FINGER VEIN EXTRACTION AND AUTHENTICATION FOR SECURITY PURPOSE

A PROJECT REPORT

Submitted by

AAYUSH SINHA [Reg No: 1031310150]
AJAY BHADU [Reg No: 1031310312]

Under the guidance of

Mr. A. Selvakumar

(Assistant Professor, Department of Computer Sciene & Engineering)

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENE & ENGINEERING

of

FACULTY OF ENGINEERING AND TECHNOLOGY

SRM UNIVERSITY
(Under section 3 of UGC Act 1956)

S.R.M. Nagar, Kattankulathur, Kancheepuram District

MAY 2017

SRM UNIVERSITY

(Under Section 3 of UGC Act, 1956)

BONAFIDE CERTIFICATE

Certified that this project report titled "**FINGER VEIN EXTRACTION AND AUTHENTICATION FOR SECURITY PURPOSE**" is the bonafide work of " **AAYUSH SINHA [Reg No: 1031310150], AJAY BHADU [Reg No: 1031310312], , ,** ", who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Mr. A. Selvakumar
GUIDE
Assistant Professor
Dept. of Computer Sciene & Engineering

SIGNATURE

Dr. A. Amutha
HEAD OF THE DEPARTMENT
Dept. of Computer Sciene & Engineering

Signature of the Internal Examiner

Signature of the External Examiner

ABSTRACT

Biometrics are in use in authentication and security since long. The most common biometric methods are fingerprint, iris scanner, face recognition, voice recognition, etc. But the fact is that these methods are not failsafe or fully secure. Hence, newer methods have been introduced in recent times that are more efficient and reliant. One of the best methods of biometric authentication is finger-vein technology. It is highly secure, reliable and efficient biometric mechanisms. In this project, we study the complete implementation of finger-vein authentication system and use it as a biometric method to provide vehicle security. The project has two components. The software component deals with the complete image processing process using MATLAB and checks for authenticity of the input image. We propose a finger-vein technique that uses HAAR transform in the Discrete Wavelet Transform to perform the feature extraction and SVM classifier for training the data set. The input image and the stored images are compared for verification. The performance is measured using many parameters like PSNR and MSE. Hardware component uses the Arduino microcontroller which will be used to decide whether the ignition will be switched on or not depending on the result of the authentication. A GSM component will be utilised to alert the user of any unauthorised access.

ACKNOWLEDGEMENTS

We owe a great many thanks to the following people who helped and supported us in the preparation of the project. We would like to thank our Director Dr. C. MUTHAMIZHCELVAN, Faculty of Engineering and Technology, SRM University, Chennai. We would like to express our deep sense of gratitude to Dr. B. Amutha, Professor and Head of Department of Computer Science and Engineering, SRM University, Chennai, for giving us the opportunity to take up this project. We would also like to thank our project coordinator, Mr. K. Senthil Kumar, Professor, Department of Computer Science and Engineering, SRM University, Chennai, for his valuable inputs during the project reviews. We wish to express our deep sense of gratitude to Mr. Nirmal Sam, Asst. Professor, Department of Computer Science and Engineering, SRM University, Chennai, for his valuable help during project reviews. We would also like to thank Mrs. Ms.Vaishnavi Moorthy, Asst. Professor, Department of Computer Science and Engineering, SRM University, Chennai, for her guidance to complete the project. We wish to express our sincere thanks to Mrs S.Priya, Asst. Professor, Department of Computer Science and Engineering, SRM University, Chennai, for her valuable contributions in making this project We take immense pleasure in conveying our gratitude to our guide, Mr. A. Selva Kumar, Asst. Professor, Department of Computer Science and Engineering, SRM University, Chennai, for his immense support and cooperation during every stage of the project. Several other people also contributed to the project. We sincerely acknowledge all their efforts which was indispensable for the completion of the project.

AAYUSH SINHA

AJAY BHADU

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

PINS	Personal Identification Pins
IDE	Integrated development environment
CCD	Charged Coupled Device
MATLAB	Matrix Laboratory
SVM	Support Vector Machine
DWT	Discrete wavelet transform
GSM	Global System for Mobile Communication
MSE	Mean squared error
HWT	HAAR Wavalet Transforms
GSM	Global System for Mobile
ATM	AUTOMETIC TRAILER MACHINE

CHAPTER 1

INTRODUCTION

1.1 BIOMETRICS

Historically, many methods for security of private information or a device have been invented. Most common means of protection used are passwords or Personal Identification Pins (PINS) (personal identification numbers), but the security provided by these methods are minimal as they can easily be bypassed using trivial methods. There is always the risk of the password being forgotten or a simple password being used. Many hacking methods are there to bypass such methods. Hence to improve the security greatly, biometric methods for security have been introduced. Biometric methods are methods that use features of human beings for authentication. Since the features such as fingerprints, voice, iris, etc are unique in each person, the security provided is very high when compared to conventional methods like passwords. These may include fingerprints, palm prints, iris scanners, voice recognition, face recognition etc. As we show subsequently, the current biometric methods are subject to risk and failure. Face recognition system depends on many external factors such as lighting, hair style of the person, etc. Aging also alters the appearance of a person significantly. This makes face recognition very less secure. Fingerprint can easily be forged or tampered and any changes to the external skin of the finger can pose a problem for authentication. A recorded voice can be used to bypass a voice recognition system. Events such as a cold can change the voice of a person. A high resolution photograph of a person can similarly be used for bypassing a face recognition system. Iris scan methods are expensive and take a lot time for authentication.

1.2 CURRENT REQUIREMENT OF A BIOMETRIC SYSTEM

The need of a biometric method that is reliable and fast is solved by the finger vein recognition system. This is a method of biometric authentication which uses the patterns of the vein beneath the skin of the finger. Since this pattern is unique for each finger, verification of identity can be done. As the blood vessels are beneath the surface of the skin, forgery is not possible. Also, the physical presence of the human is guaranteed as the authentication cannot be done otherwise. The authentication process matches the previously recorded pattern available in the database for authorisation. This technology was developed by Hitachi and is patented by them. The applications of fingerprint technology are in vehicle security, AUTOMETIC TRAILER MACHINE (ATM) security, employee time and attendance tracking, computer authentication, etc. Finger-vein system solves the need of a highly secure biometric process which is very efficient, fast and reliable.

1.3 ADVANTAGES OF FINGER-VEIN

- **Accuracy:** Finger-vein authentication is extremely accurate as wrongful acceptance or true-user rejection rates are one of the lowest amongst biometric authentication.
- **Secure:** Finger-vein technique is immensely secure as forgery is impossible. This is due to the fact that the veins are inside the skin, and forgery cannot be done. Also, physical alteration in the skin such as dryness, etc have no impact on this.
- **Size:** The devices is small, and hence can be used in a wide variety of applications.
- **Not visible:** Since the veins are inside the skin, they are invisible to the naked eye.
- **Back-up:** A user can have multiple vein patterns of different fingers stored as each finger will have a separate vein pattern, giving extra security.

CHAPTER 2

EXISTING SYSTEMS AND DRAWBACKS

2.1 Overview of Existing System

A biometric system is basically a system that uses the physical characteristics or traits of a person to identify that person accurately. Since these traits or characteristics such as eyes, voice, fingerprints, face, etc are unique from person to person, the authentication of that person is reliable. The biometric system will collect the physical data from a person and run an algorithm for a particular desired result. Authentication takes place when a new sample is matched with the previously recorded sample stored in the database. The current biometrics systems are all subject to risk and have a higher rate of failure. A comparison of biometric methods is shown in Table 2.1.

2.2 Different types of Biometric System

2.2.1 Iris recognition

Iris scanner is very expensive and it takes a long time for the authentication process. Lot of memory is also used up. Iris scanner will not be able to work correctly, if people wear lenses. External features such as eyelashes might also create problems for accurate recognition.

2.2.2 Facial recognition

Face recognition uses a system, which is dependent on changes in lighting, the person's hair, whether the person wears glasses or not and his age, as people’s faces change over time. Facial recognition will only be accurate, if the image of user’s face is evenly lit, which is not always possible and can hence become very hard for recognition.

Table 2.1: COMPARISON OF BIOMETRIC METHODS

Biometric System	Accuracy	Security	Attack Stoppage	Cost
Finger Vein	Very High	Very High	very high	Medium
Finger Vein	Very High	Very High	very High	Medium
Finger Print	High	High	High	Low
Voice Recognition	Medium	Medium	Medium	Medium
Iris Recognition	High	High	High	High
Facial Recognition	Medium Low	Medium	Medium	High

2.2.3 Voice recognition

Voice recognition has low accuracy. A person's voice can be easily recorded and used for authorization. An illness such as a cold or growing older can change a person's voice or making identification difficult or impossible.

CHAPTER 3

SYSTEM REQUIREMENTS

3.1 Hardware Requirements

The hardware requirements may serve as the basis for a contract for the implementation of the system and should therefore be a complete and consistent specification of the whole system. They are used by software engineers as the starting point for the system design. It shows what the system does and not how it should be implemented.

PROCESSOR	:	Intel(R) Core(TM) i7-5500U 2.40 GHz
RAM	:	8.00 GB
MONITOR	:	14.1”
HARD DISK	:	1 TB
CONTROLLER	:	Arduino
TRANSFORMER	:	Step-Down

3.2 Software Requirements

The software requirements are the specifications of the system. It should include both a definition and a specification of requirements. It is a set of what the system should do rather than how it should do it. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating cost, planning team activities, performing tasks and tracking the team’s and tracking the team’s progress throughout the development activity.

FRONT END	:	MATLAB AND C SHARP
OPERATING SYSTEM	:	Windows 10
IDE	:	MATLAB

CHAPTER 4

LITERATURE SURVEY

TITLE : Multimodal biometric authentication based on score level fusion of finger biometrics.
AUTHORS : Jialiang Peng a, Ahmed A. Abd El-Latif b,âĹŮ, Qiong Li c, Xiamu Niu c.
YEAR : 2014.

This article has presented a multimodal biometric authentication approach based on the score-level fusion of finger vein, fingerprint, finger shape and finger knuckle print. Multimodal biometric scores are fused by the t-norms, which do not require any learning or training with less computational complexity. The experimental tests are performed on a virtual multimodal biometric database. The results show that the finger multimodal biometrics using the proposed score-level fusion approach, SW t-norm, has made remarkable improvement over the individual finger modalities as well as other traditional score-level fusion methods like Max, Min, Sum, Weighted Sum, SVM and LR-G. In addition, the comparative experiments ascertain that the proposed SW t-norm fusion approach outperforms the other existing t-norms fusion approaches.

TITLE : A GSA- based Method in Human Identification Using Finger Vein Patterns
AUTHORS : Fateme Saadat and Mehdi Nasri
YEAR : 2016.

In this paper, a biometric system has proposed for human authentication. The proposed method used patterns of three different finger veins and fused them using score level fusion strategy. Tuning the weights of sum score level fusion using heuristic optimization strategy. Experimental results confirmed that using a heuristic method could lead to more accurate identification accuracy. The performance of the proposed heuristic score level fusion could be investigated and may improve using more advanced heuristic and metaheuristic optimization technique.

TITLE : Human Identification using Finger Images.

AUTHORS : Ajay Kumar, Yingbo Zhou.

YEAR : 2012.

In this paper, They have presented a complete and fully automated finger image matching framework by simultaneously utilizing the finger surface and finger subsurface features, i.e., from finger texture and finger vein images. They presented rigorous experimental results on the database of 6264 images acquired from 156 subjects, over a period of 11 months, to illustrate the significant improvement in the performance than those using conventional finger vein identification approaches. Secondly, They presented a new algorithm for the finger vein identification which can more reliably extract the finger vein shape features and achieve much higher accuracy than previously proposed finger vein identification approaches. Their finger vein matching scheme works more effectively in a more realistic scenarios and leads to more accurate performance as demonstrated from the experimental results. Thirdly, they proposed and investigated two new score level combination approaches, nonlinear and holistic, for effectively combining simultaneously generated finger vein and finger texture matching scores. The nonlinear approach consistently performed better than other promising approaches, i.e., average, product, weighted sum, Dempster-Shafer and likelihood ratio approaches considered in this work. Fourthly, they examined a complete and fully automated approach for the identification of low resolution finger surface/texture images for the performance improvement. This investigation and the obtained results are significant as they point towards the utility of touchless images acquired from the webcam for personal identification and its extension for other utilities like mobile phones, surveillance cameras and laptops. Finally, the availability of the acquired database from this work for the benchmarking/comparison will help to further the research efforts in this area. Currently there is no publicly available database for the performance comparison and research efforts on finger vein identification. The availability of this database, acquired in more realistic conditions, will compliment similar efforts as in. This investigation and the obtained results are significant as they point towards the utility of touchless images acquired from the webcam for personal identification and its extension for other utilities like mobile phones, surveillance cameras and laptops.

TITLE : Palmprint recognition method based on score level fusion

AUTHORS : Shuwen Zhang, Xuxin Gu.

YEAR : 2012.

In this paper, they propose a fusion scheme based on score level fusion. They first get the directional matching score of the two images by the competitive coding method. Moreover, we get the global matching score of the two images by the TPTSR method. At last, they combine the two scores to classify the test sample. Their experimental results show that the matching rate is the highest when the global matching score is weighted 0.9 in TCF method. A series of verification and identification experiments are performed on the multi-spectral palmprint database with 6000 samples from 500 individuals.

CHAPTER 5

METHODOLOGY AND ALGORITHMS

The finger vein authentication system is a method in which the pattern of the veins beneath the surface of the skin is captured. This pattern is unique for each finger of the person. A small device is used in which the person places his finger. This device consists of near infrared LED light emitter and a charge coupled device Charged Coupled Device (CCD) camera. The haemoglobin present in the blood absorbs this near infrared light which makes the vein appear in a unique pattern. The CCD camera captures this pattern. This pattern is stored in the database and compared during authentication process. According to the results the validation is done. The whole process is very accurate and fast and takes less than 0.5 seconds to match and validate.As in figure 5.1.

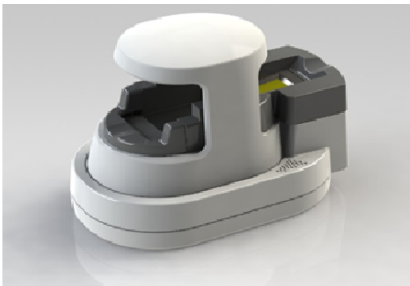


Figure 5.1: A finger-vein scanner device.

This project proposes to use finger-vein technology for authentication and security purpose by using the finger vein reader S-E3F1-609UE by Hitachi which will be used for capturing finger-vein images of various test users. The images are stored in the database and Matrix Laboratory (MATLAB) language is used for image processing to extract the features of the images, so that authentication can be performed.MATLABis chosen to perform image processing since its ease of doing image processing in it.MATLAB has a dedicated image processing toolkit that makes this task simple. It's very large number of built-in algorithms for image processing also help a lot. All the steps of image processing are performed. Before the algorithms for feature extraction can be implemented, image pre-processing steps like noise removal (Gaussian Bilateral Filter), image enhancement (Histogram Equalization), segmentation K-Means Clustering) are done. Different algorithms are used for various tasks. Once the image is ready, the feature extraction takes place. The algorithm used for feature extraction is the Discrete Wavelet Transform- HAAR wavelet. Various parameters are extracted using the above algorithm, which will then be used for authentication process. Finally, Support Vector Machine (SVM) (Support Vector Machine) algorithm is used for training the data set into whether it is an authenticated image or not. The flowchart is shown in the fig 5.2.

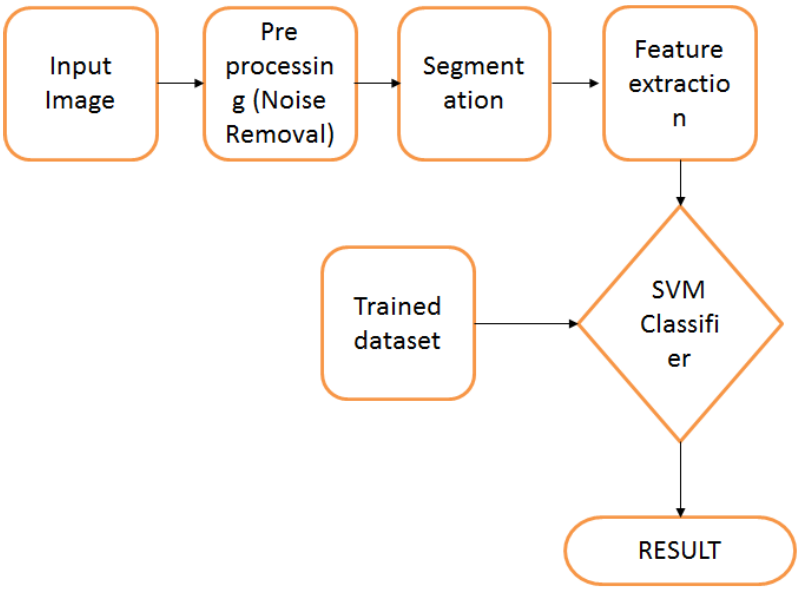


Figure 5.2: Block diagram of the whole process

This section covers the detailed processes and algorithms that were used in each of the process.

5.1 Noise Removal

Digital images are subject to many types of noise. Noise is the result of errors in the image acquisition process that result in pixel values that do not reflect the true intensities of the real scene. For image processing to take place, noise removal is important. Feature extraction will only be accurate if the image is free from noise. For this project, Gaussian Bilateral Filter is used for noise removal. As in figure5.3.

Figure 5.3: The application of Gaussian Bilateral Filter.

5.2 Image Enhancement

Image enhancement is basically improving the quality of the digital image when the source of the degradation is not known. To improve the contrast, the method used is Histogram Equalization. figure 5.4 and figure 5.5 .

5.2.1 Histogram Equalization

In image processing, histogram equalization is a technique to enhance contrast by modifying image intensities. The histogram of the image is drawn, before and after the equalization technique is applied. The enhanced image will have flattened or smoother histograms.

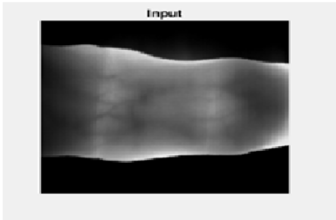


Figure 5.4: The input image of finger-vein

Figure 5.5: Histogram Equalization

5.3 Image Segmentation

In the input image, not all parts of the image is useful to perform feature extraction. Hence, using the technique of image segmentation, the input image will be divided into parts multiple parts. The basic goal of image segmentation is to modify or change the representation of the image in such a way that makes further processing easier. These separate part are known as super pixels. The pixels in each segment contain the same label. The part which contains the region of interest will then be used for the subsequent step- feature extraction.For image segmentation we use K-Means Clustering algorithm, which is explained below. figure 5.6 .

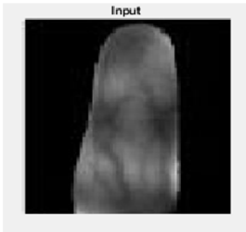


Figure 5.6: The input image of finger-vein

5.3.1 K-Means Clustering

K -means clustering algorithm is an unsupervised algorithm and it is used to segment the interest area from the background. K-Means Clustering is a least square partitioning method, which divides a collection of objects into certain K groups. The mean of each cluster and the distance of each point from each cluster is computed. K-means clustering treats each object as having a location in space. It finds partitions such that objects within each cluster are as close to each other as possible, and as far from objects in other clusters as possible. K-means clustering requires that you specify the number of clusters to be partitioned and a distance metric to quantify how close two objects are to each other.

5.4 Feature Extraction

A feature is a distinctive attribute or aspect of something that has some values or parameter which distinguishes itself from its counterparts. Feature extraction starts with an initial set of parameters and builds features that are unique. In image processing, various algorithms are used to detect the desired portion and extract the features of that region. One of the most common algorithms used for feature extraction is by using wavelet transform. Wavelets are a mathematical tool for hierarchically decomposing functions in multiple hierarchical sub bands with time scale resolutions. Since wavelet transform is not Fourier based, the areas where there is a discontinuity in data are handled better. Under the Discrete Wavelet Transforms Discrete wavelet transform (DWT), we use the HAAR Wavelet Transform HAAR Wavalet Transforms (HWT) as it has the easiest implementation of the DWT.

5.4.1 HAAR Wavelet Transform

HWT is used as it is one of the easiest of the wavelet transform. This transform cross multiplies a function with shifts, just like the Fourier transform. In the HAAR transform, the input and the output lengths are the same, but the length should be in the power of 2. When applied for image processing, the image is transformed into a matrix, in which elements of a matrix represents each pixel of the image. Now, any transformations made in the matrix will alter the pixel information of the image. Hence, the HAAR transform is derived from the HAAR matrix. Discrete HAAR functions are function determined by sampling the HAAR functions at 2 power n points.

5.5 Parameters Measured

After running the DWT-HAAR wavelet transform algorithm for the feature extraction, various parameters of the image are calculated. These parameters define the image uniquely. They are mentioned below:

- **Mean:** Mean value gives the contribution of individual pixel intensity for the entire image. It is basically a texture feature that represents the average pixel value of the given image. Mean is used in noise filtering as well.
- **Variance:** Variance is normally used to find how each pixel varies from the neighbouring pixel (or centre pixel) and is used in classify into different regions. It can also be used to determine edge position
- **Standard Deviation:** Standard Deviation shows how much variation or dispersion exists from the average (mean, or expected value). A low value of standard deviation shows that the points are in proximity to the mean while high standard deviation shows the points are widely spread. The standard deviation is calculated and the value is assigned to the central pixel.
- **Entropy:** Entropy gives information about the texture of the image. Entropy can be measured from the histogram of an image. In an 8-bit pixel there are 256 states. If all such states are equally occupied, spread of states is maximum. In this case, the entropy value will be maximum and vice-versa.
- **Skewness:** Skewness is gives an idea about the surface of the image. Darker surfaces have higher skew values than lighter values. Skewness can be used to measure the edge of an image that has a white background.
- **Kurtosis:** : In general, Kurtosis is used to determine if a distribution is flat or has a peak. It is a measure of the shape of the distribution. Kurtosis values are related to noise and resolution of the image.

5.6 Training dataset-Support Vector Machine

Support Vector Machine is an algorithm that is used with learning algorithms that are used for classification. SVMs are one of the best known methods for pattern and image classification. SVMs are used to separate a set of training in two different classes. Hence, when there are two classes and the image is to be classified as one of them, the

SVM classifier is used. Based on the features extracted of the input images, the data set is trained to be in one of the categories of ’authenticated’ or ’not authenticated’. SVM builds a separating hyper-plane based on a kernel function. If the feature vector lies on one side of the plane, it is categorised as belonging in the first class and if it lies on the other side, it is classified as belonging in the second class.

CHAPTER 6

ARCHITECTURAL DIAGRAM

An architectural diagram is one that represents all the components of the systems and how they are connected with each other. It provides a clear insight about all the components.

6.1 USE CASE DIAGRAM

A use case diagram is graphic depiction of the interactions among the elements of a system. A use case is a methodology used in system analysis is identify, clarify, and organize system requirements.

6.2 DATAFLOW DIAGRAM

In the unified modeling language, a component diagram depicts how components are wired together to form larger component and or software system. They are used to illustrate the structure of arbitrarily complex system.

6.3 CLASS DIAGRAM

A class diagram is an illustration of the relationships and source code dependencies among classes in UML.

6.4 STATE DIAGRAM

A state diagram is a type of diagram used in computer science and related fields to describe the behavior of systems. State Diagram require that the system described is

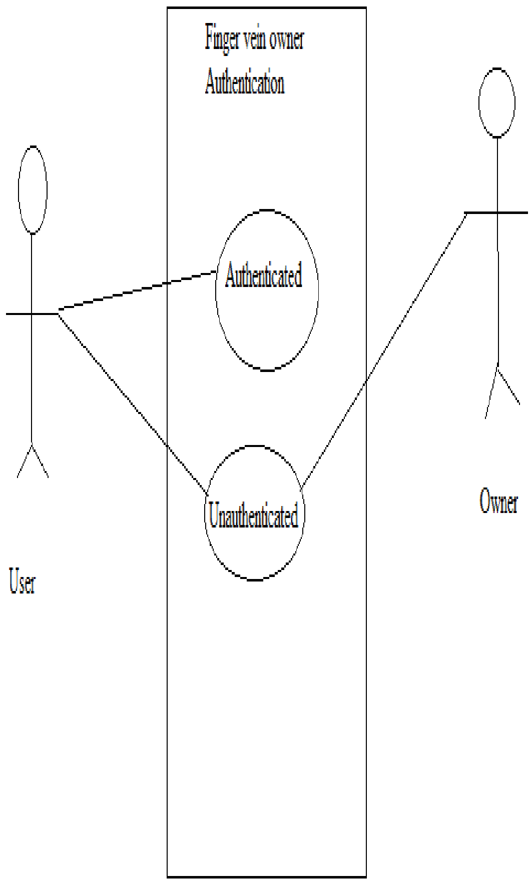


Figure 6.1: USE CASE DIAGRAM

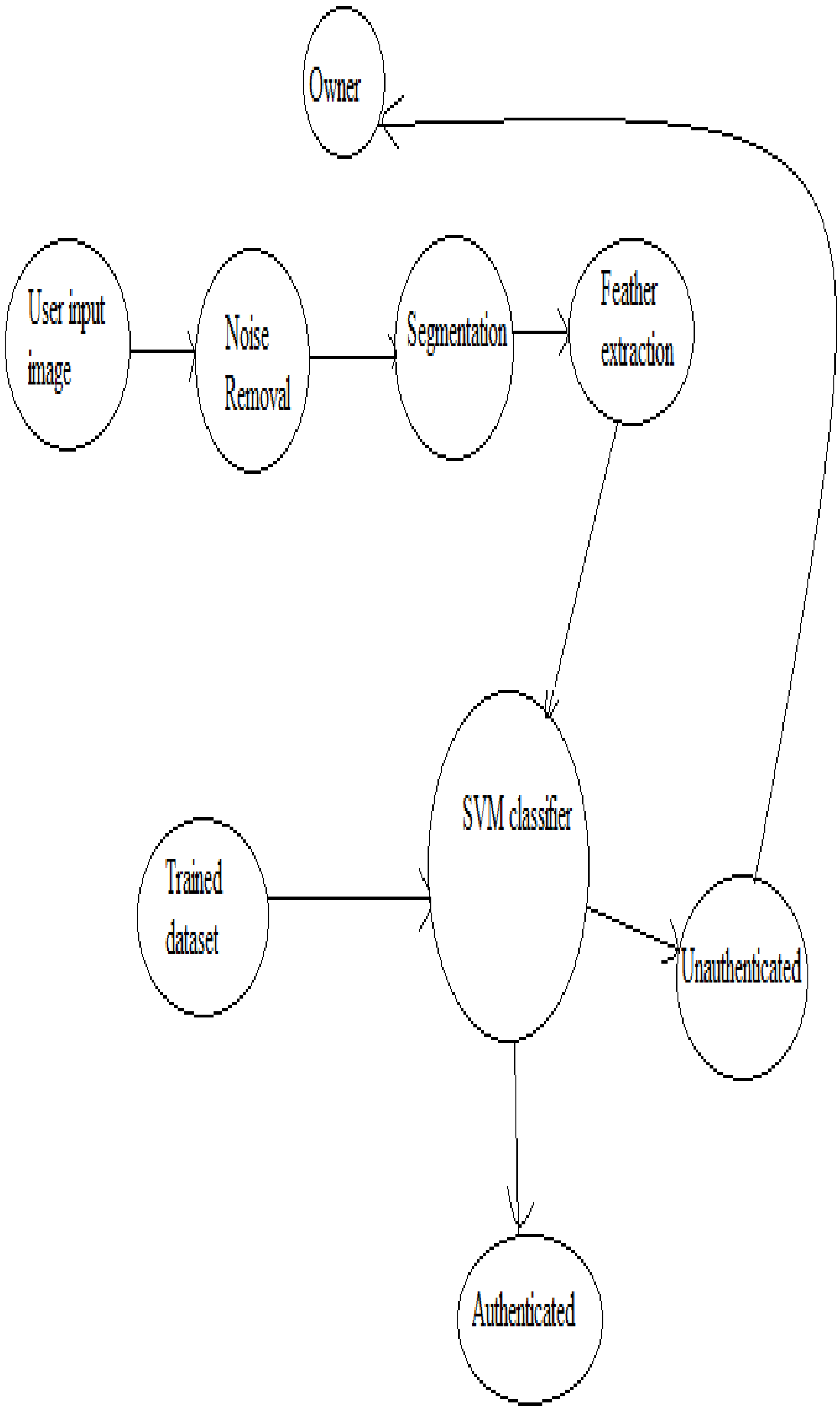


Figure 6.2: DATAFLOW DIAGRAM

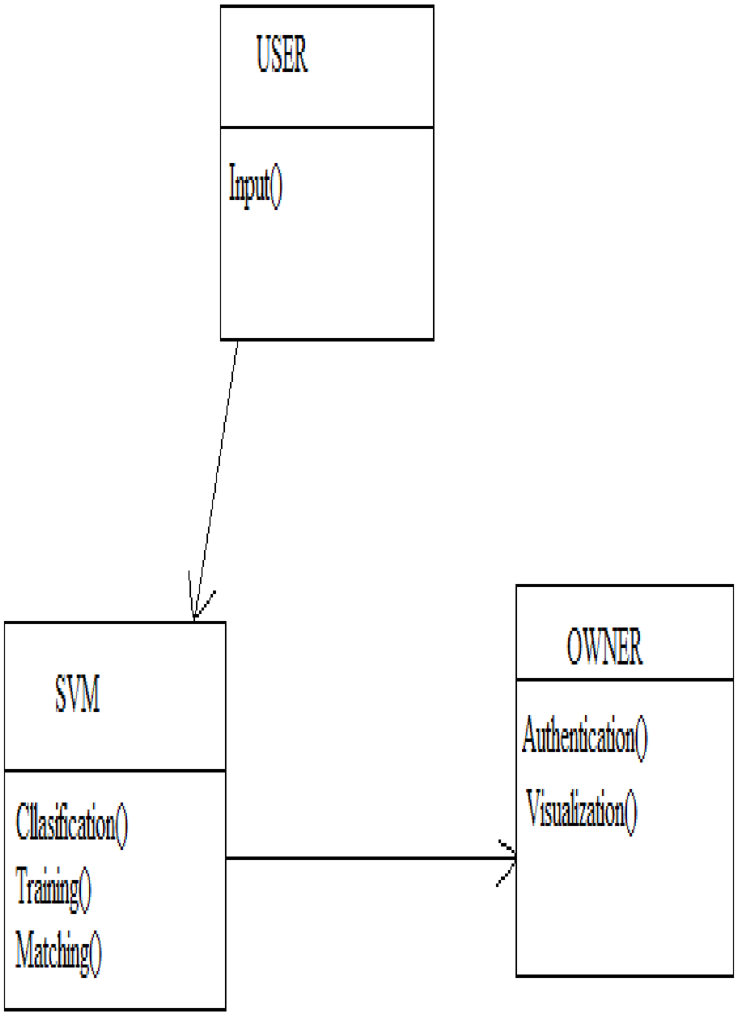


Figure 6.3: CLASS DIAGRAM

composed of a finite number of states: sometimes, this is indeed the case, whileat other times this is a reasonable abstraction.

6.5 COLLABORATION DIAGRAM

Collaboration diagram conveys the same information as sequence diagram, but focus on object roles instead of the times that messages are sent. In a sequence diagram, object roles are the vertices and messages are the connecting links.

6.6 SEQUENCE DIAGRAM

A Sequence diagram is an intraction diagram that shows how processes operate with one another and in what order. It is construct of a message sequence chart. A sequence Diagram shows object interaction arranged in time sequence.

6.7 ACTIVITY DIAGRAM

Activity diagrams are graphical representation of workflows of stepwise activities and actions with support for choice,iteration and concurrency.

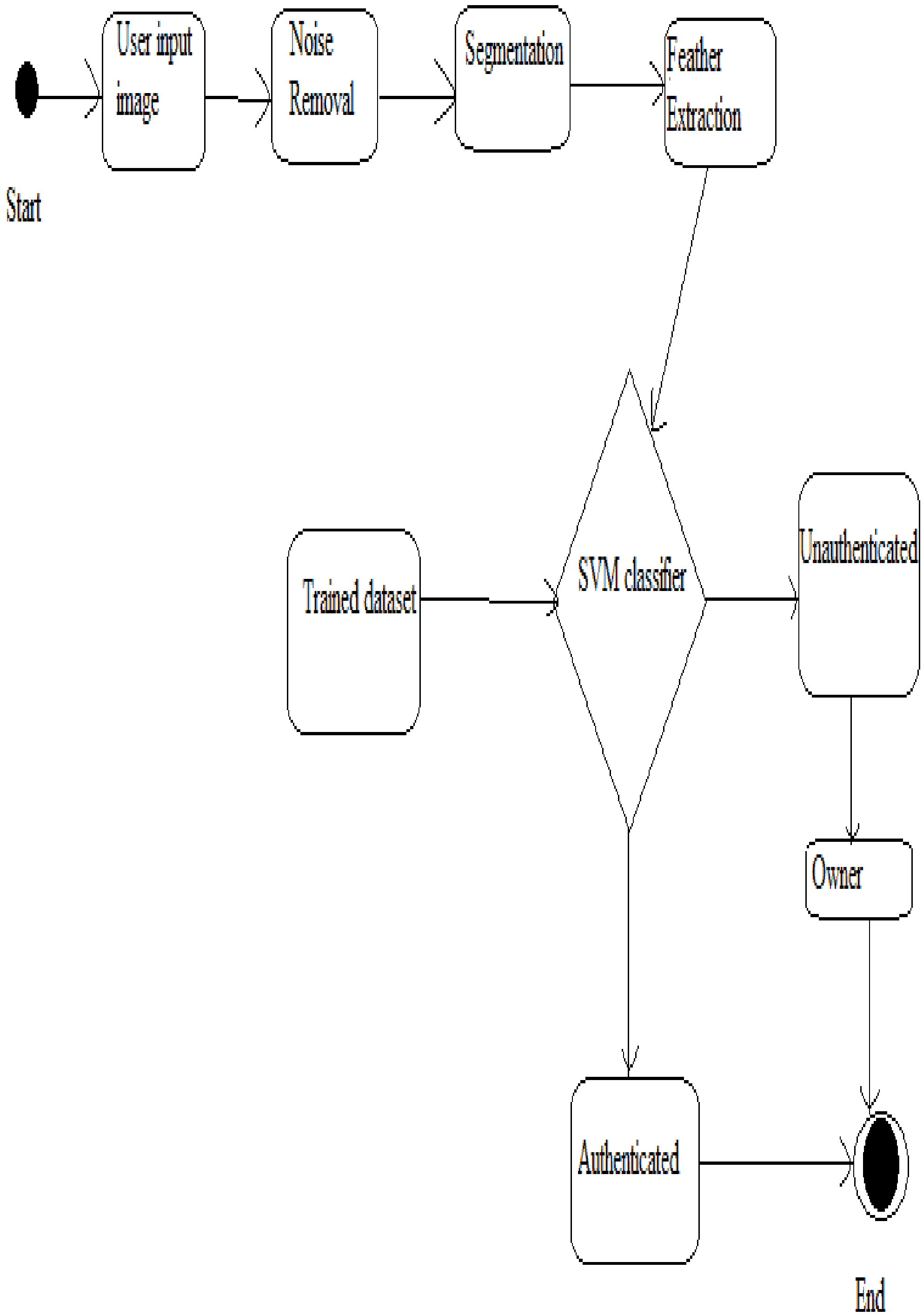


Figure 6.4: STATE DIAGRAM

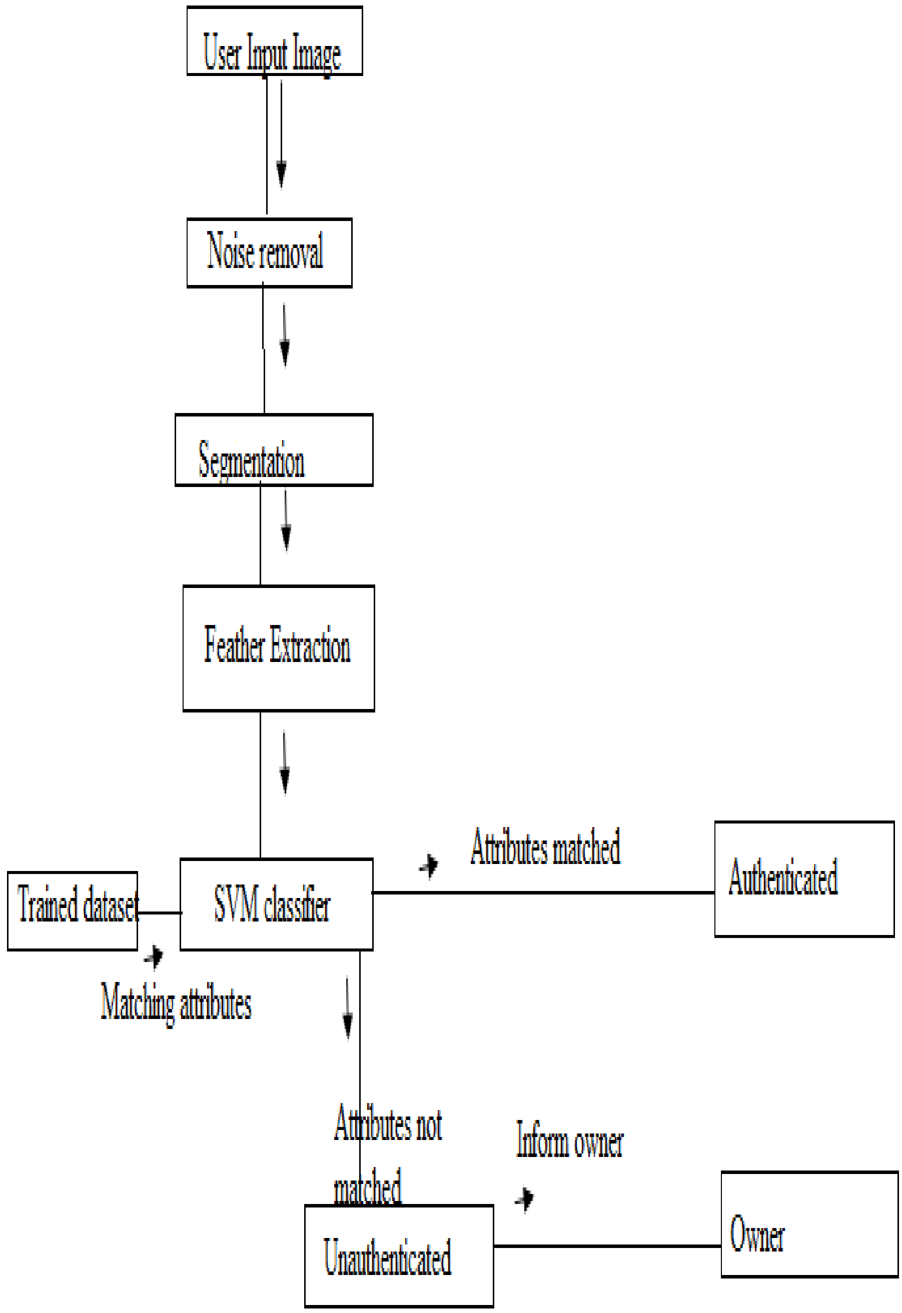


Figure 6.5: COLLABORATION DIAGRAM

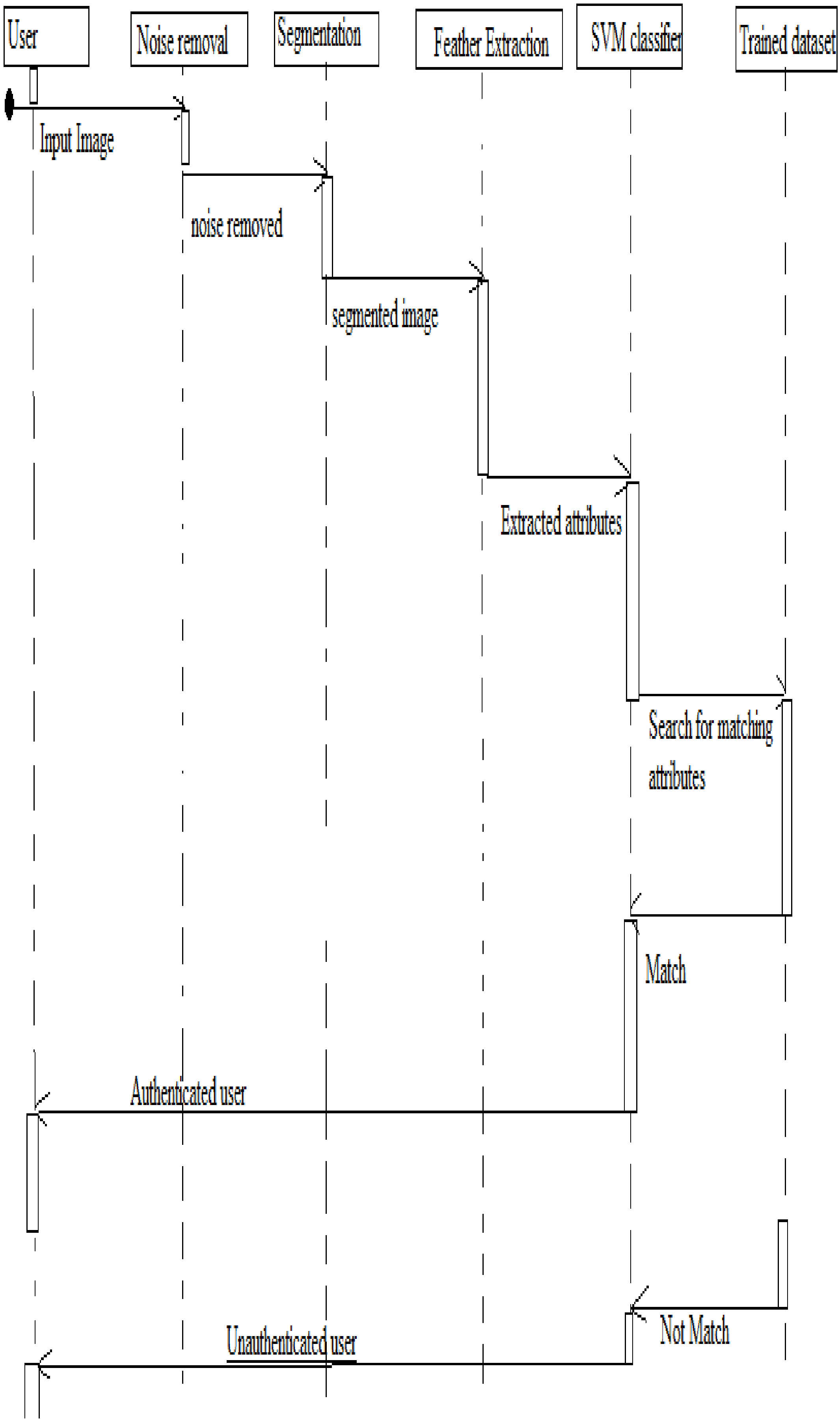


Figure 6.6: SEQUENCE DIAGRAM

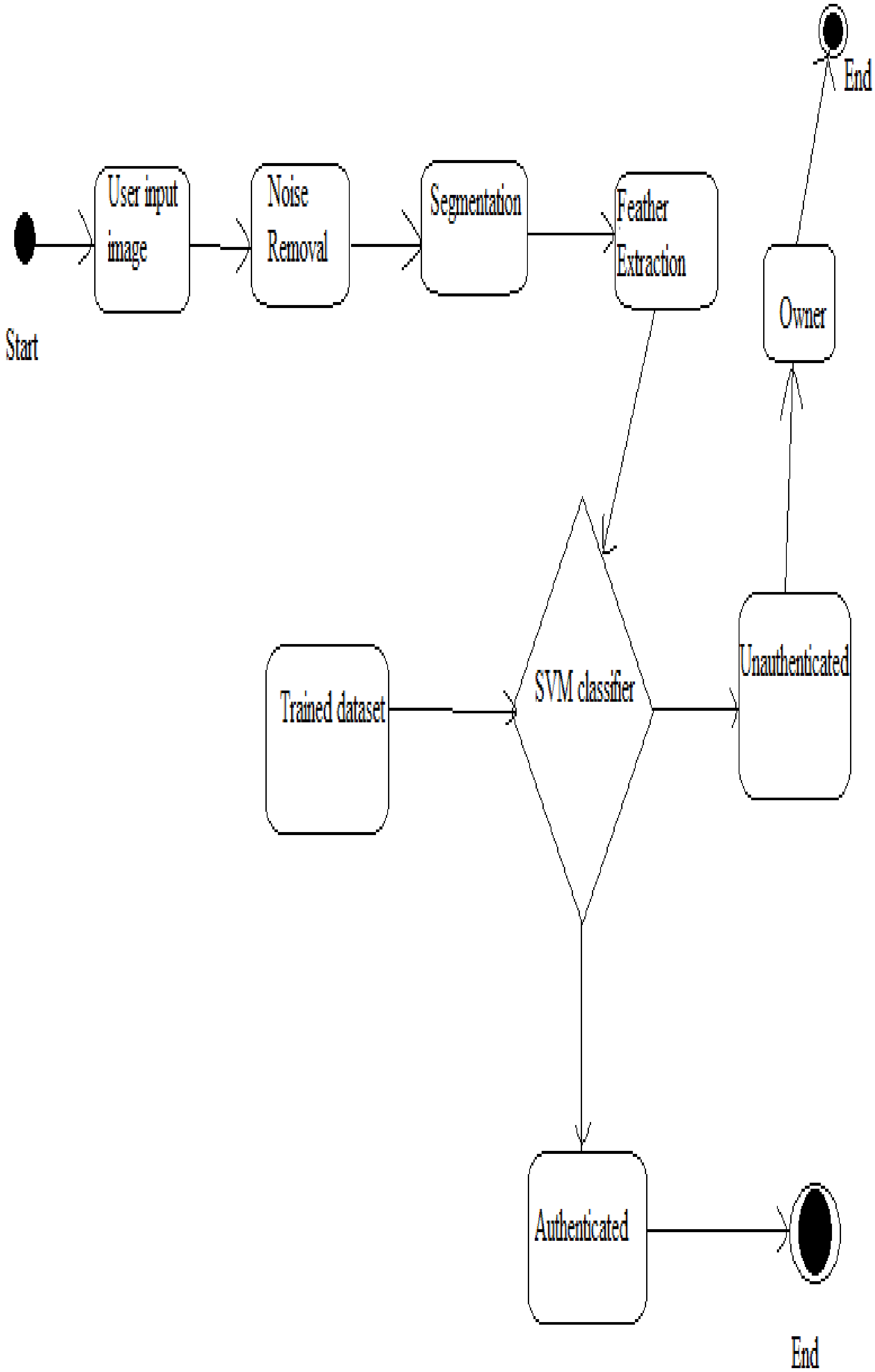


Figure 6.7: ACTIVITY DIAGRAM

CHAPTER 7

HARDWARE

7.1 OVERVIEW

In this section we describe the hardware components being used in the project. The results generated by MATLAB after performing feature extraction and training of the data set is that the authentic user would be authenticated and any other person trying to access will result in non-authentication. The result from MATLAB would be used by the Arduino controller to either run the motor or send an error message using the GSM component. Each of these components are explained below.

7.2 ARDUINO

Arduino basically is a board which can read inputs in various forms such as a light, sound or even a message and can turn those inputs into outputs. The Arduino Micro-controller is used to receive and convert the signals. The programming for this is done by the Arduino IDE which uses the programming language similar to C/C++, which is then compiled by a C-compiler figure 7.1 and 7.2.

In this project, the Arduino microcontroller will receive the input from MATLAB. This input will wither be 'authenticated' or 'non authenticated'. If the input received in authenticated, then the microcontroller will run the motor, which depicts the ignition starter of a car in this case. If the input is 'non authenticated', the microcontroller will then request the GSM component to send an error message in the mobile number of the registered user, so that he can know that someone is trying to access his vehicle.

Figure 7.1: Arduino Microcontroller

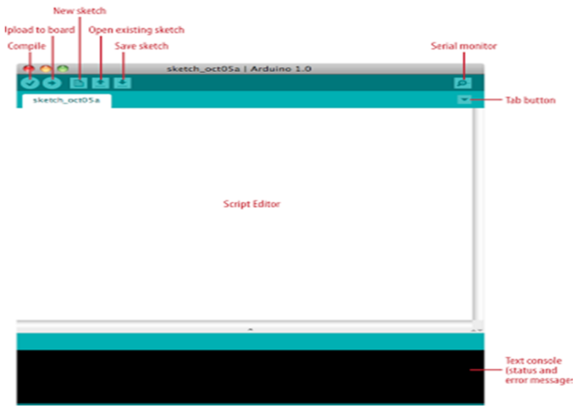


Figure 7.2: Arduino IDE

7.3 GSM COMPONENT

Global System for Mobile (GSM) stands for Global System for Mobile Communication. The GSM component is used to establish a connection between the system and a GSM device. In this case, the system will send an error warning to the user, of which the number would be pre-stored. The Arduino microcontroller, on receiving the 'non authenticated' signal, will send a request to the GSM controller to send the pre-defined error message using GSM technology. The message would help the user get intimated that false authentication is being tried. Thus, vehicle theft can be prevented figure 7.3.

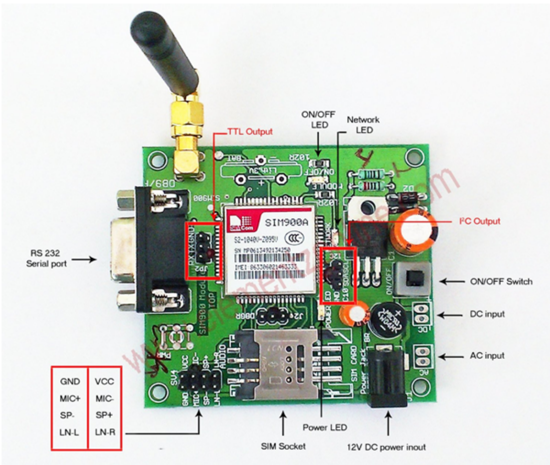


Figure 7.3: GSM Hardware component

CHAPTER 8

APPLICATIONS IN VEHICLE SECURITY AND FUTURE ENHANCEMENT

One of the application of using finger vein technology is to make an authentication system to provide vehicle security. The result from the finger-vein authentication can be used to decide whether the ignition of the vehicle should switch on or not. The database of the finger vein images of the people who are authorised to drive that vehicle can be recorded and kept. The finger-vein scanner will be attached to the microcontroller that will decide if the ignition can be switched on or not. On successful authentication of the person, a signal will be sent to the microcontroller, that will allow the ignition to switch on. If the authentication is not successful, a message can be sent to the real owner intimating about the unsuccessful attempt at authentication. This method can be used to prevent vehicle thefts and provide security to vehicles. The flowchart is shown in fig 8.1. As can be seen, the driver will have to input the finger vein image by getting it scanned by the finger vein scanner. The authentication will then be done. The results are sent to the Arduino controller, which according to the result, will allow the ignition to switch on. If the result of the match is negative, the controller will then be used to send a message to the owner, informing about the failed attempt.

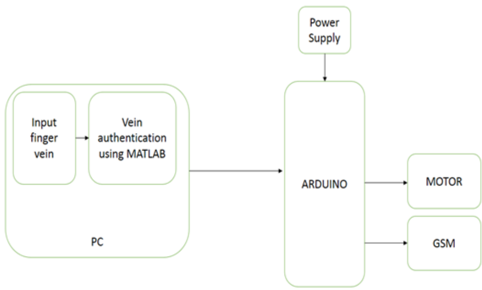


Figure 8.1: Flowchart for vehicle security

CHAPTER 9

RESULTS

The performance of the model is measured using two parameters- Mean square error Mean squared error (MSE) and peak signal to noise ratio (PSNR) [18-19]. Peak Signal to Noise Ratio **PSNR!** (**PSNR!**) and Mean Square Error (MSE) are used to compare the squared error between the original image and the new trained image. There is an inverse relationship between PSNR and MSE. Higher the value of PSNR, better is the quality of the image. For demonstration purpose as shown in table 9.1, we have tabulated the MSE and PSNR values for 6 of the finger vein images.

Table 9.1: Mean squared error and Peak signal to noise ratio for 6 finger-vein images

Image / Parameter	Mean Squared Error	Peak Signal to Noise
1	9.9691e-04	78.1443
2	4.1199e-04	81.9820
3	0.0013	77.0014
4	1.2716e-04	87.0874
5	0.0011	77.5450
6	0.0015	76.2956

CHAPTER 10

CONCLUSION AND FUTURE ENHANCEMENT

As discussed, we used finger-vein technology for authentication and described an application of it for vehicle security. The finger vein authentication is one of the best biometric methods that is available for use as it gives one of the highest accuracy amongst all the biometric methods. It can be executed easily and the whole authentication and verification process takes less than 0.5 seconds. In this project, we have been able to have the mean square error down to 0.0009059833 and the average peak signal to noise ratio as 79.67595.

10.1 FUTURE ENHANCEMENT

In the future work, our finger vein extraction and authentication for security purpose can be used in various aspects like ATM, House security, for attendance purpose in offices and college and government can use it as digital record of a person like aadhar card etc.

10.2 Sample Coding

```
function varargout = Finger_GUI(varargin)

gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @Finger_GUI_OpeningFcn, ...
                   'gui_OutputFcn',  @Finger_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

[FileName,PathName] = uigetfile('*.jpg;*.png;*.bmp','Pick an MRI Image');
if isequal(FileName,0)||isequal(PathName,0)
    warndlg('User Press Cancel');
else
    a = imread([PathName,FileName]);
    a = imresize(a,[200,200]);
   % input =imresize(a,[512 512]);

  axes(handles.axes1)
  imshow(a);title('Finger vein Image');



   I = double(a);
%figure, imshow(uint8(I));
% f = double(imread(file));
%f = double(imread(Path));
a = imresize(a,[256 256]);

%pixval on

sigmar = 40;              %sigmar: width of range Gaussian
eps    = 1e-3;            %eps: kernel accuracy

% compute Gaussian bilateral filter
sigmas = 3;               %sigmas: width of spatial Gaussian
[g,Ng] = GPA(I, sigmar, sigmas, eps, 'Gauss');



a1 = a(52:186,72:177);
```

```
%imshow(a1);
[r c] = size(a1);
a2 = imresize(a1,1/4);
%imshow(a2);

a2 = imresize(a2,[r c]);
%imshow(a2);

a3 = histeq(a2);


a4 = imresize(a3,1/3);
%imshow(a4);
[ll lh hl hh] =dwt2(a4,'haar');
feature1 = mean(mean(ll));
a5 = im2bw(a4);


 figure(2),
 subplot(2,1,1);
 imshow(a3);
 title('Histogram equalization');
 subplot(2,1,2)
 imshow(a5);
 title('Segmented image');


[row, col] = size(a);
mse = sum(sum((a(1,1) - g(1,1)).^2)) / (row * col);
psnr = 10 * log10(255 * 255 / mse);
disp('Mean Square Error ');
disp(mse);
disp('Peak Signal to Noise Ratio');
disp(psnr);


%ENHANCEMENT
%DILATION
se=strel('disk',7);
```

```
idilate = imdilate(I,se);

%HOG FEATURES
% [featureVector, hogVisualization] = extractHOGFeatures(I,[16 16]);


m=mean2(I);%mean
sd=std2(I);%std dev
en=entropy(I);%entropy
v=var(I(:));%variance
skw=skewness(I(:));%skewness
k=kurtosis(I(:));
feat=[m sd en  v skw k ];

disp('<--------------------Extracted Features------------------------>');
disp('Mean=');disp(m);
disp('Standard Deviation=');disp(sd);
disp('Entropy');disp(en);
disp('Variance');disp(v);
disp('Skewness');disp(skw);
disp('Kurtosis');disp(k);
disp('<------------------------------------------------------------->');

%  Main function for using GSA algorithm.
N=5;
 max_it=1000;
 ElitistCheck=1; Rpower=1;
 min_flag=1; % 1: minimization, 0: maximization

 F_index=1
 [Fbest,Lbest,BestChart,MeanChart]=GSA(F_index,N,max_it,ElitistCheck,min_flag,Rpow
%  figure;
%  semilogy(BestChart,'--k');
%
%  title(['\fontsize{12}\bf F',num2str(F_index)]);
%  xlabel('\fontsize{12}\bf Iteration');ylabel('\fontsize{12}\bf Best-so-far');
%  legend('\fontsize{10}\bf GSA',1);
```

```
load Trainset.mat
 xdata = meas;
 group = label;

 svmStruct1 = svmtrain(xdata,group,'kernel_function', 'linear');

 Result = svmclassify(svmStruct1,feat,'showplot',false)
  helpdlg(Result);

  s=serial('COM3', 'BaudRate', 9600);


  if strcmp(Result, 'Authenticated');
  fopen(s);
  pause(5);
  disp('B');
  fwrite(s,'B');
  pause(1);
  fclose(s);
  end
  if strcmp(Result, 'not Authenticated');
  fopen(s);
  pause(5);
  disp('A');
  fwrite(s,'A');
  pause(1);
  fclose(s);
  end
```

```
cc1=0.49;
cc2=0.30;
cc3=0.7;
cc4=0.97;
A=cat(3,a,a,a,a);
iteration=0;
while(iteration<5)
    c1=repmat(cc1,m,n);
```

```
c2=repmat(cc2,m,n);
c3=repmat(cc3,m,n);
c4=repmat(cc4,m,n);
c=cat(3,c1,c2,c3,c4);
distance=(A-c).^2;
d1=distance(:,:,1);
d2=distance(:,:,2);
d3=distance(:,:,3);
d4=distance(:,:,4);
M=cat(3,d1,d2,d3,d4);
y=zeros(m,n);
y=im2double(y);
for i=1:m
    for j= 1:n
        if min(M(i,j,:))==d1(i,j)
            y(i,j)=1;
        else if min(M(i,j,:))==d2(i,j)
            y(i,j)=2;
        else if min(M(i,j,:))==d3(i,j)
            y(i,j)=3;
        else if min(M(i,j,:))==d4(i,j)
            y(i,j)=4;
            end
            end
            end
        end
    end
end
out1=zeros(m,n);
out2=zeros(m,n);
out3=zeros(m,n);
out4=zeros(m,n);
for i=1:m
    for j= 1:n
        if y(i,j)==1
            out1(i,j)=a(i,j);
        else if y(i,j)==2
            out2(i,j)=a(i,j);
        else if y(i,j)==3
```

```
                 out3(i,j)=a(i,j);
             else if y(i,j)==4
                 out4(i,j)=a(i,j);
                 end
                 end
                 end
             end
         end
end
% Mean value for Iteration 1
count1=0;
sum1=0;
for i=1:m
    for j= 1:n
        if out1(i,j)~=0
            count1=count1+1;
            sum1=sum1+out1(i,j);
        end
    end
end
cc1=sum1/count1;
 % Mean value for Iteration 2
count2=0;
sum2=0;
for i=1:m
    for j= 1:n
        if out2(i,j)~=0
            count2=count2+1;
            sum2=sum2+out2(i,j);

           end
    end
end
cc2=sum2/count2;
 % Mean value for Iteration 3
count3=0;
sum3=0;
for i=1:m
    for j= 1:n
```

```
        if out3(i,j)~=0
            count3=count3+1;
            sum3=sum3+out3(i,j);
        end
    end
end
cc3=sum3/count3;
 % Mean value for Iteration 4
count4=0;
sum4=0;
for i=1:m
    for j= 1:n
        if out4(i,j)~=0
           count4=count4+1;
           sum4=sum4+out4(i,j);
        end
    end
end
cc4=sum4/count4;
iteration=iteration+1;
end

%%%OPTIC DISK EXTRACTION%%%
for i = 1:m
    for j=1:n
        if out3(i,j)>0
            out5(i,j)=1;
        else
            out5(i,j)=0;
        end
    end
end


for i = 1:m
    for j=1:n
        if out4(i,j)>0.2
            outb(i,j)=1;
        else
```

```
            outb(i,j)=0;
        end
    end
end

p1=imfill(outb,'holes');

se = strel('disk',2);
outd=imdilate(outb,se);
outd=bwareaopen(outd,500);
figure,imshow(outd);
title('Optic disk segmentation using k-means');
pause(2);
end
 %Bilateral Filtering

clc;
clear all;
close all;

%Get the image for filtering
[Path,U_C]=imgetfile;
f = double(imread(Path));

% filter parameters
sigmar = 40;              %sigmar: width of range Gaussian
eps    = 1e-3;            %eps: kernel accuracy

% compute Gaussian bilateral filter
sigmas = 3;               %sigmas: width of spatial Gaussian
[g,Ng] = GPA(f, sigmar, sigmas, eps, 'Gauss');

%Display the input and filtered image
colormap gray,
subplot(1,2,1), imshow(uint8(f)),
title('Input', 'FontSize', 10), axis('image', 'off');
subplot(1,2,2), imshow(uint8(g)),
title('Gaussian Bilateral Filter','FontSize', 10),axis('image', 'off');
[Path,U_C] = imgetfile;
```

```
    I = double(imread(Path));
figure, imshow(I);

%l=size(31);
feat=[m sd en  v skw k ];

load Trainset.mat
 xdata = meas;
 group = label;
 %svmStruct = svmtrain(xdata,group,'showplot',false);
 % Result = svmclassify(svmStruct,feat)
 svmStruct1 = svmtrain(xdata,group,'kernel_function', 'linear');

 Result = svmclassify(svmStruct1,feat,'showplot',false)
  helpdlg(Result);

  function Nest=estN(sigmar,T,eps)
% sigmar        : width of range Gaussian
% T             : dynamic range of image is [0,2T]
% eps           : kernel approximation accuracy
% Nest          : approximation order
if  sigmar > 70
    N=10;
elseif sigmar < 5
    N=800;
else
    lam=(T/sigmar)^2;
    p = log(exp(1)*lam);
    q = -lam - log(eps);
    t = q*exp(-1)/lam;
    W = t - t^2 + 1.5*t^3 - (8/3)*t^4;
    N = min(max(q/W,10),300);
    if sigmar < 30
        for iter = 1:5
            N = N - (N*log(N)-p*N-q)/(log(N)+1-p);
        end
    end
end
Nest = ceil(N);
```

```
end

clc;clear all;close all;
l = 'Authenticated';
n = 'not Authenticated';
srcFiles = dir('D:\Image processing\Finger_vein\Code\AAyush\dataset\*.jpg');
for i = 1 : length(srcFiles)
    filename = strcat('D:\Image processing\Finger_vein\Code\AAyush\dataset\',srcFi
    I = double(imread(filename));
figure, imshow(uint8(I));



se=strel('disk',7);
idilate = imdilate(I,se);



%

%HOG FEATURES
 [featureVector, hogVisualization] = extractHOGFeatures(I,[16 16]);
%      figure;
%     imshow(image); hold on;
%      plot(hogVisualization);
% title('HOG');

m(i,1)=mean2(I);%mean
sd(i,1)=std2(I);%std dev
en(i,1)=entropy(I);%entropy
v(i,1)=var(I(:));%variance
skw(i,1)=skewness(I(:));%skewness
k(i,1)=kurtosis(I(:));

meas=[m sd en  v skw k ];

label = {l;n;n};

end
```

```
save Trainset.mat meas label
%Bilateral Filtering

clc;
clear all;
close all;

%Get the image for filtering
[Path,U_C]=imgetfile;
f = double(imread(Path));

% filter parameters
sigmar = 40;            %sigmar: width of range Gaussian
eps    = 1e-3;          %eps: kernel accuracy

% compute Gaussian bilateral filter
sigmas = 3;             %sigmas: width of spatial Gaussian
[g,Ng] = GPA(f, sigmar, sigmas, eps, 'Gauss');

%Display the input and filtered image
colormap gray,
subplot(1,2,1), imshow(uint8(f)),
title('Input', 'FontSize', 10), axis('image', 'off');
subplot(1,2,2), imshow(uint8(g)),
title('Gaussian Bilateral Filter','FontSize', 10),axis('image', 'off');
if strcmp(flag,'Gauss')
    L=round(3*W);
    Hs=fspecial('gaussian',2*L+1,W);
elseif  strcmp(flag,'box')
    L=W;
    Hs=fspecial('average',2*L+1);
else
    error('not enough arguments');
end
T = 128;
N = estN(sigmar,T,eps);
f=padarray(f,[L,L]);
H=(f-T)/sigmar;
F=exp(-0.5*H.^2);
```

```
G=ones(size(H));
P=zeros(size(H));
Q=zeros(size(H));
Fbar=imfilter(F,Hs);
for n = 1 : N
    Q=Q+G.*Fbar;
    F=H.*F/sqrt(n);
    Fbar=imfilter(F,Hs);
    P=P+G.*Fbar*sqrt(n);
    G=H.*G/sqrt(n);
end
g= T+sigmar*(P(L+1:end-L,L+1:end-L) ...
    ./Q(L+1:end-L,L+1:end-L));
g(g<0)=0;
g(g>255)=255;
end

Fmax=max(fit); Fmin=min(fit); Fmean=mean(fit);
[i N]=size(fit);

if Fmax==Fmin
   M=ones(N,1);
else

   if min_flag==1 %for minimization
      best=Fmin;worst=Fmax; %eq.17-18.
   else %for maximization
      best=Fmax;worst=Fmin; %eq.19-20.
   end

   M=(fit-worst)./(best-worst); %eq.15,

end

M=M./sum(M); %eq. 16.

function fit=test_functions(L,F_index,dim)

%Insert your own objective function with a new F_index.
```

```
if F_index==1
fit=sum(L.^2);
end

if F_index==2
fit=sum(abs(L))+prod(abs(L));
end

if F_index==3
    fit=0;
    for i=1:dim
    fit=fit+sum(L(1:i))^2;
    end
end

if F_index==4
    fit=max(abs(L));
end

if F_index==5
    fit=sum(100*(L(2:dim)-(L(1:dim-1).^2)).^2+(L(1:dim-1)-1).^2);
end

if F_index==6
    fit=sum(floor((L+.5)).^2);
end

if F_index==7
    fit=sum([1:dim].*(L.^4))+rand;
end

if F_index==8
    fit=sum(-L.*sin(sqrt(abs(L))));
end

if F_index==9
    fit=sum(L.^2-10*cos(2*pi.*L))+10*dim;
end
```

```
if F_index==10
    fit=-20*exp(-.2*sqrt(sum(L.^2)/dim))-exp(sum(cos(2*pi.*L))/dim)+20+exp(1);
end

if F_index==11
    fit=sum(L.^2)/4000-prod(cos(L./sqrt([1:dim])))+1;
end

if F_index==12
    fit=(pi/dim)*(10*((sin(pi*(1+(L(1)+1)/4)))^2)+sum((((L(1:dim-1)+1)./4).^2).*..
        (1+10.*((sin(pi.*(1+(L(2:dim)+1)./4)))).^2))+((L(dim)+1)/4)^2)+sum(Ufun(L,
end
if F_index==13
    fit=.1*((sin(3*pi*L(1)))^2+sum((L(1:dim-1)-1).^2.*(1+(sin(3.*pi.*L(2:dim))).^2
        ((L(dim)-1)^2)*(1+(sin(2*pi*L(dim)))^2))+sum(Ufun(L,5,100,4));
end

if F_index==14
aS=[-32 -16 0 16 32 -32 -16 0 16 32 -32 -16 0 16 32 -32 -16 0 16 32 -32 -16 0 16 3
-32 -32 -32 -32 -32 -16 -16 -16 -16 -16 0 0 0 0 0 16 16 16 16 16 32 32 32 32 32];
    for j=1:25
        bS(j)=sum((L'-aS(:,j)).^6);
    end
    fit=(1/500+sum(1./([1:25]+bS))).^(-1);
end

if F_index==15
    aK=[.1957 .1947 .1735 .16 .0844 .0627 .0456 .0342 .0323 .0235 .0246];
    bK=[.25 .5 1 2 4 6 8 10 12 14 16];bK=1./bK;
    fit=sum((aK-((L(1).*(bK.^2+L(2).*bK))./(bK.^2+L(3).*bK+L(4)))).^2);
end

if F_index==16
    fit=4*(L(1)^2)-2.1*(L(1)^4)+(L(1)^6)/3+L(1)*L(2)-4*(L(2)^2)+4*(L(2)^4);
end

if F_index==17
    fit=(L(2)-(L(1)^2)*5.1/(4*(pi^2))+5/pi*L(1)-6)^2+10*(1-1/(8*pi))*cos(L(1))+10;
```

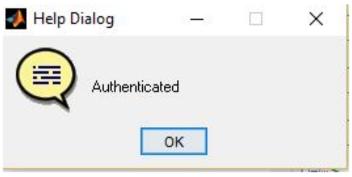
```
end

if F_index==18
    fit=(1+(L(1)+L(2)+1)^2*(19-14*L(1)+3*(L(1)^2)-14*L(2)+6*L(1)*L(2)+3*L(2)^2))*
        (30+(2*L(1)-3*L(2))^2*(18-32*L(1)+12*(L(1)^2)+48*L(2)-36*L(1)*L(2)+27*(L(2
end

if F_index==19
    aH=[3 10 30;.1 10 35;3 10 30;.1 10 35];cH=[1 1.2 3 3.2];
    pH=[.3689 .117 .2673;.4699 .4387 .747;.1091 .8732 .5547;.03815 .5743 .8828];
    fit=0;
    for i=1:4
    fit=fit-cH(i)*exp(-(sum(aH(i,:).*((L-pH(i,:)).^2))));
    end
end

if F_index==20
    aH=[10 3 17 3.5 1.7 8;.05 10 17 .1 8 14;3 3.5 1.7 10 17 8;17 8 .05 10 .1 14];
cH=[1 1.2 3 3.2];
pH=[.1312 .1696 .5569 .0124 .8283 .5886;.2329 .4135 .8307 .3736 .1004 .9991;...
.2348 .1415 .3522 .2883 .3047 .6650;.4047 .8828 .8732 .5743 .1091 .0381];
    fit=0;
    for i=1:4
    fit=fit-cH(i)*exp(-(sum(aH(i,:).*((L-pH(i,:)).^2))));
    end
end

aSH=[4 4 4 4;1 1 1 1;8 8 8 8;6 6 6 6;3 7 3 7;2 9 2 9;5 5 3 3;8 1 8 1;6 2 6 2;7 3.6
cSH=[.1 .2 .2 .4 .4 .6 .3 .7 .5 .5];

if F_index==21
    fit=0;
  for i=1:5
    fit=fit-((L-aSH(i,:))*(L-aSH(i,:))'+cSH(i))^(-1);
  end
end

if F_index==22
    fit=0;
```

```
  for i=1:7
    fit=fit-((L-aSH(i,:))*(L-aSH(i,:))'+cSH(i))^(-1);
  end
end

if F_index==23
    fit=0;
  for i=1:10
    fit=fit-((L-aSH(i,:))*(L-aSH(i,:))'+cSH(i))^(-1);
  end
end

function y=Ufun(x,a,k,m)
y=k.*((x-a).^m).*(x>a)+k.*((-x-a).^m).*(x<(-a));
return
```

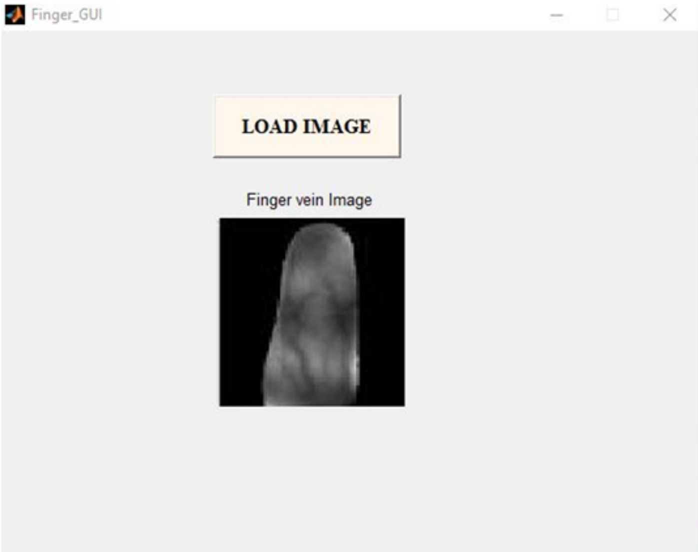
CHAPTER 11

SCREEN SHOTS

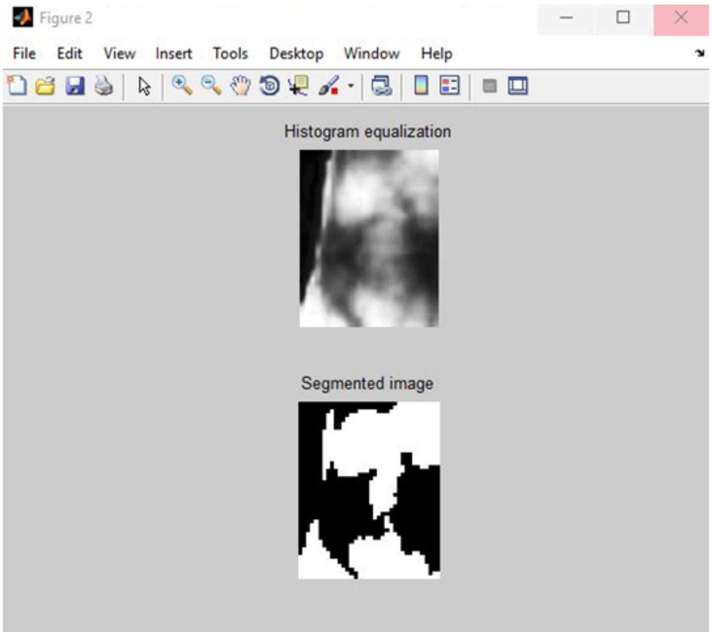
Multiple remote data collection points



Multiple remote data collection points



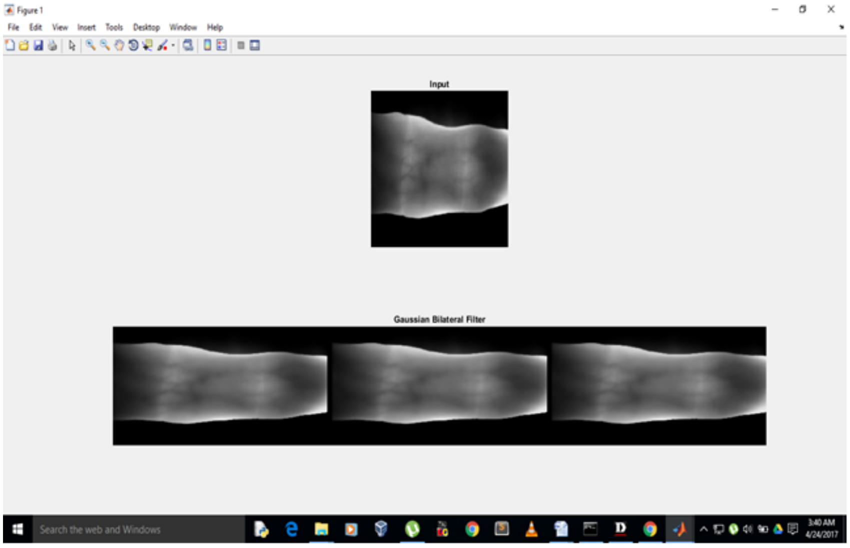
Multiple remote data collection points



Multiple remote data collection points

Multiple remote data collection points

Multiple remote data collection points



Multiple remote data collection points

4x6 double

	1	2	3	4	5	6	7	8	9
1		51.5461	0.8814	2.6570e+03	1.0408	3.6342			
2	76.5771	80.9443	0.8812	6.5520e+03	0.4411	1.7108			
3	49.4603	40.9469	0.0162	1.6766e+03	0.5567	2.6393			
4	59.2927	57.3347	0.9065	3.2873e+03	0.5513	2.3641			
5									
6									
7									
8									
9									
10									
11									
12									

Command Window

New to MATLAB? See resources for Getting Started.

CHAPTER 12

REFERENCES

1. Hashimoto J.,Finger Vein Authentication Technology and its Future, 2006 Symposium on VLSI Circuits, Digest of Technical Papers, pp. 5-8, 2006.

2. Lee H, S. H. Lee, T. Kim, and H. Bahn, Secure User Identification for Consumer Electronics Devices, IEEE Transactions on Consumer Electronics, Vol. 54, No. 4, pp. 1798-1802, Nov. 2008.

3. Desong Wang, Jianping Li, and Gokhan Memik, User Identification based on Finger-vein Patterns for Consumer Electronics Devices, IEEE Transactions on Consumer Electronics, Vol. 56, No. 2, pp. 799-804, May 2010.

4. V.R.Vijaykumar et al. Fast and Efficient Algorithm to Remove Gaussian Noise in Digital Images IAENG International Journal of Computer Science, 37:1, 1 Feb, 2010.

5. Wu J. D., and S. H. Ye Driver Identification using Finger-Vein Patterns with Random Transform and Neural Network, Expert System Applications, Vol. 36, pp. 5793-5799, 2009.

6. S. P. Lloyd, Least squares quantization in PCM , IEEE Trans. Inf.Theory, vol. IT-28, no. 2, pp. 129 âĂŞ 136, Mar.1982.

7. Ramakrishnan S. et al., SVD Based Digital Image Watermarking Using DWT, Proceedings of the National Conference on Intelligent Computing and Control Engineering Applications, pp. 83-86, Anna University of Technology, Coimbatore.

8. G. Xuan, Q. Yao, C. Yang, J. Gao, P. Chai, Y. Q. Shi, and Z. Ni. Lossless data hiding using histogram shifting method based on integer wavelets. 2006, Proc. Int. Workshop on Digital Watermarking, Vol. 4283, pp.323âĂŞ332.

9. Wen-Chang C, Ding-Mao J. Triaxial Accelerometer-Based Fall Detection Method Using a Self-Constructing Cascade-AdaBoost. Biomedical and Health Informatics, IEEE Journal. 2013; pages 411-419.

10. Anita Pati Mishra Noise Smoothing- Improving Image Filtering Methodology International Journal of Research Volume 03 Issue 09 May 2016.

11. Ms.Seema Rajput Prof.S.R.Suralkar Comparative Study of Image Enhancement Techniques IJCSMC, Vol. 2, Issue. 1, January 2013, pg.11 âĂŞ 21.

12. Rajesh Garg, Bhawna Mittal Histogram Equalization Techniques For Image Enhancement IJECT Vol. 2, Issue 1, March 2011.

13. Nameirakpam Dhanachandra et al. Image Segmentation Using K -means Clustering Algorithm and Subtractive Clustering Algorithm Procedia Computer Science Volume 54, 2015, Pages 764-771.

14. Tara Saikumar et al. Colour Based Image Segmentation Using Fuzzy C-Means Clustering 2011 International Conference on Computer and Software Modelling IPCSIT vol.14 (2011).

15. Ravindra S et al. Segmentation of Google Map Images Based on Color Features Proceedings of International Conference on Communication, Computation, Management and Nanotechnology (ICN-2011), September 23-25, 2011.

16. Murali Mohan. S et al. VLSI Architecture for Fast Computation of 2D-Discrete Wavelet Transform and Low Power Feed Forward Neural Network Architecture for Image Compression American Journal of Engineering Research (AJER) e-ISSN : 2320-0847 p-ISSN : 2320-0936 Volume-02, Issue-10, pp-136-145 www.ajer.org.

17. Kamrul Hasan Talukder, Koichi Harada HAAR Wavelet Based Approach for Image Compression and Quality Assessment of Compressed Image IAENG International Journal of Applied Mathematics, Volume 36, Issue 1, 2007.

18. Cui, L., and Allen, A.R. . An Image Quality Metric Based on a Colour Appearance Model. 2008 Proceedings of the 10th International Conference on Advanced Concepts for Intelligent Vision Systems, France, pp.696-707.

19. Wang, Z., Bovik, A.C., Sheikh, H.R., and Simoncelli, E.P (2004) Image quality Assessment: from error visibility to structural similarity. IEEE Tran. Image Processing 13(4), pp. 600-612.