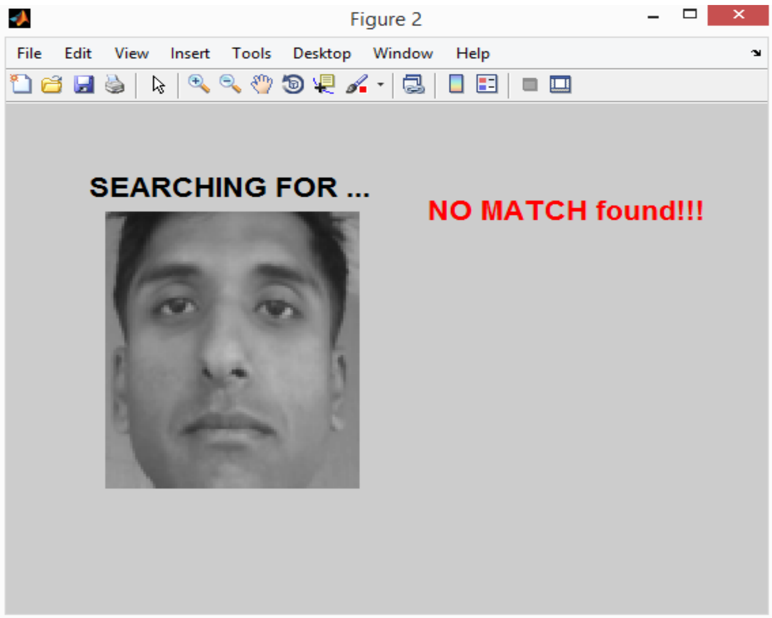


Figure 19: No Match found

4.1.3 New User:

When the new user button is pressed a new GUI window is popped up, prompting the password. When the password entered is correct the following window is displayed.

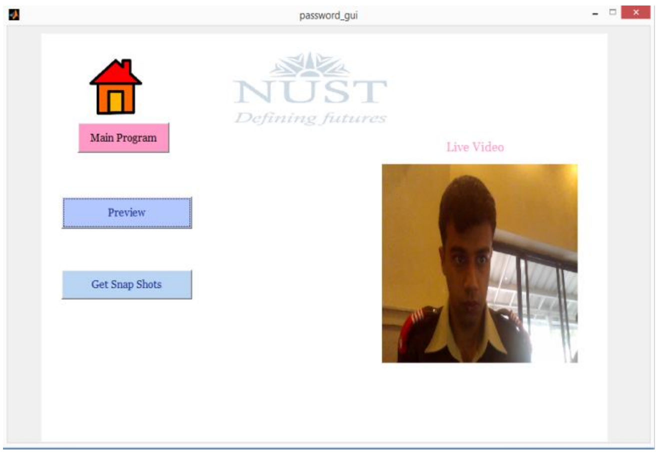


Figure 20: New User Window

When get snapshots button is pressed, the programs takes 10 images of a user and store it in the existing database.

4.2 Viola Jones openCV algorithm

Picture is procured from put away database, then it go through viola jones calculation for face location. Viola jones identifies the face in a picture utilizing Haar components and ada-boast system.

Figure 21: Face detection using Viola Jones Algorithm

Figure 22: Face extraction of selected image

4.3 DATABASE of Course TE-48

Database of students from course TE-48 is gathered in such a way, to the point that it ought to satisfy our venture prerequisites. For this we took 10 pictures of every individual with distinctive appearances, with or without glasses, then from every picture face is separated utilizing Viola Jones calculation then it is changed over into ".pgm" format. After that picture standardization is done over it to get the best conceivable match.

4.4 Discrete Wavelet Transform

DWT is connected over entire database to diminish processing for this db4 wavelet, level 1 is utilized. Level 1 implies that picture processing is diminished by 1/16. By expanding further levels computational multifaceted nature each time is further diminished, and framework effectiveness increments. Figure 23 shows results when db4 wavelet is applied on a picture

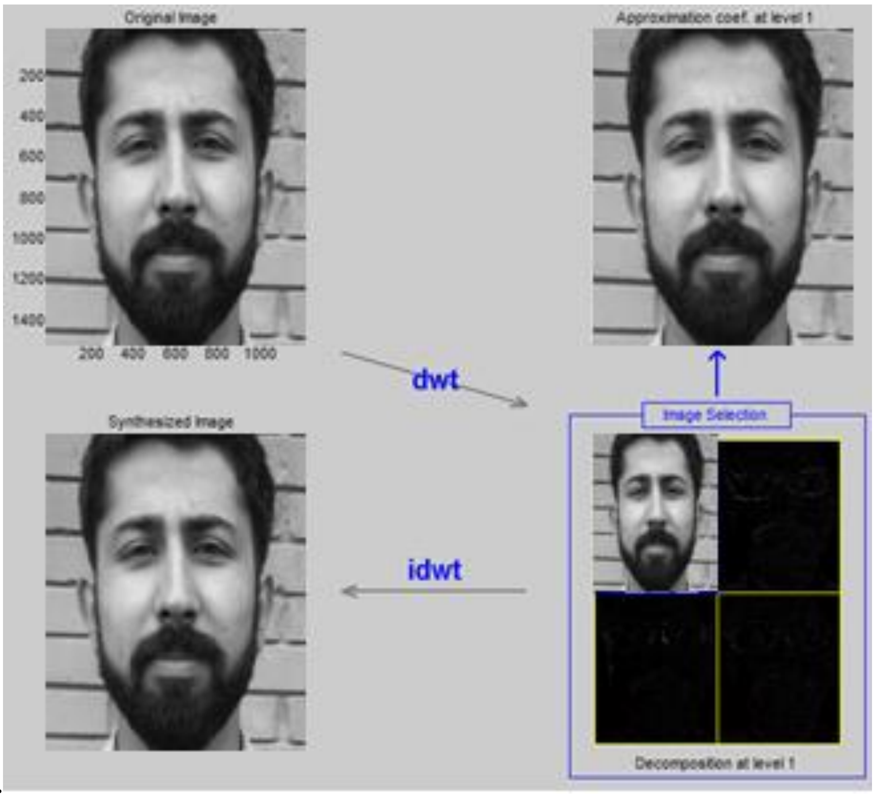


Figure 23: DWT using db4 Wavelet Level 1

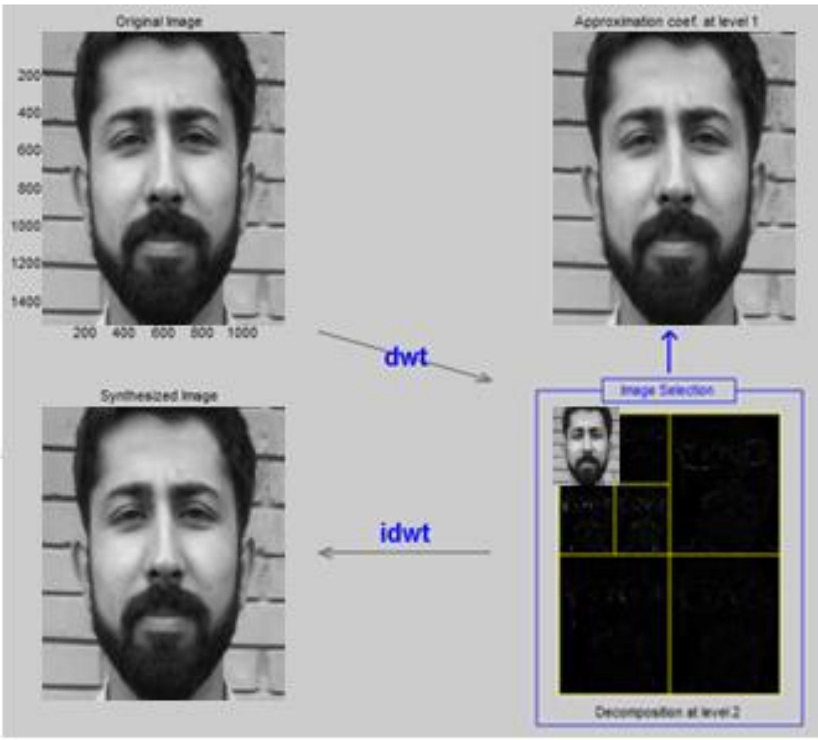


Figure 24: DWT using db4 Wavelet Level 2

4.5 Face Normalization

In the wake of applying DWT, next step is face normalization. This is the most critical stage in facial recognition. The capacity to accurately order the picture relies on upon different variables including lighting (enlightenment reliance), posture, outward appearances, and picture quality. The face ought to additionally be in an extremely steady position inside of the pictures, (for example, the eyes being in the same pixel coordinates), predictable size, revolution edge, hair and cosmetics, feeling (grinning, irate, and so on), position of lights (to one side or above, and so forth). This is the reason it is so critical to utilize a decent picture preprocessing channels before applying facial recognition. Before performing face recognition it is consequently officeholder to change over the caught picture into a standard arrangement such that it doesn't frustrate and lessen the exactness of face recognition yet just guides it.

Face is recognized with different expressions, when a person enters, his image is taken using camera, then using viola jones algorithm his face is detected and passed to the algorithm and is checked for the closest match, if **correct match is found** person is going to be verified by his RFID tag and his details are displayed otherwise **no match found** will be displayed .

Following results show how face normalization is achieved in the project.

4.5.1 Tolerance to facial expressions (emotions)

STUDENT NAME: PC ARSLAN MOVEED

Figure 25: Image of student PC Arslan present in stored database

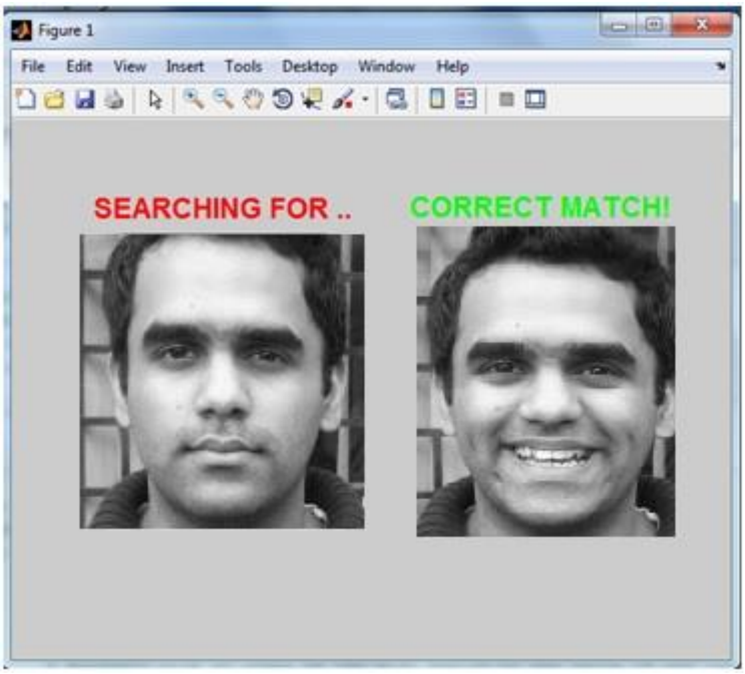


Figure 26: Correct match is achieved for PC Arslan

In above figure, it is demonstrated that the project has identified completely two diverse articulation of same individual via looking for the nearest match in entire database. For this picture that must be sought is initially erased from the database and contrasted and remaining information base.

4.5.2 Cunning Facial Expressions

Image of cunning facial expression on left right matched with smiling face of same person.

STUDENT NAME: BILAL RIAZ

Figure 27: Image of Student GC Bilal Riaz present in stored database

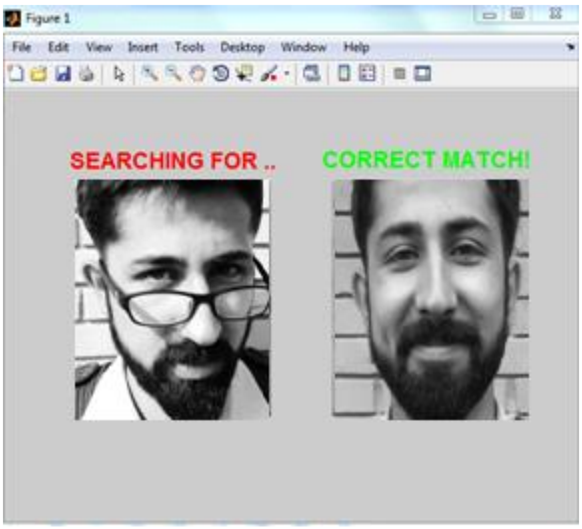


Figure 28: Correct match is achieved for GC Bilal Riaz

In above figure, it is demonstrated that the project has identified completely two diverse articulation of same individual via looking for the nearest match in entire database.

Image of cunning facial expression on left right matched with smiling face of same person.

4.5.3 FACE POSE:

STUDENT NAME: GC ABDUL HANNAN

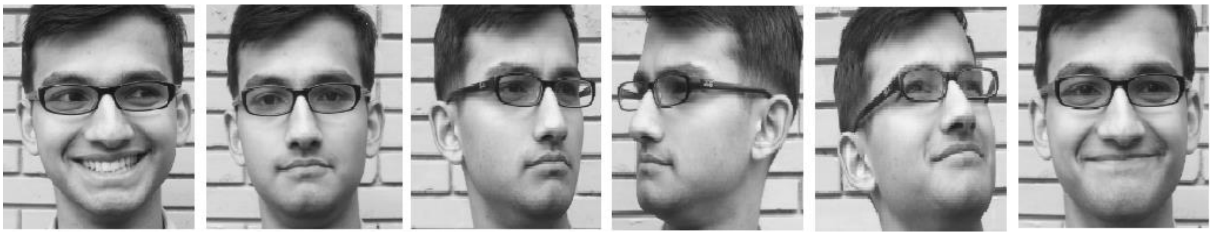


Figure 29: Images of student GC Abdul Hannan present in stored database

Figure 30: Correct match is found for GC Abdul Hannan

For recognition of this specific student his image that has to be searched is first deleted from the database and compared with remaining data base. As upright face matched with side pose of same person.

4.5.4 Tolerance to facial appearance (Facial hair)

STUDENT NAME: MEHMOOD AHMAD

Figure 31: Image of student GC Mehmood present in stored database

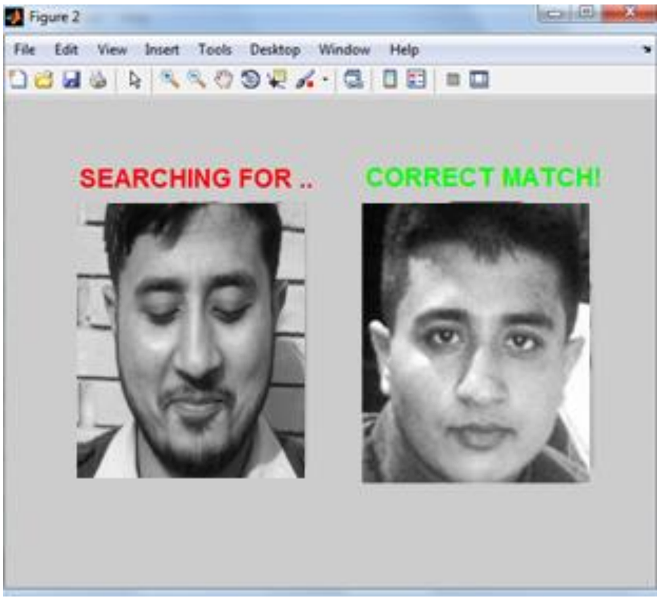


Figure 32: Correct match is found for GC Mehmood

In above figure, this chose picture that must be sought is initially erased from the database and contrasted and remaining information base. The individual on left has facial hair, and right is clean shaved. Both pictures are of same individual.

4.5.5Facial occlusion:

STUDENT: PC HAMZA MEHMOOD

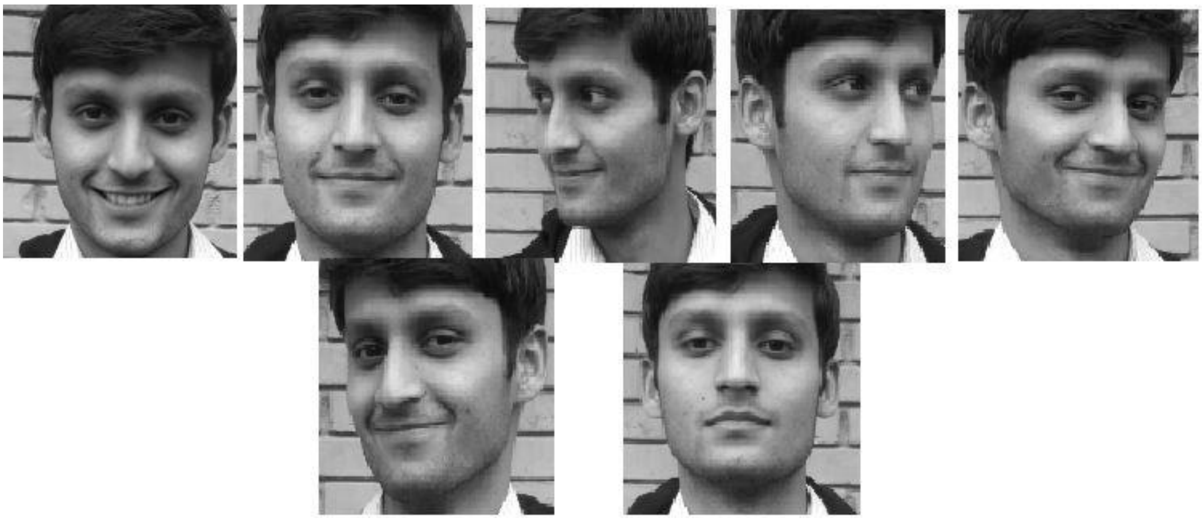


Figure 33: Images of Student PC Hamza Mehmood Present in stored database

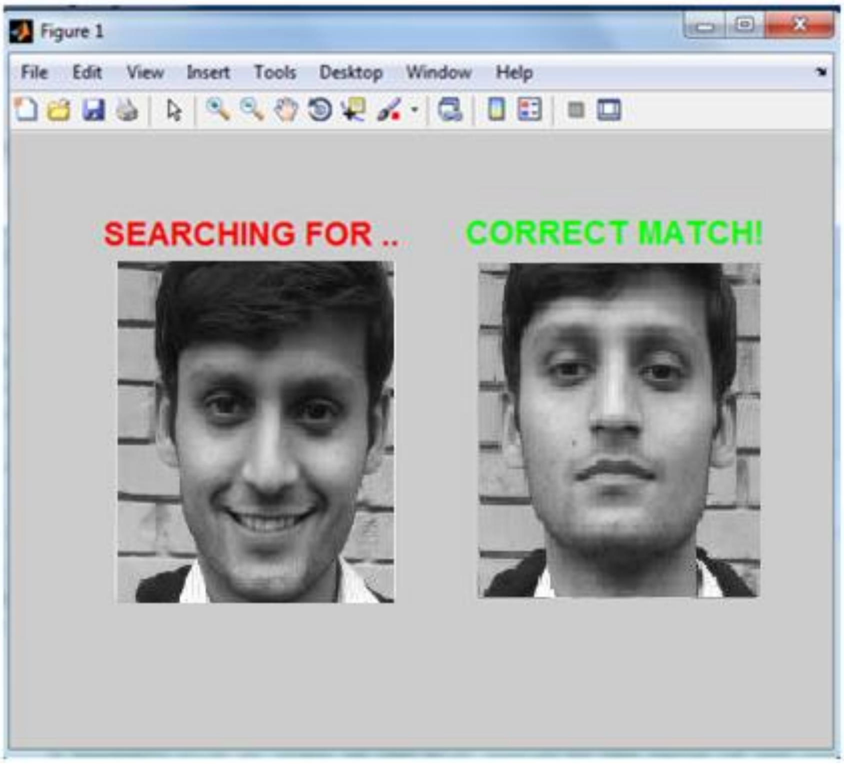


Figure 34: Correct match is found for PC Hamza Mehmood

In above figure, this chose picture that must be looked is initially erased from the database and contrasted and remaining information base .Frontal face of same individual is coordinated with his own particular eyes down picture.

4.5.6 Detecting funny facial expression

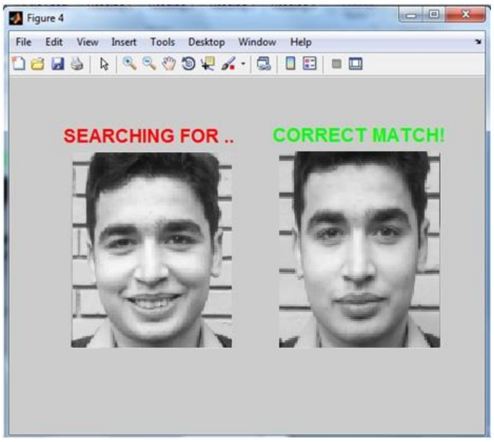


Figure 35: Correct match is found for GC Wasil

For this above student, his Funny face is matched with his shocking expression present in the stored database depending on the closest matched image

4.5.7 Facial Illumination Normalization

For brightening standardization, INFace toolbox has been utilized as a part of our task .The INFace (Illumination Normalization systems for strong Face recognition) toolbox in its present structure is an accumulation of capacities which perform enlightenment standardization and, thus, handle one of the best difficulties in face recognition. We have utilized two methods from this toolbox for cooking photometric and luminance issues.

4.5.8 HISTOGRAM EQUALIZATION:

Different histogram methods are utilized to get best conceivable results. For instance a picture that has high splendor is standardized by histogram leveling. Utilizing INFace toolbox, different histogram strategies are connected on a picture and base on its histogram plot that picture is handled further for discovery and recognition. Unique picture is standardized regarding its luminance and photometric properties.

4.5.9 ILLUMINATION NORMALIZATION USING DWT:

Force of dim picture is diminished as shine is expanded utilizing DWT with HAAR wavelet level1. Four unique disintegrations are acquired LL, LH, HL, and HH. Among them LL is chosen that gives u approximated subtle elements of a picture. Applying DWT builds splendor as well as lessen calculations various times as clarified prior.

4.6 Hardware Implementation

To build a prototype of any gate of an Organization, a hardware module is designed that grant access only to the authorized users who are registered in the database of the project.

4.6.1 Schematics and PCB Design of Hardware Module

The proteus simulation diagram and the PCB layout is given below.

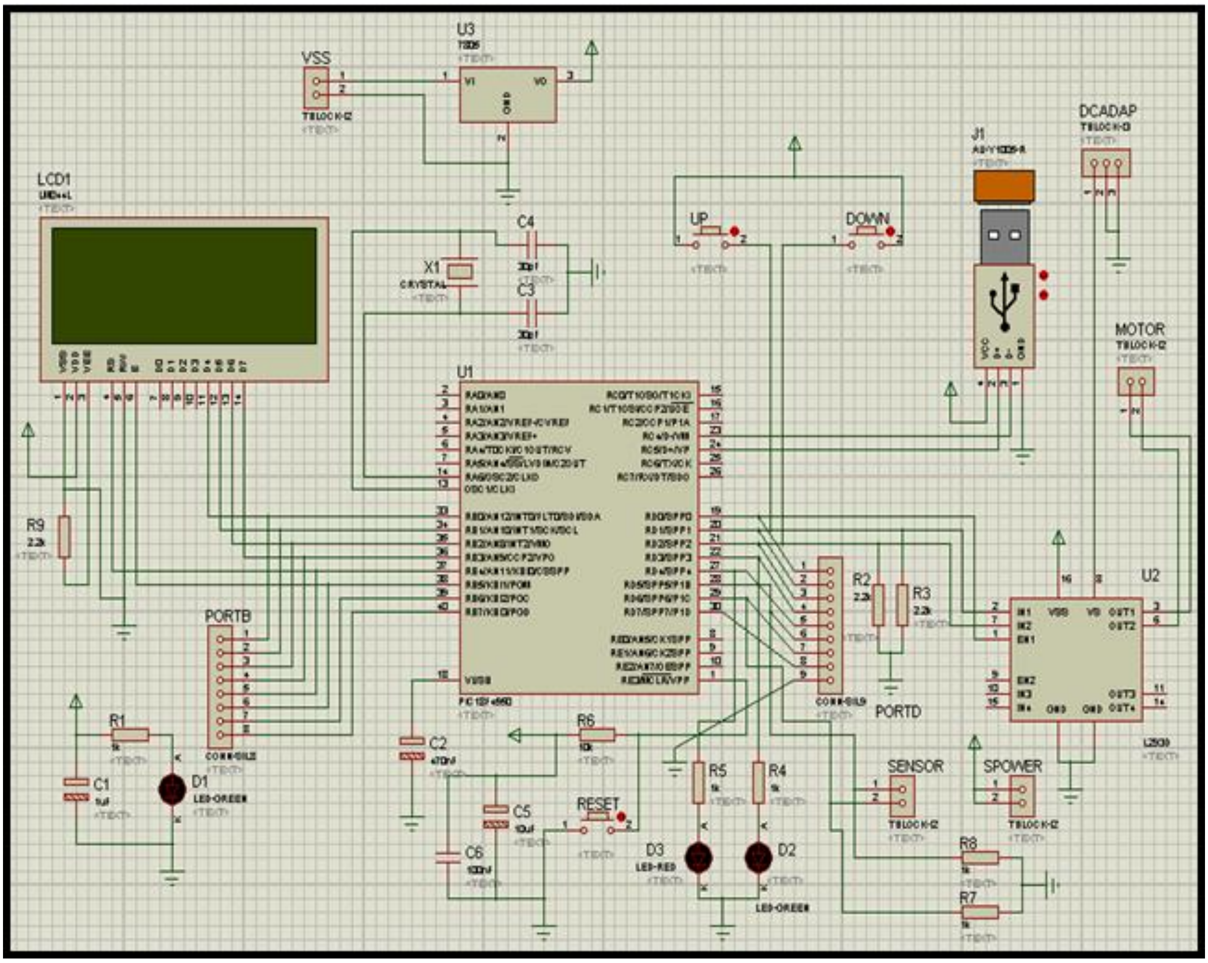


Figure 36: Proteus Simulation

4.6.2 PCB Trace and Component layout

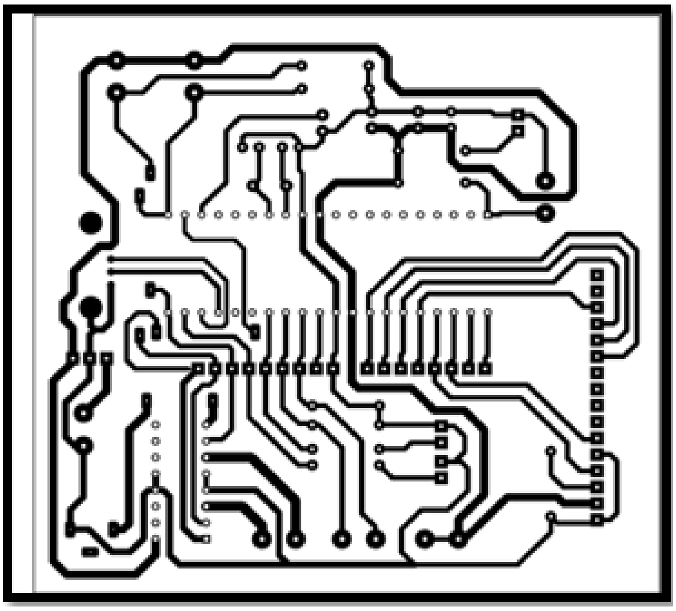


Figure 37: PCB Trace Layout

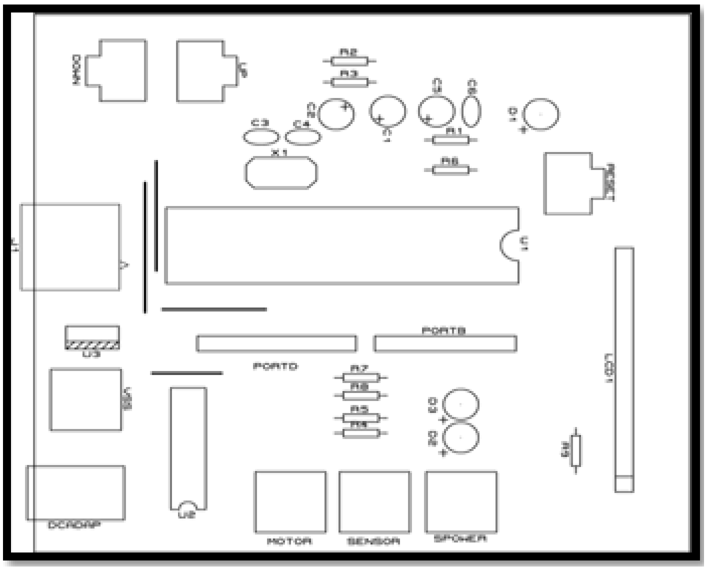


Figure 38: PCB Component Layout

4.7 Efficiency of Face Recognition Algorithm

The database of the project contains 10 images of 40 persons. These images are taken with variations in pose, facial expression and sizes. The normalized size of all images is 92 x 112.

Figure 39: Sample image from database

Table 5-1: Results of Testing on Database

Test	Smiling Faces	Angry Faces	Frontal Face	Other	Total
No. of images	40	40	80	240	400
False	7	6	5	28	46
Correct %	82.5%	85%	93.75%	88.33%	88.5%

4.8 DWT Results

Discrete wavelet transform is applied to reduce the computational complexity. Number of computations can be decreased by using DWT. The syndicate members calculated the time for loading the database before and after applying the DWT.

Table 5-1: Results of Testing on Database

Serial	Database	Time
1	Without DWT	1.458 sec
2	With DWT	0.961 sec

Chapter 5

5.1 Recommendations for future work

5.1.1 Standalone Platform

This project has been mainly implemented on MATLAB, thus using a whole personal computer (PC) for the facial Recognition purpose. For future work, this project can be implemented on a standalone hardware kit like FPGA, arduino, DSP kit etc. thus eliminating a use of a computer for image processing. This will be a great success for anyone extending this project as it will be a whole standalone project and there will be no use of implementing a circuit designed on a PCB.

5.1.2 Biometric Authentication

Moreover other Biometric authentication methods can be combined with the Face Recognition techniques to produce better results. These biometric authentication techniques include

1. Fingerprints Scan
2. Hand geometry
3. Earlobe geometry
4. Retina and iris patterns
5. Voice Recognition

5.1.3 Multiple Face Detection

Multiple face detection and recognition algorithm can be made as it will ensure better efficiency in different conditions and scenarios. Viola jones face detection algorithm only detects one face in an image, so for this project one person at a time can be detected. For multiple face detection, an algorithm which can detect more than one face in the image can be used. It will increase the performance of this system.

Chapter 6

6.1 Conclusion

The RFID and Face Recognition Based Entry System is a prototype developed for the use at any entry point of an organization. This system not only increases the efficiency of the Face Recognition but has made the recognition process faster. Any person entering the premises of an organization can be checked if he is the authenticated person entering the premises or not.

6.1.1 Drawbacks:

Based upon our results, we have following drawbacks in our project.

1. The light must be optimum for the system to recognize the face in the image. This issue has made the project suitable only for those places where the light is optimum.
2. No face can be detected if it is not aligned with the webcam (Non-frontal view of face)
3. Facial changes due to aging are not detected.
4. Due to Serial Communication with the hardware, the system takes time to read the values from the Microcontroller.

6.1.2 Advantages:

The project has following main advantages based on our results.

1. The Face Recognition Process is made faster by accessing a user's folder in the database with the help of the RFID input.
2. The Efficiency of the Face Recognition system is increased by the use of RFID system in the project.

Chapter 7

7.1 Demonstration Outline

All the project features are demonstrated below.

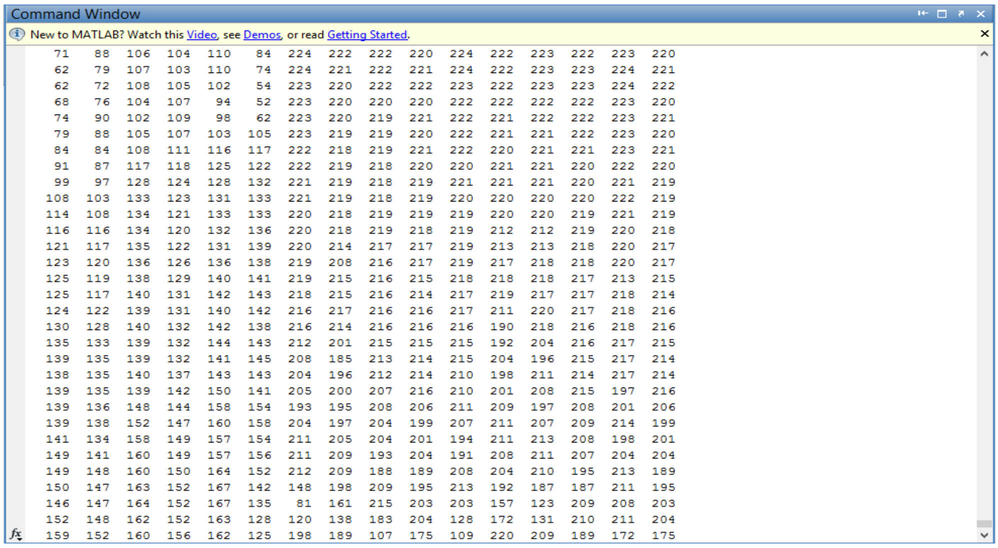
Figure 40: Main Program GUI

The main program GUI consists of the following pushbuttons.

1. Database
2. Webcam
3. New user

4.1.1 Database:

When the database pushbutton is pressed, it loads all the feature vectors of the images stored in the database.



Command Window

New to MATLAB? Watch this Video, see Demos, or read Getting Started.

71	88	106	104	110	84	224	222	222	220	224	222	223	222	223	220
62	79	107	103	110	74	224	221	222	221	224	222	223	223	224	221
62	72	108	105	102	54	223	220	222	222	223	222	223	223	224	222
68	76	104	107	94	52	223	220	220	220	222	222	222	222	223	220
74	90	102	109	98	62	223	220	219	221	222	221	222	222	223	221
79	88	105	107	103	105	223	219	219	220	222	221	221	222	223	220
84	84	108	111	116	117	222	218	219	221	222	220	221	221	223	221
91	87	117	118	125	122	222	219	218	220	220	221	221	220	222	220
99	97	128	124	128	132	221	219	218	219	221	221	221	220	221	219
108	103	133	123	131	133	221	219	218	219	220	220	220	220	222	219
114	108	134	121	133	133	220	218	219	219	219	220	220	219	221	219
116	116	134	120	132	136	220	218	219	218	219	212	212	219	220	218
121	117	135	122	131	139	220	214	217	217	219	213	213	218	220	217
123	120	136	126	136	138	219	208	216	217	219	217	218	218	220	217
125	119	138	129	140	141	219	215	216	215	218	218	218	217	213	215
125	117	140	131	142	143	218	215	216	214	217	219	217	217	218	214
124	122	139	131	140	142	216	217	216	216	217	211	220	217	218	216
130	128	140	132	142	138	216	214	216	216	216	190	218	216	218	216
135	133	139	132	144	143	212	201	215	215	215	192	204	216	217	215
139	135	139	132	141	145	208	185	213	214	215	204	196	215	217	214
138	135	140	137	143	143	204	196	212	214	210	198	211	214	217	214
139	135	139	142	150	141	205	200	207	216	210	201	208	215	197	216
139	136	148	144	158	154	193	195	208	206	211	209	197	208	201	206
139	138	152	147	160	158	204	197	204	199	207	211	207	209	214	199
141	134	158	149	157	154	211	205	204	201	194	211	213	208	198	201
149	141	160	149	157	156	211	209	193	204	191	208	211	207	204	204
149	148	160	150	164	152	212	209	188	189	208	204	210	195	213	189
150	147	163	152	167	142	148	198	209	195	213	192	187	187	211	195
146	147	164	152	167	135	81	161	215	203	203	157	123	209	208	203
152	148	162	152	163	128	120	138	183	204	128	172	131	210	211	204
159	152	160	156	162	125	198	189	107	175	109	220	209	189	172	175

Figure 41: Loading Database

4.1.2 Webcam:

When the Webcam button is pressed, it runs the main program. Firstly it is required that a user must scan his RFID card.

Figure 42: Scanning RFID Card

After it is read by the microcontroller and finally it is read by the MATLAB. It triggers on the Webcam and user's Image is taken.

Figure 43: Webcam Image

After taking the Image, the PCA is implemented and the image taken is matched with user's images in the database.

If the image is matched with a stored image, the gate present at the hardware module is opened.

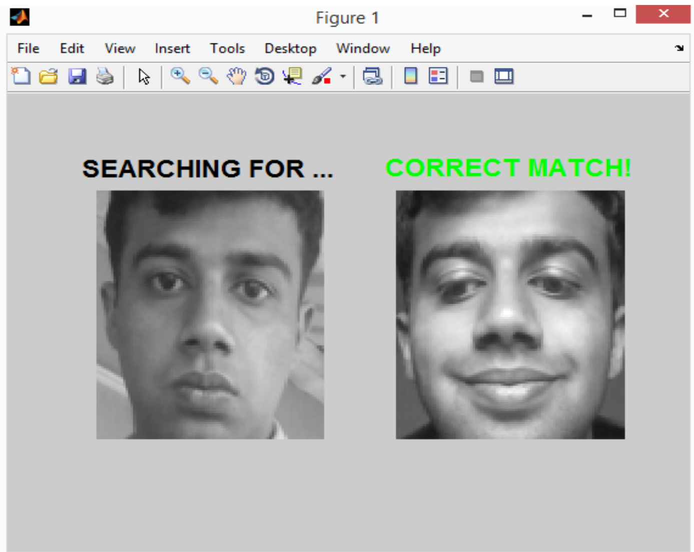


Figure 44: Image is matched

If the image is not found, the gate remains closed.

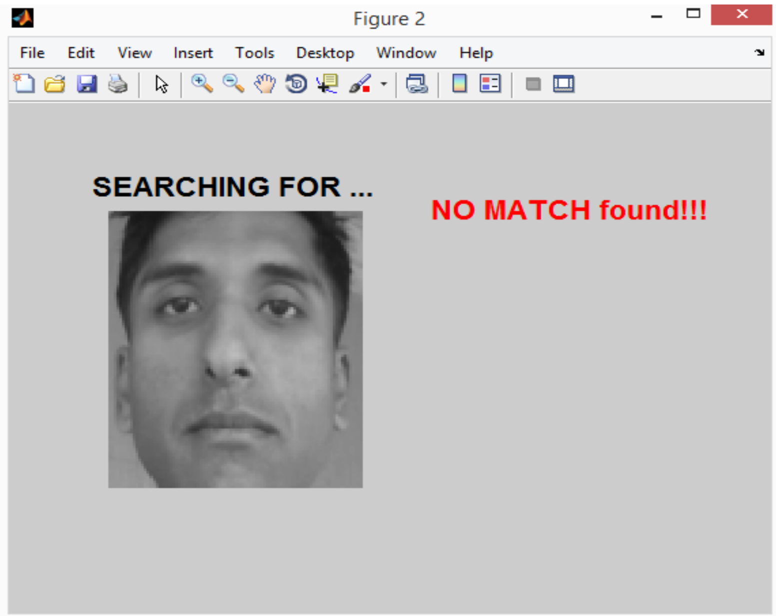


Figure 45: No Match found

4.1.3 New User:

When the new user button is pressed a new GUI window is popped up, prompting the password. When the password entered is correct the following window is displayed.

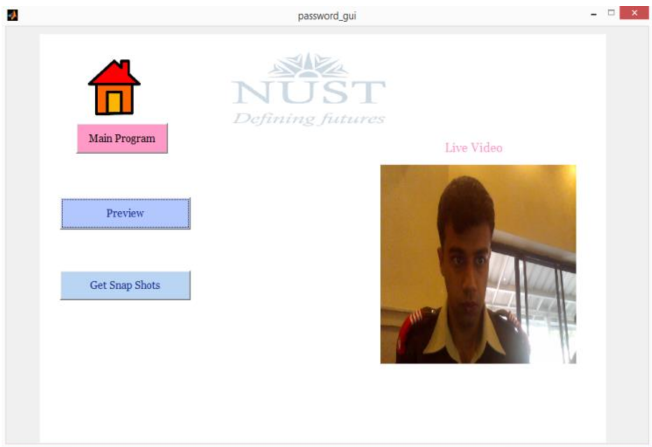


Figure 46: New User Window

When get snapshots button is pressed, the programs takes 10 images of a user and store it in the existing database.

Bibliography

[1] Shahryar A. Khan; "Attendance System Using RFID and Face Recognition," Bahria University, 2010

[2] Adnan Affandi, Mohammed Awedh, Mubashshir Husain, Ahmed Alghamdi; "RFID and Face Recognition Based Security and Access Control System", International Journal of Innovative Research in Science, Engineering and Technology Vol. 2, Issue 11; ISSN: 2319-8753; pp 5955-5964, November 2013

[3] Toshiaki Kondo, Hong Yan, "Automatic human face detection and recognition under Non-uniform illumination", Pattern Recognition 32 pp 1707-1718, (1999)

[4] Moon H, Phillips P J, "Computational and performance aspects of PCA-based face-recognition algorithms", Perception 30(3) PP.303 – 321, 2001

[5] Yasufumi Suzuki and Tadashi Shibata, "Multiple-Clue Face Detection Algorithm Using Edge-Based Feature Vectors", the Second International Conference on Internet Multimedia Computing and Service pp.737-740,2004.

[6] PIC Microcontroller Using Assembly and C; Muhammad Ali Mazidi, Sarmad Naimi, Sepehr Naimi; 792 pages; 2010

[7] *PIC Microcontroller and Embedded Systems: Using Assembly and C*; Muhammad Ali Mazidi, Sarmad Naimi, Sepehr Naimi; 648-650 pages; 2010;

[8] *PIC Microcontroller and Embedded Systems: Using Assembly and C*; Muhammad Ali Mazidi, Sarmad Naimi, Sepehr Naimi; 680-681 pages; 2010;

[9] Marcin Kociołek, Andrzej Materka, Michał Strzelecki, Piotr Szczypiński; "DISCRETE WAVELET TRANSFORM – DERIVED FEATURES FOR DIGITAL IMAGE TEXTURE ANALYSIS";

Appendix A

Code for PIC 18F26K22 Microcontroller

```
#include <18F26k22.h>

#use delay(clock=16000000)

#fuses HSH,NOWDT,NOPROTECT,NOLVP

#use rs232(baud=9600, xmit=PIN_C6, rcv=PIN_C7, stream=UART1)

#use rs232(baud=9600, xmit=PIN_B6, rcv=PIN_B7, stream=UART2)

char ID1[14]="66003E41A5BC";

char ID2[14]="66003E46BAA4";

int8 c=0,counter=0,check=0,open;

char buff[16],rd=0;

int ok_flag=0,rec_flag=0,start_flag=0,open_flag=0;

#INT_RDA

void RDA_isr(void)

{

     c=fgetc(UART1);

      if(c==0x03)

          {

            rec_flag=1;

            start_flag=0;

            counter=0;

          }
```

```
      if(start_flag==1)
     {    buff[counter++]=c;
     }
      if(c==0x02){ start_flag=1;counter=0;}
}
#INT_RDA2
void RDA2_isr()
{
   rd=fgetc(UART2);
   if(rd=='O'||rd=='P'||rd=='E'||rd=='N')open++;
   else open=0;
   if(open>=4){open_flag=1; open=0;}
}
void main()
{
   enable_interrupts(INT_RDA);
   enable_interrupts(GLOBAL);
 int i=0;
   while(TRUE)
   {
      if(rec_flag==1)
      {
         for( i=0;i<15;i++)
```

```
        if(buff[i]==ID1[i]){                              check++;if(check==11){ok_flag=1;
rec_flag=0;break;fputc(8,UART1);}}
        else check=0;
        for( i=0;i<15;i++)
        if(buff[i]==ID2[i])
          {check++;if(check==11){ok_flag=1;rec_flag=0;break;fputc(2,UART1);}}
          else check=0;
          }
      if(open_flag==1)
      {
      output_high(PIN_A0);delay_ms(5000);output_low(PIN_A0);counter=0;open_flag
      =0;
      }
      counter=0;
      check=0;
      delay_ms(200);
    }
}
```

Appendix B

Matlab Code for Face Recognition

```
function varargout = project1(varargin)

gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @project1_OpeningFcn, ...
                   'gui_OutputFcn',  @project1_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end


function project1_OpeningFcn(hObject, eventdata, handles, varargin)
%main program gui


axes(handles.axes11);
imshow('prj.png');
axes(handles.axes12);
imshow('user.png');

[a,map]=imread('webcam.png');
[r,c,d]=size(a);
x=ceil(r/100);
y=ceil(c/80);
g=a(1:x:end,1:y:end,:);
g(g==255)=5.5*255;
set(handles.pushbutton4,'CData',g);


[aa,map]=imread('databas.png');
[rr,cc,dd]=size(aa);
xx=ceil(rr/100);
yy=ceil(cc/80);
gg=aa(1:xx:end,1:yy:end,:);
gg(gg==255)=5.5*255;
set(handles.pushbutton1,'CData',gg);
```

```
function varargout = project1_OutputFcn(hObject, eventdata, handles)

%varargout{1} = handles.output;





function pushbutton1_Callback(hObject, eventdata, handles)    %train
database pushbutton


load_database1()


function pushbutton4_Callback(hObject, eventdata, handles)    %Webcam
detection pushbutton

clear all;
%w=load_database();
v=zeros(10304,10);
% for i=1:13



bc=serial('COM3');
fopen(bc);
u=fscanf(bc);
fclose(bc);
%u=input('enter eigen values')
%for i=1:2
        cd(strcat('s',num2str(u)));
        for j=1:10
            a=imread(strcat(num2str(j),'.pgm'));
            v(:,j)=reshape(a,size(a,1)*size(a,2),1);

        end
         cd ..

       % count=count+1;
   %h=waitbar(count/8);

   %end


    w=uint8(v); % Convert to unsigned 8 bit numbers to save memory.

%end

% loaded=1;  % Set 'loaded' to aviod loading the database again.
% out=w;

%close(h);
```

```
          vid = videoinput('winvideo',2);
          preview(vid);
          pause(3);
          pic = getsnapshot(vid);
          imwrite(pic,'pic.pgm');
          q=imread('pic.pgm');
r=voilajones(q);
          r=imresize(r,[112 92]);
  v=w;
delete('pic.pgm');
N=2;

O=uint8(ones(1,size(v,2)));
m=uint8(mean(v,2));
vzm=v-uint8(single(m)*single(O));


L=single(vzm)'*single(vzm);
[V,D]=eig(L);
V=single(vzm)*V;
V=V(:,end:-1:end-(N-1));

cv=zeros(size(v,2),N);
for i=1:size(v,2);
    cv(i,:)=single(vzm(:,i))'*V ;
end


figure,subplot(121);
imshow(r);
title('SEARCHING                                                    FOR
...','FontWeight','bold','Fontsize',16,'color','black');
subplot(122);

r=imresize(r,[10304,1]);
p=r-m;
s=single(p)'*V;
z=[];
for i=1:size(v,2)
    z=[z,norm(cv(i,:)-s,2)];
    if(rem(i,20)==0),imshow(reshape(v(:,i),112,92)),end;
    drawnow;
end

[a,i]=min(z);
i


subplot(122);
imshow(reshape(v(:,(i)),112,92));title('CORRECT
MATCH!','FontWeight','bold','Fontsize',16,'color','green');
%x=recognition(z)



%msgbox(x)
%!mode com6:9600,n,8,1
```

```
%fid = fopen('com6:','w')

%fwrite(fid,'1','int8')

%fclose all
bc=serial('COM3');
fopen(bc);
fprintf(bc,'OPEN')
fclose(bc);
```

```
function axes6_CreateFcn(hObject, eventdata, handles)

axes(hObject)
imshow('index.jpg')


function    pushbutton9_Callback(hObject,    eventdata,    handles)
%Pushbutton for new registration


password_gui
%regis



function pushbutton10_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton10 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

w=load_database1();

bc=serial('COM3');
fopen(bc);
u=fscanf(bc);
fclose(bc);

         vid = videoinput('winvideo',2);
         pic = getsnapshot(vid);
         imwrite(pic,'pic.pgm');
        q=imread('pic.pgm');
r=voilajones(q);
         r=imresize(r,[112 92]);
  v=w;
```

```
delete('pic.pgm');
N=2;

O=uint8(ones(1,size(v,2)));
m=uint8(mean(v,2));
vzm=v-uint8(single(m)*single(O));


L=single(vzm)'*single(vzm);
[V,D]=eig(L);
V=single(vzm)*V;
V=V(:,end:-1:end-(N-1));

cv=zeros(size(v,2),N);
for i=1:size(v,2);
    cv(i,:)=single(vzm(:,i))'*V ;
end


figure,subplot(121);
imshow(r);
title('SEARCHING                                           FOR
...','FontWeight','bold','Fontsize',16,'color','black');
subplot(122);

r=imresize(r,[10304,1]);
p=r-m;
s=single(p)'*V;
z=[];
for i=1:size(v,2)
    z=[z,norm(cv(i,:)-s,2)];
    if(rem(i,20)==0),imshow(reshape(v(:,i),112,92)),end;
    drawnow;
end

[a,i]=min(z);
i





subplot(122);
imshow('');title('NO                                            MATCH
found!!!','FontWeight','bold','Fontsize',16,'color','red');
bc=serial('COM3');
fopen(bc);
fprintf(bc,'OPEN')
fclose(bc);





function edit1_Callback(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit1 as text
```

```
%         str2double(get(hObject,'String')) returns contents of edit1
as a double


% --- Executes during object creation, after setting all properties.
function edit1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     empty - handles not created until after all CreateFcns
called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if        ispc        &&        isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% --- Executes on button press in pushbutton11.
function pushbutton11_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton11 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
close();


% --- Executes on button press in pushbutton13.
function pushbutton13_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton13 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
```

Password GUI

```
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @password_gui_OpeningFcn, ...
                   'gui_OutputFcn',  @password_gui_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
```

```
% End initialization code -


% --- Executes just before password_gui is made visible.
function    password_gui_OpeningFcn(hObject,    eventdata,    handles,
varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to password_gui (see VARARGIN)

% Choose default command line output for password_gui
handles.output = hObject;

%%%%%%%%%%%%55

axes(handles.axes3);
imshow('nust.png')

axes(handles.axes1);
imshow('no.jpg')

axes(handles.axes4);
imshow('hom.jpg')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Update handles structure
guidata(hObject, handles);

set(gcf, 'visible','off')

% question for password
%c = cell(1)
password='nust';
prompt = {'Enter password'};
dlg_title = 'Password';
num_lines = 1;
def = {'????','hsv'};
answer_x = inputdlg(prompt,dlg_title,num_lines,def);
answer_xx=cell2struct(answer_x, 'word',1);
answer=answer_xx.word;

if answer==password
    set(gcf, 'visible','on')
else
    close all
end

% UIWAIT makes password_gui wait for user response (see UIRESUME)
% uiwait(handles.figure1);


% --- Outputs from this function are returned to the command line.
```

```
function   varargout   =   password_gui_OutputFcn(hObject,   eventdata,
handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;


% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)



axes(handles.axes1);
vidObj = videoinput('winvideo',2);
videoRes = get(vidObj, 'VideoResolution');
numberOfBands = get(vidObj, 'NumberOfBands');
handleToImage     =     image(     zeros([videoRes(2),     videoRes(1),
numberOfBands], 'uint8') );
preview(vidObj, handleToImage);
handles.vidObj = vidObj;
    guidata(hObject,handles)




% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

 global p

%vid = videoinput('winvideo',2);

for   i=1:10

         pic = getsnapshot(handles.vidObj);
         imwrite(pic,'pic.pgm');
        q=imread('pic.pgm');
r=un(q);


  a= strcat(int2str(i),'.pgm') ;
  imwrite(r,a)

   b=imread (a);
   b=imresize(b,[112 92]);
  %  n=input('give the image name to b used for saving the image in
pgm format', 's')
   % a= strcat(int2str(i),'.pgm')
```

```
    imwrite(b,a);
end

%[s, mess, messid] = mkdir('C:\Users\BADAR MUNIR\Desktop\DR IMRAN',
's');
  files = dir('*.pgm');
  p=7;
  p=p+1;
  b= strcat('s',int2str(p))

for k = 1:numel(files)-4


%r = imread(files(k).name);
 a= strcat(int2str(k),'.pgm') ;

movefile(a,b)

 %imwrite(I, ['cropped\' files(k).name]);
  cd .

end
 copyfile('resize.m',b);
  cd(b);

  resize()
  cd .



%frame = getsnapshot(handles.vidObj);
 %   imshow(frame);


% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
project1
```

Load Database

```
function out=load_database1();
% We load the database the first time we run the program.
%waitbar(0,'Please wait...');

%count=1;
%h=waitbar(0,'Please wait...');

persistent loaded;
persistent w;
if(isempty(loaded))
    v=zeros(10304,43);
```

```
    for i=1:43
        cd(strcat('s',num2str(i)));
        for j=1:10
            a=imread(strcat(num2str(j),'.pgm'));
            v(:,(i-1)*10+j)=reshape(a,size(a,1)*size(a,2),1);

        end
         cd ..

       % count=count+1;
   %h=waitbar(count/8);

    end


    w=uint8(v); % Convert to unsigned 8 bit numbers to save memory.

end

loaded=1;  % Set 'loaded' to aviod loading the database again.
out=w;

%close(h);
```