



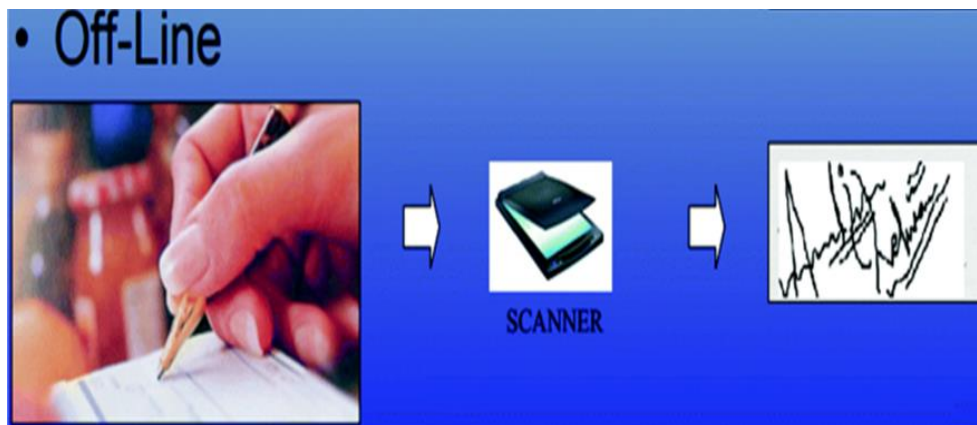
An-Najah National University
Faculty of Engineering and IT
Department of Telecommunication Engineering

Graduation Project Report 1

Offline Signature Recognition Using ANN

Submitted in partial fulfillment of the requirements for Bachelor
degree in Telecommunication Engineering

Supervisor: Dr.Khadija Mayyaleh



Proposed by

Ekram Othman [11613715]

Maram Bani jabber [11611935]

Academic year : 2020/2021

Acknowledgment

First of all, we would like to thank Allah who has given us the power to complete this project.

We would like to express our gratitude for everyone who helped us during the graduation project starting with endless thanks for our supervisor Dr. Khadija Mayyaleh who didn't keep any effort in encouraging us to do a great job, providing our group with valuable information and advices to be better each time. Thanks for the continuous support and kind communication which had a great effect regarding to feel interesting about what we are working on and for guide us in this project such as improve our searching capabilities and how to write the report.

We would like also to thank our parents for their support .Lastly, we offer our regards and blessing to all of those who inspired us during the completion of this project.

DISCLAIMER

This report was written by students at the Telecommunication Engineering Department, Faculty of Engineering, An-Najah National University. It has not been altered or corrected, other than editorial corrections, as a result of the assessment and it may contain language as well as content errors. The views expressed in it together with any outcomes and recommendations are solely those of the students. An-Najah National University accepts no responsibility or liability for the consequences of this report being used for a purpose other than the purpose for which it was commissioned.

APPENDIX

This report was written by Maram Bani Jaber and Ekram Othman in the Department of Telecommunications Engineering, College of Engineering, An-Najah National University. It has not been modified or corrected, as a result of the evaluation, and may contain errors in the language as well as the content. The opinions expressed in it together with any findings and recommendations are only those of Maram Bani Jaber and Ekram Othman. An-Najah National University assumes no responsibility or liability for the consequences of using this report for a purpose for which it was assigned.

Table of contents

ACKNOWLEDGMENT.....	I
DISCLAIMER.....	II
APPENDIX.....	III
LIST OF FIGURES.....	VIII
LIST OF TABLES.....	X
LIST OF ABBREVIATIONS.....	XI
ABSTRACT.....	XII

CHAPTER 1

Introduction	1
1.1 Objectives And Statement Of The..	1
1.2 The Importance Of The	1
1.3 The Scope And Limitation Of The Project	2
1.4 Organization Of The Report	2

CHAPTER 2

Constraints, Standards/ Codes and Earlier Coursework.....	3
2.1 Constraints	3
2.2 Standards/Codes.....	3
2.3 Earlier Coursework.....	3

CHAPTER 3

Literature Review.....	4
3.1 Introduction.....	4
3.1.2 Main Concepts.....	5

3.2 Image Pre-Processing.....	5
3.2.1 Alorithm For The Image Processing (Canny Edged Based Segmentation).....	6
3.3 Feature Extraction.....	6
3.3.1 Algorithm For Feature Extraction (The Pattern Averaging).....	6
3.4 Neural Network Training	6
3.4.1 Algorithm For Training Method (Back Propagation Algorithm)	7
3.5 Data Base	7

CHAPTER 4

Mythology.....	8
4.1 Introduction To ANN.....	8
4.2 Structure Of ANN.....	8
4.2.1 Layers.....	8
4.2.2 Weights.....	10
4.2.3 Activation Function Or Transfer Functions.....	10
4.2.4 Classification Of ANNs.....	10
4.2.5 Training Methods Of ANNs.....	11
4.3 The Developed Frame Work.....	11
4.3.1 Image Processing Phase.....	12
4.3.1.1 Introduction	12
4.4 Signature Image Processing	14
4.4.1 RGB To Grayscale Conversion	14
4.4.2 Image Smoothing Using Median Filtering	15
4.4.2.1 Matlab Code About Image Smoothing Using Median Filtering.....	16
4.4.2.2 Matlab Code About Remove Noise Using An Averaging Filter and A Median Filter And Compare Between Them	17
4.4.3 Adjustment Of Image Intensities	20

4.4.4 Thresholding	21
4.4.5 Canny Edge Based Segmentation.....	23
4.4.5.1 Edge Detection	23
4.4.6 Feature Extraction And Rescaling Using Pattern Averaging.....	30
4.5 Segmented Signatures.....	31
4.6 Network Training And Performance	32
4.6.1 Introduction.....	32
4.6.2 Backpropagation Neural Network Classification.....	33

CHAPTER 5

RESULTS DISCUSION AND COMPARISON	35
5.1 Results	35
5.2 Over Flowchart Diagrams and codes	36
5.2.1 Matlab code for Image Processing	37
5.3 Experimental Setup && Performance Evaluation	39
5.4 Result Comparison	40

CHAPTER 6

Discussion.....	41
6.1 Completing the proposed system	41
6.2 SWOT Analysis	41
6.2.1 Project Strength	41
6.2.2 Project Weaknesses	42
6.2.3 Project Opportunities	42
6.2.4 Project Threats	42

CHAPTER 7

Conclusion and Future Work.....	44
--	-----------

7.1 Conclusion	44
----------------------	----

7.2 Future Works	45
------------------------	----

CHAPTER 8

References	46
-------------------------	-----------

LIST OF FIGURES

Figure 3-1 : Neural Network Training	4
Figure 3-2:- Typical Handwritten Signature Authentication System	5
Figure 4-1: Layers structure in ANNs	9
Figure 4-2 : Basic structure of artificial neural network	10
Figure 4-3: The proposed framework	12
Figure 4-3.1 :- Gray Image of signature	12
Figure 4-3.2 :- Adjusted Image of signature	13
Figure 4-3.3 :- Thresholded Image of signature	13
Figure 4-3.4 :- Segmented Image of signature	13
Figure 4-3.5 :- Rescaled Image of signature	13
Figure 4-4: Matlab code for RGB_Conversion for a normal image.	14
Figure 4-5: RGB_Conversion for a signature image	14
Figure 4-6:-Median filter process	16
Figure 4-7:- The input signature image after applying median filter	16
Figure 4-8 : Matlab code for image smoothing by using median filter	16
Figure 4-9 :-Image smoothing using Median Filters	17
Figure 4-10:-Matlab code of Average Filter	18
Figure 4-10.1:-Average filter	18
Figure 4-11:-Matlab Code of Median Filtering	19
Figure 4-11.1:- Median Filtering	19
Figure 4-12:- Gamma correction (Gonzalez & Woods, 2001)	20
Figure 4-13:- Adjusted image intensities	21
Figure 4-14:- Thresholding of the adjusted image	22
Figure 4-15:- Hard activation function	23
Figure 4-16:- Segmented signature using Canny edge detection	24
Figure 4-17:- Edge based segmentation (Saif et al., 2012)	25
Figure 4-18: Dimension coordinate of Laplacian of Gaussian (LOG)	29
Figure 4-19: The results of differentiation of using the ramp edges	29

Figure 4-20:- Rescaled image using pattern averaging	30
Figure 4-21:- : Sample of the processed and segmented signature image using the proposed system	31
Figure 4-22:- Sample of the rescaled signatures processed using the proposed system	31
Figure 4-23 : Training and testing phases of the developed signature recognition system	33
Figure4 -24 :- Sample of training images	34
Figure 5.1 :-flow chart of signature recognition by canny edge detection	36
Figure 5.2 :-matlab code for image processing in a signature	38
5.3 :-result image processing code	38

LIST OF TABLES

Table4-1: Comparison Between Filters	20
Table 5-1:- The effect of number of nodes in the hidden layers and different features on the training time and recohntion rate	39
Table 5-2:- Results comparison with other works	40

LIST OF ABBREVIATIONS

ANNs: Artificial Neural Networks

BPNN: Back Propagation Neural Network

DSP: Digital Signal Processing

LOG: Laplacian of Gaussian

RGB: Red Green Blue Colours

SVM: Support Vector Machine

Abstract:

A person's signature is an important vital feature for a person that can be used to verify human identity, so we used the artificial neural network method to recognize the signature, and it consists of simple elements that work in parallel, these elements are inspired by the biological nervous system. The principle of its work is that the signature is captured and presented to the user in a form picture. Signature verification can be classified into online signature verification and offline signature verification. Online verification is based on dynamic capturing of signatures when they are made whereas Offline verification generally uses a scanned image of signatures. The objective of this project is to focus on the offline model of verification where several signatures are put through various processes before finally verifying it to be true or forged through Artificial Neural Networks (ANN). . To perform verification or identification of a signature, several steps must be performed. These steps * Image pre-processing * Feature extraction * Neural Network are:

Training

From many algorithms and methods with different accuracy percentages In this project we propose a human signature recognition system based canny edge detection and pattern averaging and back propagation neural network system..

CHAPTER ONE : INTRODUCTION

In this chapter we will discuss the problems that motivates us to think about this project, the objectives of our project , The importance of the Study(why we select this algorithm and methods??),the scope and Limitation of the project.

1.1_ Objectives and Statement of the problem

With the progress of new innovation, the technology security frameworks are by a great deal more propelled methods to identify a person. being supplanted biometrics, which include checking a person's These procedures are called unique finger impression, iris, voice, ,organic attributes, for example, face, retina signature and so forth. Formally, biometrics alludes to the ID of people by their attributes or traits. In this project we propose a human signature recognition system based canny edge detection and pattern averaging and backpropagation neural network system, that has the capability of determining the human signature images of different individuals handwritten signatures of presented with different scales, illuminations and different signature writing style of same signature image.

The progress of intelligent systems that uses neural systems is interesting and recently it has attracted more scientists into studying the potential uses of such systems in the signature recognition applications. The learning technique of a neural network is mimicked from the human brain one which relies on the features extracted from a seen image that retrieves the brain memory and generalizes the whole scene or image.

1.2_ The importance of the Study

Why we select this algorithms and methods?

A) This project proposes a simple, easy, and fast processing approach to extracting an average of useful features from a signature image using a technique called pattern averaging. This technique plays an important role in reducing the processing and training time and also in improving the recognition rate of the neural network. The experimental results show that the trained back propagation neural network is capable of Recognizing human handwritten

signatures regardless of scale, illumination, and difference is Writing style of the signatures.

B) canny edge detection and pattern averaging and backpropagation neural network have the capability of determining the human handwritten signatures of presented signature images of different individuals with different scales, illuminations and different Signature writing style of same signature image.

1.3) The scope and Limitation of the project

The system in this study is work on computer with matlab software.

1.4) Organization of the report:

This report consists of sequential parts to clarify the purpose of the project.

In chapter 1: Beginning clarifying the abstract and the introduction to our project.

In Chapter 2: we discussed the objective and statement of the project, and the importance of the project, and the scope and limitation of the project.

In chapter 3 we showed the Literature Review and quick background for each topic in this project in a general topics, in order to identify the reader to know what we are going to show.

In chapter 4 : *this is the main chapter in our project which we discussed the mythologies' and algorithms we used for each steps in our project ,and show the main body of the project .*

In chapter 5: *this chapter showed the results and comparison method , and the result we reached until now .*

In chapter 6: *this is the discussion chapter and SWOT analysis ,and show the project strength and weaknesses ,opportunities, and threats .*

In chapter 7 : *conclusion and future works*

In chapter 8 : *the references.*

Chapter 2: Constraints, Standards/ Codes, Earlier coursework

2.1 Constraints

The determinants of the project are to take into account work within security and to ensure a high accuracy rate consistent with the application in which proof of the correct signature of the forger will be employed, especially if these methods are used in the process of establishing crimes and police operations that require a high accuracy rate, or even in transactions Bank to prove the customer's personal information. Perhaps the second most important determinant is working in fast ways and does not take a long time for the proof process, so the time delay is supposed to be very little or non-existent in order to make the proof process faster and without time obstacles. In our first graduation project 1, work was done within the implementation boundaries of the part for codes using the Matlab program, which included the off line signature recognitions, provided that the part is executed Hardware part using online recognition in graduation project requirements 2.

2.2 Standards/ Codes

Because the project depends mainly on operations on the images process, we have chosen the Matlab program and approved the ANN available in it.

2.3 Earlier coursework

In our project, we have benefited from several university courses that we studied, especially those in which we learned image processing and the use of some filters. These materials are **signals and systems DSP and its laboratory, Multimedia, and Information and Partial Conventional Equation** .

CHAPTER 3: LITERATURE REVIEW

This chapter will contain some important concepts in our project, and the used algorithms and methods

3.1_INTRODUCTION

Artificial Neural Networks Artificial neural networks are a structure that has inspired its origins from the human thinking center or the brain. This structure has been inspired and developed to build a mechanism that can solve difficult problems in the science. Most of the structures of neural networks are similar to the biological brain in the need for training before being able to do a required task [1]. Similar to the principle of the human neuron, neural network computes the sum of all its inputs.

In recent years, there has been a significant increase in demand for services and systems based on checks and human signatures. This has led to an increase in fraud, forgery, signature counterfeiting, and impersonation. With the development of intelligent systems, an artificial neural network, inspired by the biological nervous system, was created to verify the signature of each person by storing the signature form in the database. Compare the sample signature on the check.

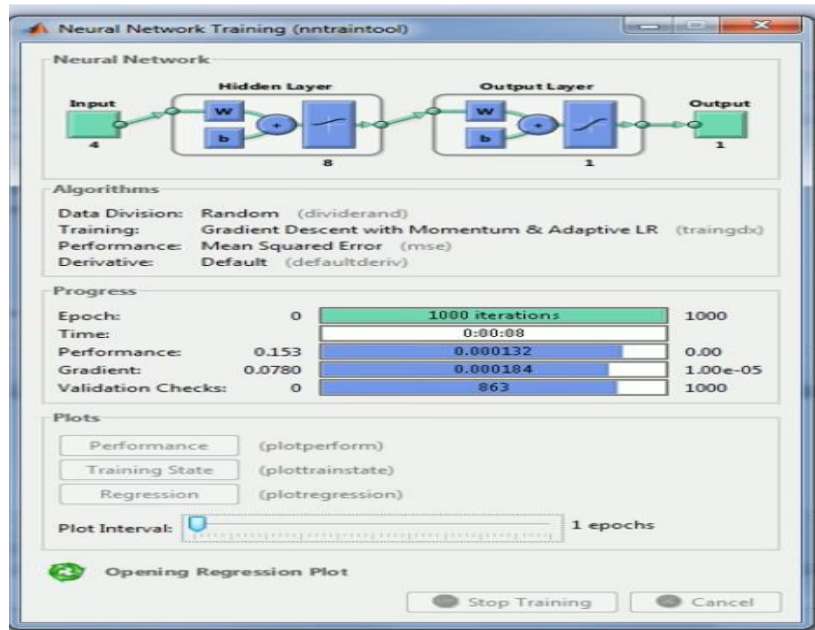


Figure 3-1 :- Neural Network Training

3.1.2 Main Concepts

The developed framework consists of two **main phases** which are the **processing phase** and the **classification phase** in which the image is classified as many signatures .

To perform verification or identification of a signature several steps must be performed.[2]

These steps are

- A. Image pre-processing
- B. Feature extraction
- C. Neural network training

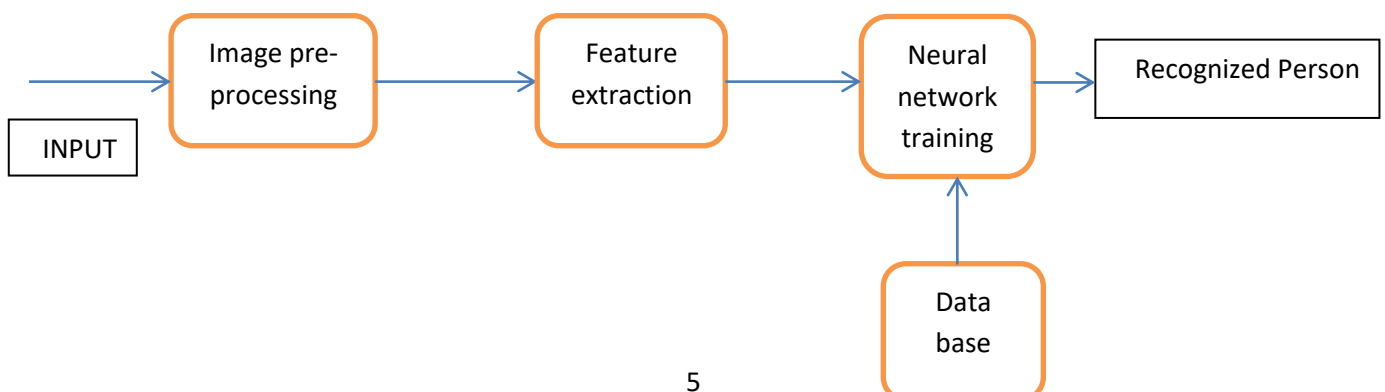


Figure 3-2:- Typical Handwritten Signature Authentication System

3.2_Image Pre-Processing

Image pre-processing represents a wide range of techniques that exist for the manipulation and modification of images. It is the first step in signature verification and recognition. A successful implementation of this step produces improved results and higher accuracy rates. In the image processing phase the signatures are processed using many techniques such as **conversion to grayscale**, filtering using **median filter**, and **segmentation using canny edge detection**. These techniques are done in order to enhance the quality of images and to extract the important features in such a way to take only the signature and ignoring the other features and parts of the image.

3.2.1_ Algorithm for the image processing (Canny Edged Based Segmentation)

➡ In this project, we used high pass and low pass image filtering to extract the features. Canny edge detection is the most popular method for edge detection and is a form of highpass filter. On the other hand, the averaging filter is a form of lowpass filter which will blur the image, Canny edge detector produced higher recognition rate as it captures the unique characteristics of the signature.

➡ Segmentation can be defined as grouping of the image part into many regions. The goal of such image processing operation is to represent some meaningful and needed areas of the image.[2]

3.3 Feature Extraction

The last phase here is the feature extraction phase in which the pattern averaging technique is used. This technique is a size reduction technique using averaging of the image pixels. Hence, averaging of the image pixels reduces the size of the image by considering only the important features or pixels of the

image which facilitates the training phase and reduces the number of input neurons of the network

3.3.1_ algorithm for Feature Extraction (the Pattern Averaging)

After the segmentation process using the canny edge detection, the images size should be reduced in order to be fed to the neural network. To reduce the size of images while keeping the useful and needed features extracted by the previously used methods, we used patten averaging.

3.4 Neural Network Training

Neural networks - like human beings - depend on the idea of learning in order to achieve any task. They learn through training on a large number of data, which enables them to create a pattern with time, that they will use later. They are very helpful in detecting patterns that are complicated and

hard to derive by humans or by simple techniques. Just like the case of signature recognition, it is very hard to tell whether a signature is original or forged, especially if it is carried out by a skilled forger. Thus a more advanced technique to detect the differences is needed to achieve a decision on its authenticity. Neural networks do not follow a set of instructions, provided for them by the author, but they learn as they go case by case.

3.4.1_Algorithm for traning method (Back propagation algorithm)

The back propagation training algorithm uses a feed forward process, a back propagation updating method, and supervised learning topology. This algorithm was the reason of neural networks development in the 80s of the last century. Back propagation is a general purpose learning algorithm. Although it is very efficient, it is costly in terms of processing requirements for learning. A back propagation network with a given hidden layer of elements can simulate any function to any degree of accuracy.

3.5_data base

Neural networks are highly reliable when trained using alarge amount of data base. They are used in applications where security is highly valued, In our project, using the canny edge detection metod, we were able to obtain different resolutions depending on the size of the signature images in the input data base

and thus These images are stored in a database which we are going to use in training & testing of ANN.

CHAPTER 4: MYTHOLOGY

The most important thing we want to discuss in this chapter is the main components used in our project, and the basic mythologies.

4.1_(Introduction to AAN):

Artificial neural networks (ANNs) are the simple simulation of the structure and the function of the biological brain. The complex and accurate structure of the brain makes it able to do hard different simultaneous tasks using a very huge (number of biological neurons connected together in grids. [3].

One of the most common methods of teaching ANNs to perform given tasks is the back propagation algorithm. It is based on a multi-stage dynamic system optimization method proposed by Arthur E. Bryson and Yu-Chi Ho in 1969 (Ho, 1969).

4.2_(Structure of ANN) :

The structure of ANNs consists mainly of three aspects in addition to the learning method. These aspects are the layers, weights, and activation functions. Each one of these three parts play a very important rule in the function of the ANN. The

learning function is the algorithm that relates these three parts together and ensures the correct function of the network.

4.2.1 (Layers):

ANN is constructed by creating connections between different layers to each other. Information is being passed between the layers through the synaptic weights. In a standard structure of ANN there are three different types of layers

1] Input layer: The input layer is the first one in a neural network. Its rule is the transmission of input information to the other layers. An input layer doesn't process the information; it can be considered as the sensors in biological system. It can also be called non processing layers

2] Output layer: The last layer in the neural network whose output is the • output of the whole network. In contrary to the input layer, the output layer is a .processing layer

3] Hidden layers: This is the main part of the network. It consists of one or • more of processing layers. They are connecting the input layers to the output layers. layers are the main processing layers where the weights are being Each one of the hidden layers connects between two .updated continuously hidden layers or one hidden and input or output layer.

Fig(4-1) presents the layers of the neural network and the connections between the layers. As shown in the figure, the inputs are fed to the input layer. The output of the input layer is fed to the hidden layers. The output obtained from the hidden layers is fed to the output layer that generates the output of the network.

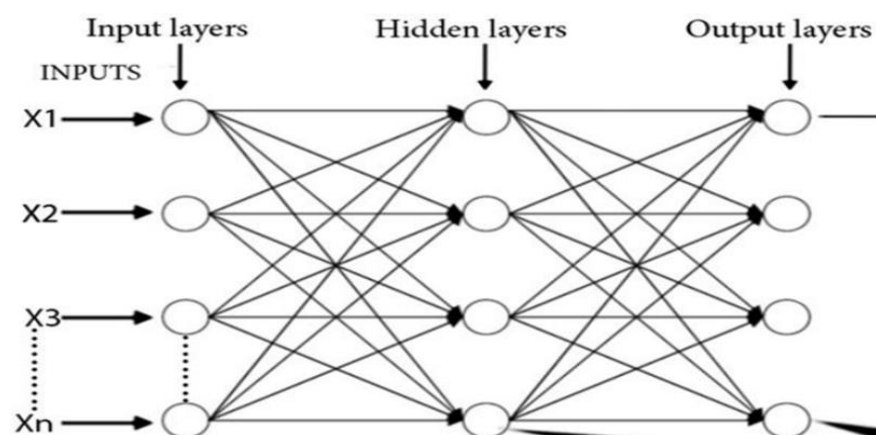


Figure 4-1: Layers structure in ANNs

The output function is the output of the neuron in this structure. The input of the activation function is given by:

$$TP = \sum x_n \omega_n$$

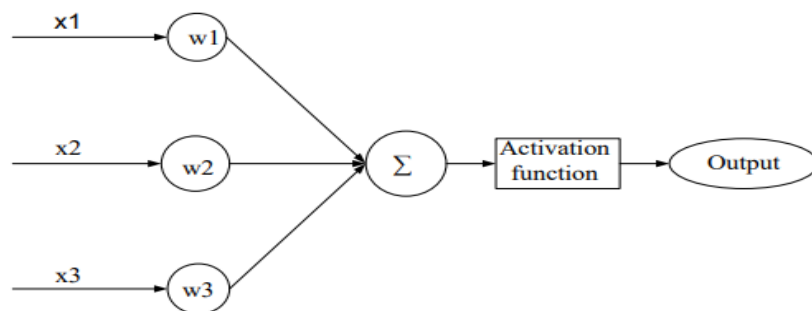


Figure 4-2 : Basic structure of artificial neural network

4.2.2 (Weights)

The weights in an ANN represent the memory of that network in which all information is stocked. The values of the weights are updated continuously during the training of the network until the desired output is reached. The memory or weights are then stored to be used in future. After learning the values of these weights are used as the memory of network [5].

4.2.3_ (Activation functions or transfer functions)

When the inputs are fed to the layers through the associated weights and finding the sum of them, an activation or transfer function is used to determine whether the output is to be activated or not. Or in some activation functions, the function is used to determine how much the processed input will share in constructing the total output of the network. Activation functions are very important in neural networks because they can decide whether the input to the neuron is enough to be passed to the next layer or not. There are many types of activation functions in artificial neural networks

- 1]_ linear activation functions or ramp
- 2]_ Threshold function (Hard activation function

3] Sigmoid function

4.2.4_Classification of ANNs ANNs

can be classified based on different aspects; these are the flow of information, function or task, and the training method. The flow of information can be either from input layer toward hidden and output layers. It can also flow from next layer to the previous layer. According the function, neural networks are used to accomplish many different tasks. These tasks can be categorized into four main categories

- A]** .Classification: Where an object is assigned to a group of known categories
- B]** .Association: Linking objects to more précised categories
- C]** Optimization: Where the task is to find the best solution for a case or problem

4.2.5) Training methods of ANNs

Generally, the training of a network is an attempt to lead the network to converge toward desired output or outputs. two main learning methods are used in teaching the networks. These are the supervised and the unsupervised

A] Supervised learning: The ANN is provided by input data and .learning method desired target for this data. The network then updates its weights according to a defined algorithm rule until it converges to a minimum error or reaches a maximum number of iterations. A very important example of the supervised .learning method is the error **back propagation method**.

B] Unsupervised learning: In this method, the input data is provided to the network which in turn modifies its weights according to defined conditions



4.3 The developed frame work

As we mention in chapter 3 , The developed framework consists of two main phases

A] which are the processing phase and

B] the classification phase in which the image is classified as many signatures To perform verification or identification of a signature several steps must be performed.

These steps are

A. Image pre-processing

B. Feature extraction

C. Neural Hnetwork training

4.3.1 IMAGE PROCESSING PHAES

4.3.1.1_Introduction

Image analysis involves the conversion of features and objects in image data into quantitative

information about these measured features and attributes. The proposed

signature recognition system is based on the processing of the signature images in order to extract the only needed features that can distinguish human signatures so that it can be easy for the neural network to

classify them into different signatures. Such image processing involves many techniques. The images are first **converted to grayscale** since they were acquired **using a camera** (Figure 4-3.1). **The noise is later on reduced using median filter**; a non-linear filter used to remove noise with preserving the edges of the image. The intensities of the pixels of the image are then mapped into a new range of intensities which results in a better image in which the signature is being darker (Figure 4-3.2) however; the other part is getting white. **After increasing the intensities of the**

pixels in the image; **Thresholding** takes place. This is a technique is a non-linear operation that converts a gray-scale image into a binary image where the two levels are assigned to pixels that are below or above a certain threshold (Figure 4-3.3)By applying this technique the image edges get smoothed and clear, therefore, this makes it easy to be detected. For the detection of edges

Canny operators are used (Figure “ 4-3.4)

Finally the image is rescaled **with preserving the important features using pattern averaging technique** (Figure 4-3.5)



Figure 4-3.1 :- Gray Image of signature



Figure 4-3.2 :- Adjusted Image of Signature



Figure 4-3.3 :- Tresholed image of Signature

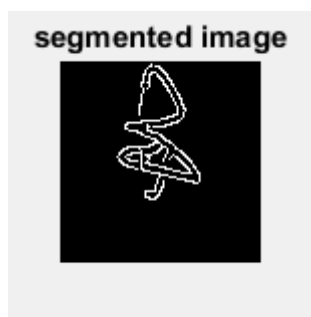


Figure 4-3.4.:- Segmented image of Signature

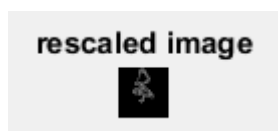


Figure 4-3.5 :- Rescaled Image of Signature

4.4 : Signature Images Processing

4.4.1_RGB to grayscale conversion:

The images were first converted from RGB to grayscale in which this conversion is done using the luminosity method (Figure 27). This method is a more sophisticated version of the average method. It also averages the values of the image matrix, but it forms a weighted average to account for human perception since humans are more sensitive to green than other colors, therefore; green is weighted most heavily. The formula for luminosity is

$$0.21R + 0.72 G + 0.07 B$$

Matlab code

```
1 - RGB = imread('peppers.png');  
2 - imshow(RGB)  
3 - I = rgb2gray(RGB);  
4 - figure  
5 - imshow(I)
```

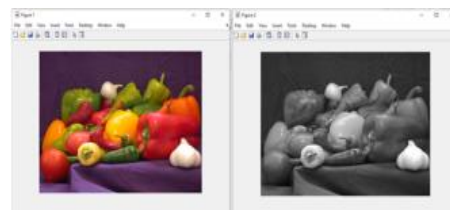


Figure 4-4: Matlab code for RGB_Conversion for a normal image.



Figure 4-5: RGB_Conversion for a signature image

which relies on the contribution of each color of the three RGB colors. Using this method, the grayscale image is brighter since the colors are weighted according to their contribution in the RGB image not averagely (6). Figure 4-5 illustrates the conversion of a signature image from our database into a grayscale image .using luminosity method

4.4.2 :Image smoothing using median filtering

also called (Noise reduction or “noise filtering”) using median filtering(non linear filter)

Smoothing, so called blurring, is an image processing technique used in order to reduce the noise in an image to produce less pixelated and clearer image. Most smoothing techniques are based on low pass non-linear filters. It is mostly based on the averaging technique of the input image or the middle (median) value technique [6]

To perform a smoothing operation we will apply a filter to our image. The most common type of filters is the linear filters such as median filter which is used in our proposed system. This filter is used to reduce impulsive noise or the salt-and pepper in an image with preserving the useful features and image edges. Median filtering is a non-linear process in which the output of the being processed pixel is found by calculating the median of a window of pixels that surrounds that studied pixel. In other words, the median filter goes through each element of the image and replaces each pixel with the median of its neighboring pixels which are located in a square neighborhood (kernel) around the evaluated pixel.

1] Fetching entries from other places in the signal. With images for example, entries from the far horizontal or vertical boundary might be selected

2] Shrinking the window near the boundaries, so that every window is full.

Figure 4-6 illustrates an example of a median filter and its mechanism to reduce the noise in an image by setting a kernel or window that goes through the whole matrix and find an output for the processed pixel by calculate the median of the pixels in the window [6].

Figure 4-6 shows the input image after applying the median filter.

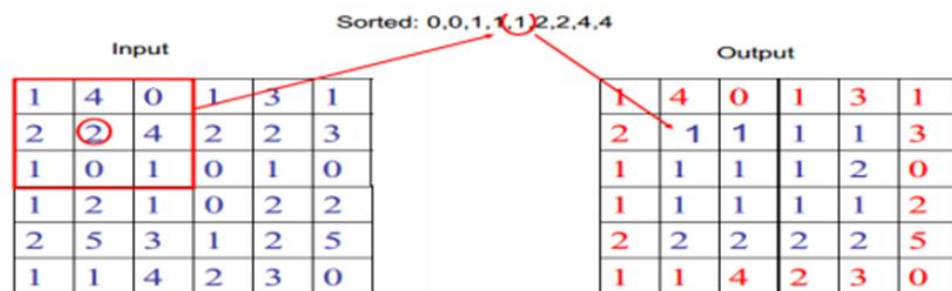


Figure 4-6:-Median filter procrss

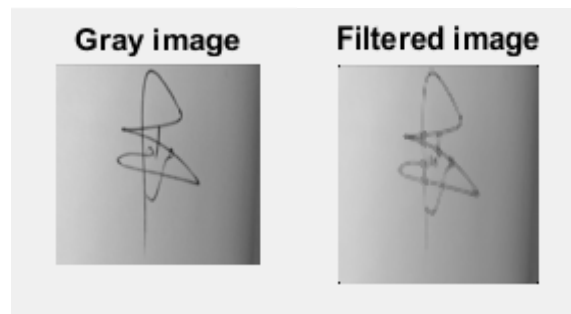


Figure 4-7:- The input signature image after applying median filter

4.4.2.1 Matlab code about Image smoothing using Median filter :[7]

Matlab Code

```

1  %%Generate a sinusoidal signal sampled for 1 second at 100 Hz.
2  %%Add a higher-frequency sinusoid to simulate noise
3  - fs = 100;
4  - t = 0:1/fs:1;
5  - x = sin(2*pi*t*3)+0.25*sin(2*pi*t*40);
6  %%Use a 10th-order median filter to smooth the signal. Plot the result.
7  - y = medfilt1(x,10);
8  - plot(t,x,t,y)
9  - legend('Original','Filtered')
10 - legend('boxoff')

```

Figure 4-8 : Matlab code for image smoothing by using median filters

Code Output

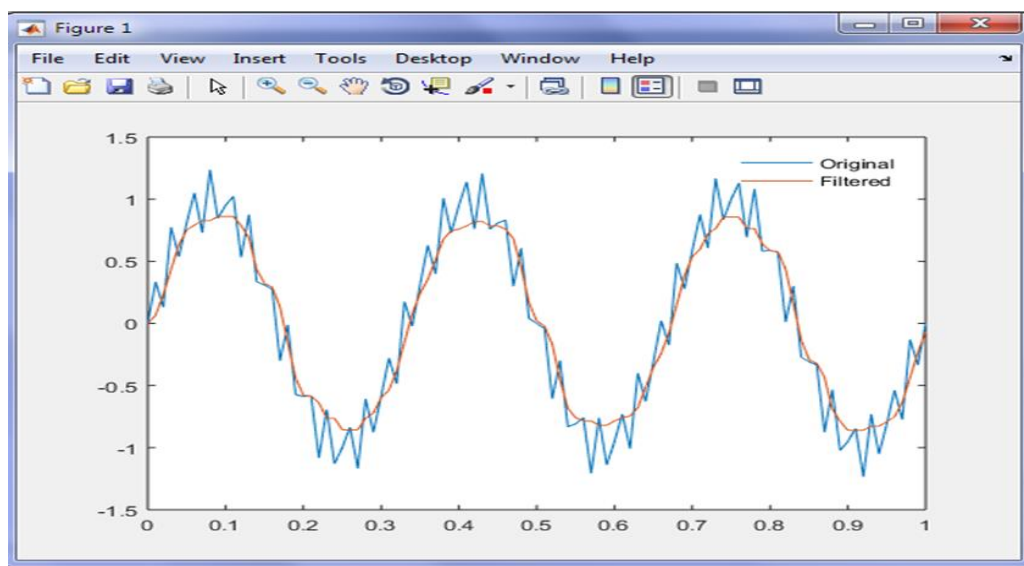


Figure 4-9 :-Image smoothing using Median Filters

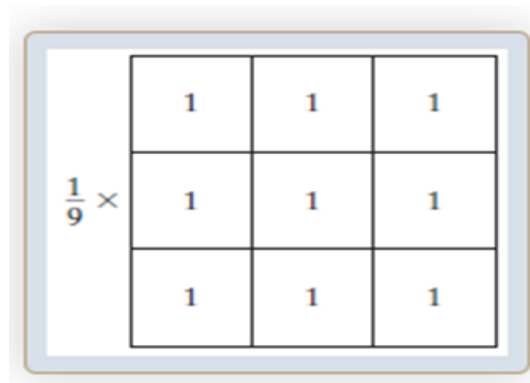
4.4.2.2 : Matlab code about (Remove Noise Using an Averaging Filter and a Median Filter& compare between them

I. Average Filtering(Smoothing Linear Filters)

The output of a smoothing, linear spatial filter is simply the average of the pixels contained in the neighborhood of the filter mask. These filters sometimes are called averaging filters. For reasons explained in they also are referred to a low pass filters. The idea behind smoothing filters is straightforward. By replacing

the value of every pixel in an image by the average of the gray levels in the neighborhood defined by the filter mask, this process results in an image with reduced “sharp” transitions in gray levels. Because random noise typically consists of sharp transitions in gray levels, the most obvious application of smoothing is noise reduction.

Average Filter mask is as follows:



$$\frac{1}{9} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Matlab code:

```

1      % Read Image for Noise Addition
2 -    img=imread('lena.bmp');
3      % Add Noise
4 -    Noi_img = imnoise(img,'salt & pepper', 0.02);
5      % Mask Definition
6 -    f=1/9*[1,1,1;1,1,1;1,1,1];
7      % Apply filter2 function
8 -    de_noi=filter2(f,Noi_img);
9 -    figure;
10 -    subplot(1,3,1);imshow(img);title('Original image')
11 -    subplot(1,3,2);imshow(Noi_img);title('Noisy image')
12 -    subplot(1,3,3);imshow(uint8(de_noi));title('Denoised image')|

```

Figure 4-10:-Matlab code of Average Filter

Code output

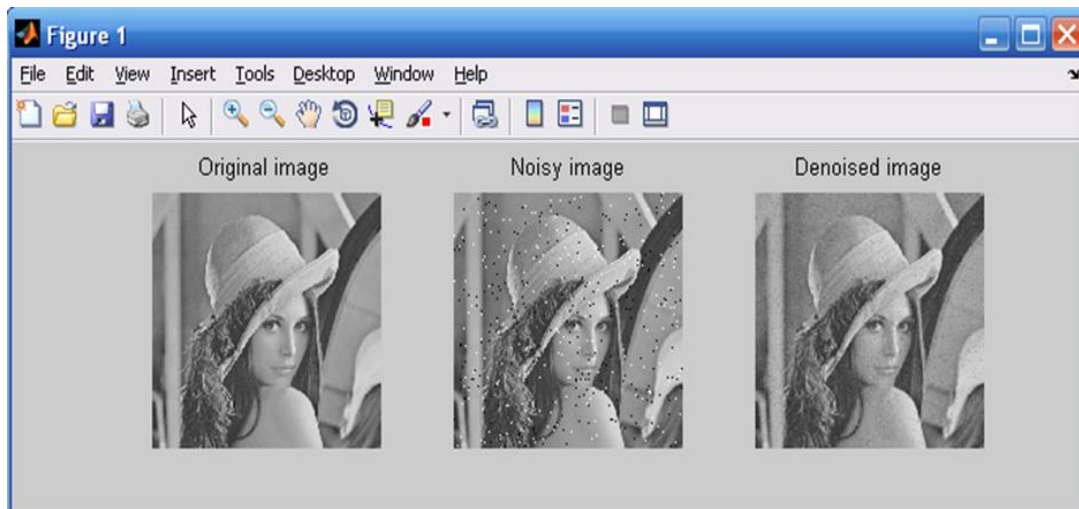


Figure 4-10.1:-Average filter

2-Median Filtering(Smoothing Non-Linear Filter)

The best-known example in this category is the median filter, which, as its name implies, replaces the value of a pixel by the median of the gray levels in the neighborhood of that pixel (the original value of the pixel is included in the computation of the median). Median filters are quite popular because, for certain types of random noise, they provide excellent noise-reduction capabilities, with considerably less blurring than linear smoothing filters of similar size. Median filters are particularly effective in the presence of impulse noise, also called salt-and-pepper noise because of its appearance as white and black dots superimposed on an image.

Matlab code:

```
1 % Read Image for Noise Addition
2 - img=imread('lena.bmp');
3 % Add Noise
4 - Noi_img = imnoise(img,'salt & pepper', 0.02);
5 % Apply medfilt2 function
6 - de_noi=medfilt2(Noi_img,[3 3]);
7 - figure;
8 - subplot(1,3,1);imshow(img);title('Original image')
9 - subplot(1,3,2);imshow(Noi_img);title('Noisy image')
10 - subplot(1,3,3);imshow(uint8(de_noi));title('Denoised image')|
```

Figure 4-11:-Matlab Code of Median Filtering

Code Output:

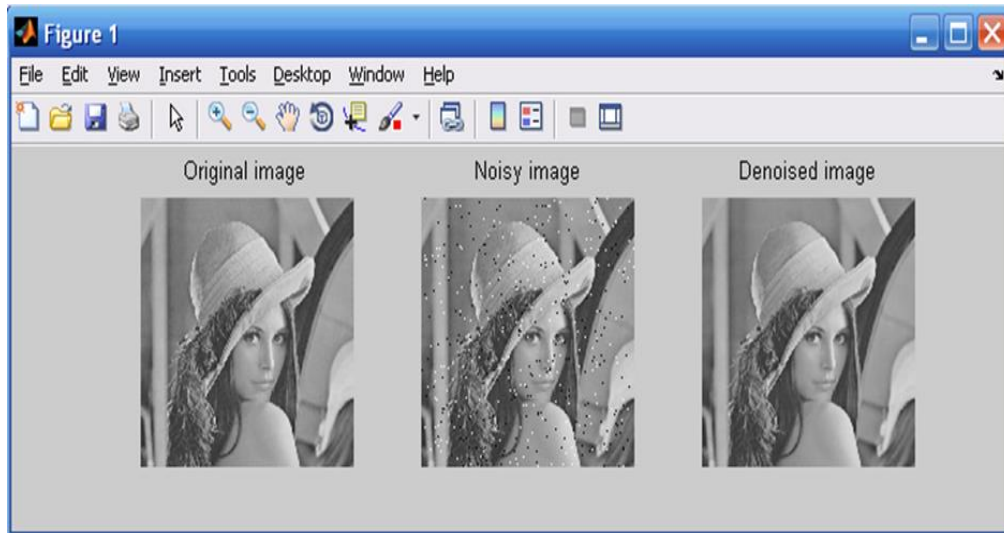


Figure 4-11.1:- Median Filtering

****Comparison Between Filters : [8]**

Parameters	Average Filter	Median Filter
Noise Reduction	Reduces Noise but it introduces blurring effect at edges.	Blurring effect is less as compared with Average filter
Percentage of noise Reduction	100% noise Not Reduced	Almost 100% noise reduced.
Size of Filter	As we increase the size of the filter mask, Noise reduces but blurring effect increases.	As we increase the size of the filter mask, 100% of Noise reduces but blurring effect at edges increases.
Mask	1/9x[1,1,1,1,1,1,1,1,1]	Pixel value is replaced by median value of <u>neighborhood</u> .
MATLAB Function	Filter2(<u>mask</u> ,Noisy_img)	medfilt2(<u>Noisy_img</u> ,[3 3])

Table4-1: Comparison Between Filters

4.4.3 _Adjustment of image intensities

For the purpose of increasing the image intensity and enhance its quality, the images undergo intensity adjustment. This image processing technique that aims to enhance the contrast of the image by increasing the intensity of its pixels. During this operation, the intensity value of each pixel in the input image is transformed using a transfer function to form a contrast-adjusted image. Gamma

contrast adjustment is the most common used transfer function (Gonzalez (&Woods, 2004).

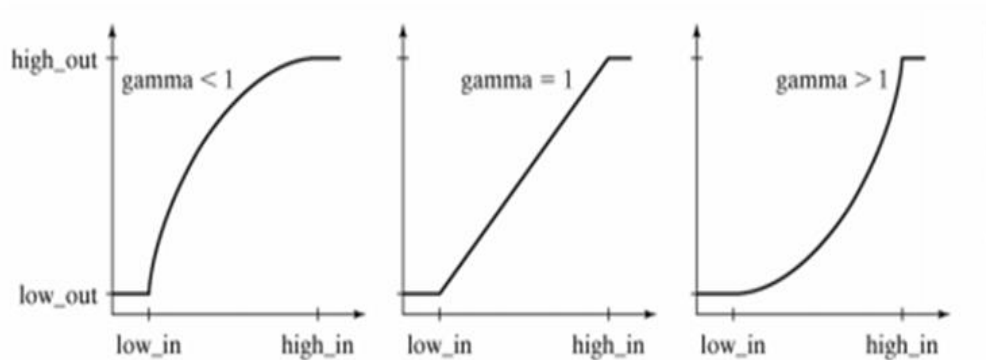


Figure 4-12:- Gamma correction (Gonzalez & Woods, 2001)

Ankit Aggarwal, R.S. Chauhan and Kamaljeet Kaur developed a system for the adaptive image enhancement technique preserving brightness level using gamma correction. Their proposed technique is that the weighted average of the histogram leveled, gamma corrected and the first picture are consolidated to acquire the upgraded processed image. The proposed calculation accomplish contrast enhancement as well as preserve the brightness level of images (Ankit Aggarwal et al., 2013). The figure below represents the adjustment of an image and its effects in enhancing the image contrast

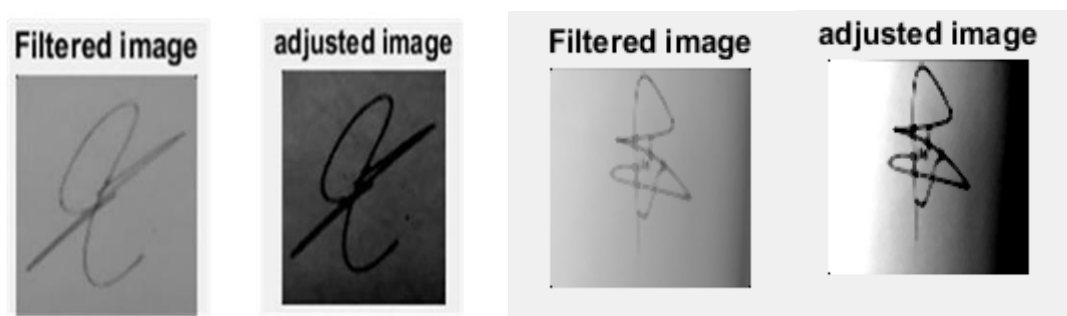


Figure 4-13:- Adjusted image intensities

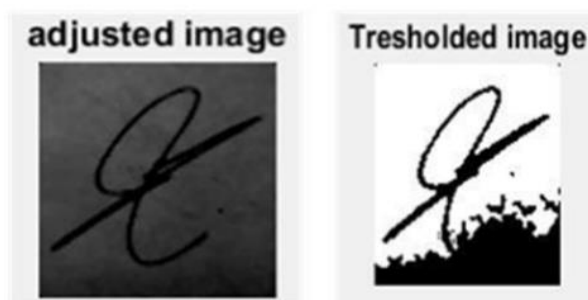
It can be seen that the signature is clearer and getting extracted after applying this technique. We can notice from the above image that the image adjustment operation has a great effect in enhancing the contrast and brightness of the image, so it is clearer and its features bright and shown. This helps in detecting the edges and are more features of the image in the next process.

4.4.4_(Thresholding (Hard activation function)):

Thresholding is the separation of region of images into two regions. One region corresponds to the foreground region, in which it contains the objects that we are interested in. The other region is the background, corresponds to the unneeded objects. This provides segmentation of the image based on the image different intensities and intensity discontinuities in the foreground and background regions. The input of this method is usually a grayscale or color image, while the output is a binary image representing the segmentation. The black pixels refer to background and white pixels refer to foreground. The segmentation is achieved by a single parameter known as

the intensity threshold. This is set by analyzing the histogram of the image which represents the intensity distributions of the image. During Thresholding, each pixel is compared to that

threshold value. If the pixel value is greater than that threshold, then this pixel is considered as foreground pixel (white). If the pixel value is lower than that threshold value, then the pixel is considered as background pixel (black) (Gonzalez & Woods, 2002). Figure 4-14 illustrates a signature image that undergoes thresholding of 0.42 as threshold value



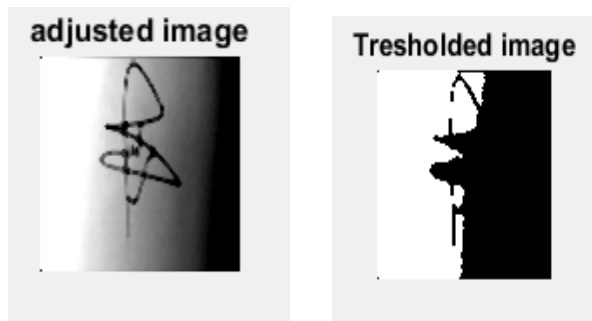


Figure 4-14:- Thresholding of the adjusted image

In the threshold function the output is zero if the summed input is less than certain value of threshold, and 1 if the summed input is greater than threshold. This way the output is oscillating between two values (Yuhong & Weihua, 2010). It can be either activated or deactivated. The function of the hard function is defined by:

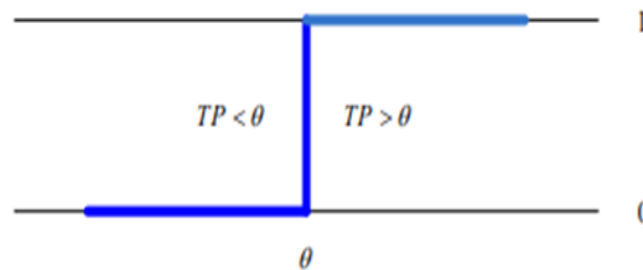


Figure 4-15:- Hard activation function

4.4.5_ Canny edged based segmentation

Segmentation can be defined as grouping of the image parts into many regions. The goal of such image processing operation is to represent some meaningful and needed areas of the image, such as tumors, faces etc...

In other words, the segmentation is the grouping of interesting regions of the image into foreground regions of interest and background regions to be ignored using some techniques such as thresholding, which is done by setting a threshold value. Thus, the pixel values that are lower than the threshold are considered as 0's (black or background), while the pixel values higher than

the threshold are considered as 1's (white or foreground) (Shapiro & Stockman, 2004). Pixel edges are associated with some intensity changes or discontinuities; therefore, edge detection is the process of identifying such sharp intensity contrasts (i.e., discontinuities) in an image. Classical edge detection operators Sobel and Prewitt uses 3×3 kernels which are convolved with the original image to calculate approximations of the derivatives - one for

horizontal changes, and one for vertical. In this proposed system, we detected edges using canny operators. This technique is the most common used method for detecting edges and segmenting the image. The Canny edge detector is considered as one of the best currently used edge detectors since it provides good noise immunity and detects the true edges or intensity discontinuities while preserving a minimum error. Canny operator has been used for such

:(algorithm with regard to the following criteria (Jamil & Saif , 2012

- *.To maximize the signal-to-noise ratio of the gradient
- *.To ensure that the detected edge is localized as accurately as possible
- *.To minimize multiple responses to a single edge

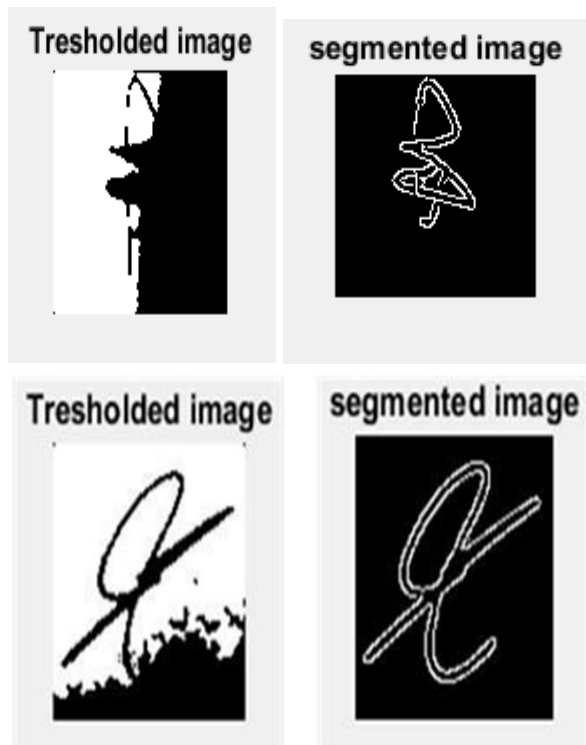


Figure 4-16:- Segmented signature using Canny edge detection

4.4.5.1_ Edge Detection -:

Edges are boundaries between different textures. Edge also can be defined as discontinuities in image intensity from one pixel to another. The edges for an image are always the important characteristics that offer an indication for a higher frequency. Detection of edges for an image may help for image segmentation, data compression, and also help for well matching, such as image reconstruction and so on.

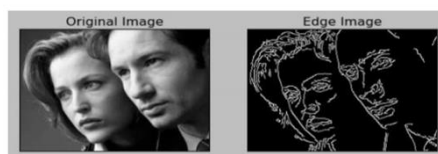


Figure 4-17:- Edge based segmentation (Saif et al., 2012)



There are many methods to make edge detection. The most common method for edge detection is to **calculate the differentiation of an image**. The **first-order**

derivatives in an image are computed using the gradient, and the second-order derivatives are obtained using the Laplacian.

First-Order Derivative Edge Detection

$$\nabla \mathbf{f} = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}.$$

An important quantity in edge detection is the magnitude of this vector, denoted $|\nabla f|$, Where

$$|\nabla f| = \sqrt{G_x^2 + G_y^2}.$$

Another important quantity is the direction of the gradient vector. That is

$$\text{angle of } \nabla \mathbf{f} = \tan^{-1} \left(\frac{G_y}{G_x} \right)$$

Computation of the gradient of an image is based on obtaining the partial derivatives of $\partial f / \partial x$ and $\partial f / \partial y$ at every pixel location. Let the 3×3 area shown in Fig. 4.1 represent the gray levels in a neighborhood of an image. One of the simplest ways to implement a first-order partial derivative at point z_5 is to use the following Roberts cross-gradient operators.

$$G_x = (z_9 - z_5)$$

$$G_y = (z_8 - z_6)$$

AND

These derivatives can be implemented for an entire image by using the masks .shown below with the procedure of convolution

Another approach using masks of size 3×3 shown below which is given by

$$G_x = (z_7 + z_8 + z_9) - (z_1 + z_2 + z_3)$$

AND

$$G_y = (z_3 + z_6 + z_9) - (z_1 + z_4 + z_7)$$

a slight variation of these two equations uses a weight of 2 in the center coefficient:a slight variation of these two equations uses a weight of 2 in the center coefficient:

$$G_x = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)$$

$$G_y = (z_3 + z_6 + z_9) - (z_1 + z_4 + z_7)$$

A weight value of 2 is used to achieve some smoothing by giving more importance to the center point. The following table called the Sobel operators, is used to implement these two equations.

The diagram illustrates the decomposition of a 3x3 matrix into a 3x3 matrix of 1x1 blocks and a 3x3 matrix of 3x3 blocks. The top part shows a 3x3 matrix with blocks Z_1, Z_2, Z_3 in the first row, Z_4, Z_5, Z_6 in the second row, and Z_7, Z_8, Z_9 in the third row. The bottom part shows two 3x3 matrices. The left matrix has blocks $0, 0, 0$ in the first row, $0, -1, 0$ in the second row, and $0, 0, 1$ in the third row. The right matrix has blocks $0, 0, 0$ in the first row, $0, 0, -1$ in the second row, and $0, 1, 0$ in the third row.

The Roberts operators.

-1	-1	-1
0	0	0
1	1	1

-1	0	1
-1	0	1
-1	0	1

The Prewitt operators

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

The Sobel operators

Second-Order Derivative Edge Detection

The Laplacian of a 2-D function $f(x, y)$ is a second-order derivative defined as

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

There are two digital approximations to the Laplacian for a 3×3 region

$$\nabla^2 f = 4z_5 - (z_2 + z_4 + z_6 + z_8)$$

$$\nabla^2 f = 8z_5 - (z_1 + z_2 + z_3 + z_4 + z_6 + z_7 + z_8 + z_9)$$

Masks for implementing these two equations are shown in Fig. 4-20

0	-1	0
-1	4	-1
0	-1	0

(a)

-1	-1	-1
-1	8	-1
-1	-1	-1

(b)

Figure 4-18:- Two kind of 3×3 Laplacian mask

The Laplacian is usually combined with smoothing as a precursor to finding edges via zerocrossings. The 2-D Gaussian function

$$h(x, y) = -e^{-\frac{x^2 + y^2}{2\sigma^2}}$$

where σ is the standard deviation, blurs the image with the degree of blurring being determined by the value of σ . The Laplacian of h is

$$\nabla^2 h(x, y) = -\left[\frac{x^2 + y^2 - 2\sigma^2}{\sigma^4} \right] e^{-\frac{r^2}{2\sigma^2}}$$

This function is commonly referred to as the Laplacian of Gaussian (LOG)

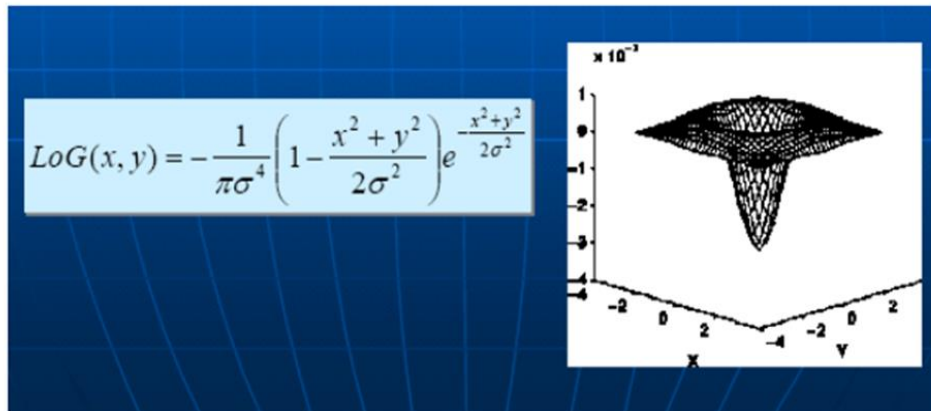


Figure 4-18: Dimension coordinate of Laplacian of Gaussian (LOG)

After calculating the two-dimensional second-order derivative of an image, we find the value of a point which is greater than a specified threshold and one of its neighbors is less than the 20 negative of the threshold. The property of this point is called zero-crossing and we can denote it as an edge point

We note two additional properties of the second derivative around an edge:

(1) It produces two values for every edge in an image (an undesirable feature); and (2) an imaginary straight line joining the extreme positive and negative values of the second derivative would cross zero near the midpoint of the edge. This zero-crossing property of the second derivative is quite useful for locating the centers of thick edges

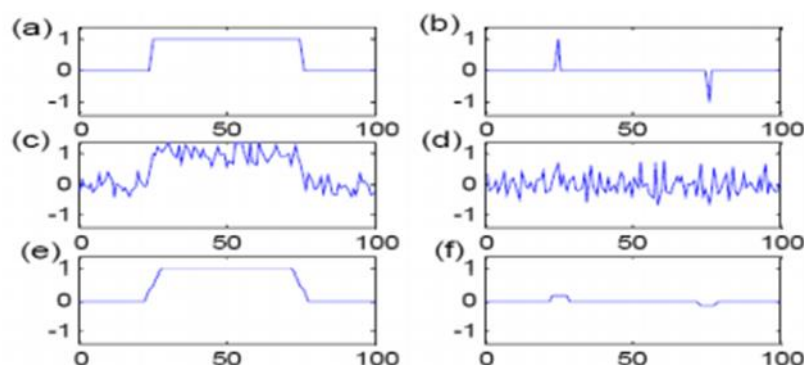


Figure 4-19: The results of differentiation of using the ramp edges

4.4.6_ Features extraction and rescaling using pattern averaging

After the segmentation process using the canny edge detection, the images size should be reduced in order to be fed to the neural network. To reduce the size of images while keeping the useful and needed features extracted .by the previously used methods, we used patter averaging

This technique is defined as the averaging of the defined segments of the image by selecting a window 4*4 segments that are averaged. Therefore, each studied pixel is then the average of the 16 neighbor's pixels in the selected window. Thus, we come up with a rescaled image of size 64*64 pixels with the same features and properties of the original one for the purposes of fast

processing and easy computing. An intelligent blood cell identification system was developed by Adnan Khashman (Adnan Khashman, 2008) for the identification of the three blood cells. The authors used pattern averaging to .reduce the size of the blood cell images while preserving the needed features

Below in Figure 4-20 is shown some of processed rescaled images.

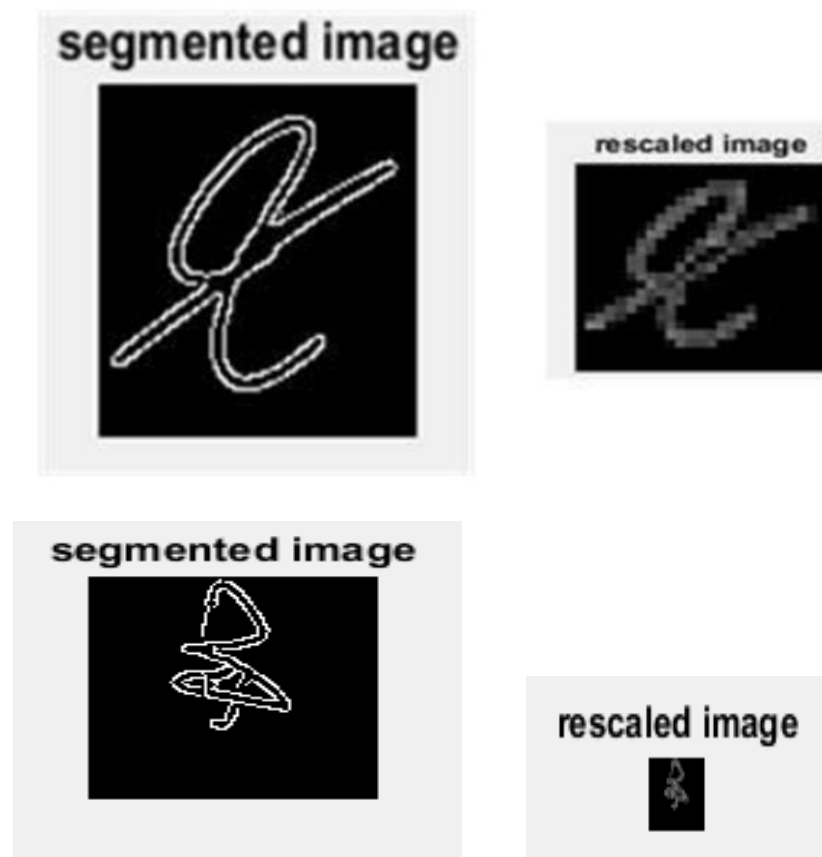


Figure 4-20:- Rescaled image using pattern averaging

4.5 Segmented Signatures:

At the end of this image processing phase the images are saved all in a file. This was meant to evaluate the effectiveness of the system and to assure that all images were correctly segmented and rescaled. Figure 4-21 below shows a sample of the segmented signatures processed using the developed system

Figure 4-22 shows a sample of the rescaled processed images using the proposed system.

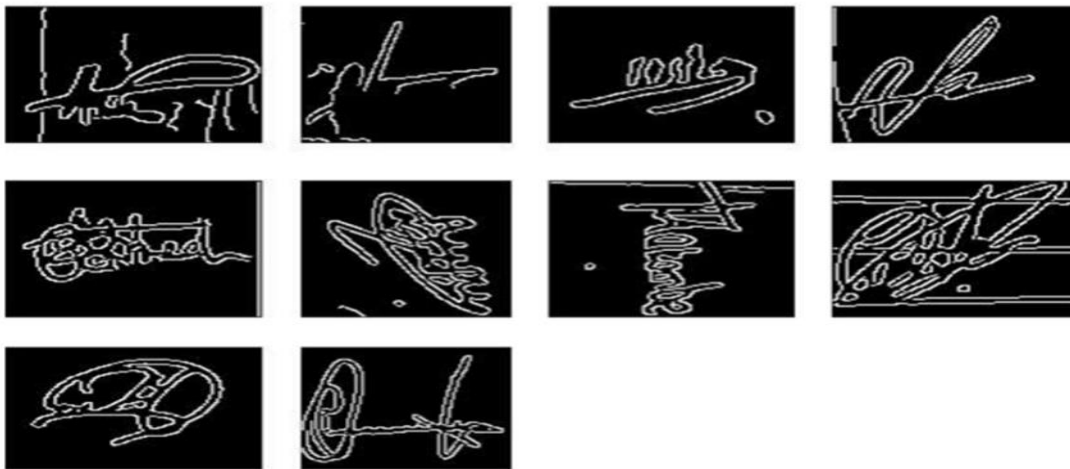


Figure 4-21:- : Sample of the processed and segmented signature image using the proposed system



Figure 4-22:- Sample of the rescaled signatures processed using the proposed system

4.6_ NETWORK TRAINING AND PERFORMANCE :

4.6.1_Introduction

This chapter describes the collection of the images that is used in both training and testing this work, as well as the backpropagation neural network design, training, and testing which are implemented to evaluate the aim of this research, and the simulation of the developed system to show its effectiveness in recognizing the human signatures.

It is the aim to develop an artificial vision system that can perform the task of recognizing signatures in images. In this work, considering challenges such as object illumination, scale, translation, simple differences in signature, etc. which make the recognition a complex problem for such an open recognition problem, we resolve to implement an intelligent system which can Somewhat graciously cope with the aforementioned recognition constraints. A back propagation neural network (BPNN) has been used in this work as an intelligent classifier to perform this recognition task. This research is achieved in two phases. First is the human signatures processing phase by segmenting the signatures in an image and rescaling it to fit the designed neural network. The second phase is the recognition of signatures from images using the back propagation neural network. The second phase involves training and testing of the network by many signature images collected from a classroom in the Near East University.

The flowchart for the system is shown in Figure 4-23; and both phases are briefly described below.

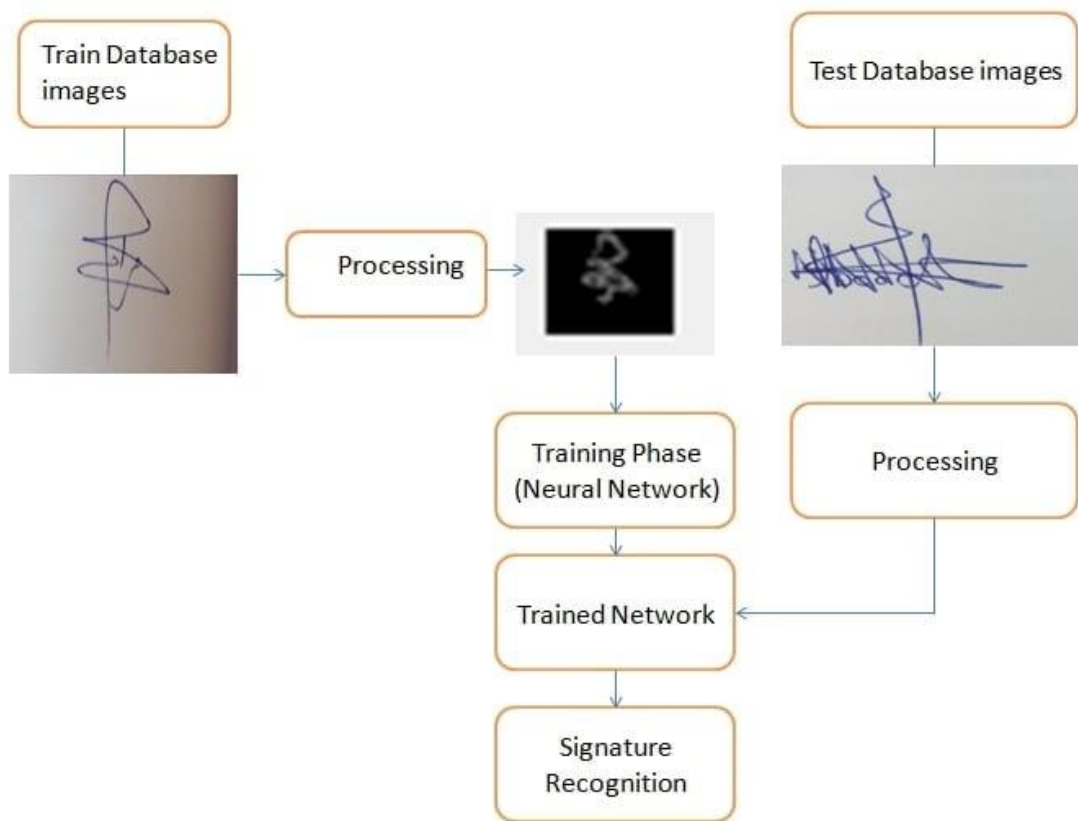


Figure 4-23 : Training and testing phases of the developed signature recognition system

The figure above shows the flowchart of the proposed intelligent signature recognition system using backpropagation neural network. The images are used to train the backpropagation network through their features which are extracted pattern averaging after the signature is segmented using thresholding and canny edge detection techniques. After training and convergence, the testing images are used then for testing the neural network after they pass through the segmentation and rescaling phase.

4.6.2_ Backpropagation Neural Network Classification

In this phase: a back propagation neural network is trained to recognize human signatures in an image. In order to achieve this binary classification task, training data is collected to have many different signature images. As previously discussed some signatures were written more than one time in order to increase the efficiency of the system and to use the others for training. In this project, all training and testing data are collected from our colleagues. Since, the actual

Samples of positive and negative examples collected from the internet are shown below in Figure 4-24 .



Figure4 -24 :- Sample of training imager

A back propagation neural network is trained on the collected sample signature images. These one or more of each signature were used for training the back propagation neural network (BPNN). All images are converted to from color to grayscale, filtered, segmented using tresholding and canny edge detection, and finally rescaled to 64*64 (4096 pixels) using pattern averaging for fast processing and preserving features. The whole data is divided into training and testing data. The testing data allows the observation of performance of the trained BPNN on unseen or new data. It is very desirable that trained ANNs can perform well on unseen data. Note that unseen data means same data used for training but written as a second or third time by our colleagues which causes a change in the image. This change may be either scale translation, or simple differences in signature. This leads to a robust system capable of recognizing signatures in different shift translations and different hand writings.

Chapter 5: Results and Analysis

5.1_ Results:

This project describes a challenging task in the field of artificial intelligence, as well as image processing. We demonstrated that backward diffusion neural network (BPNN) can be used to learn robust recognition / classification of human handwritten signatures. Then a BPNN trained with a non-nested sampling method is used to "check" the target images containing the identifiable signatures. It is a very effective system for recognizing signatures after image segmentation using image processing techniques. And smart so that picture scene conditions such as shift translation and small signature differences only slightly affect the overall efficiency of the system.

One the most difficult challenges in this work were to obtain high recognition rate in the training phase. This is due to the fuzziness of the segmented images since they are signatures which may be sometimes very similar or have no much difference so that the network will learn. Therefore this makes it difficult for the network to learn.

Thus, our system is to segment the images first

using image processing techniques such as **Canny edge detection that is used to reduce the**

amount of data to be learnt by the network. In addition, to reduce the unneeded features of the image, **pattern averaging technique was used to reduce image size and to extract the important features of the signatures.**

5.2_Overall Flowchart Diagrams and codes

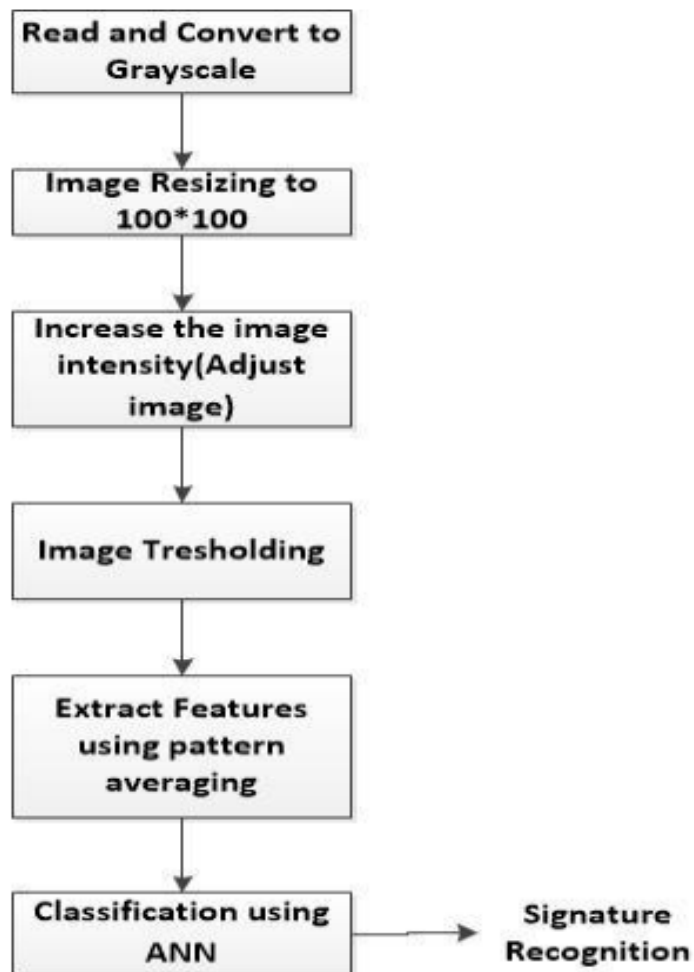


Figure 5.1 :-flow chart of signature recognition by canny edge detection

5.2.1 _ Matlab code for Image Processing

In this part we wrote a matlab code for the 6 image processing we used in our project

As shown in the next image :

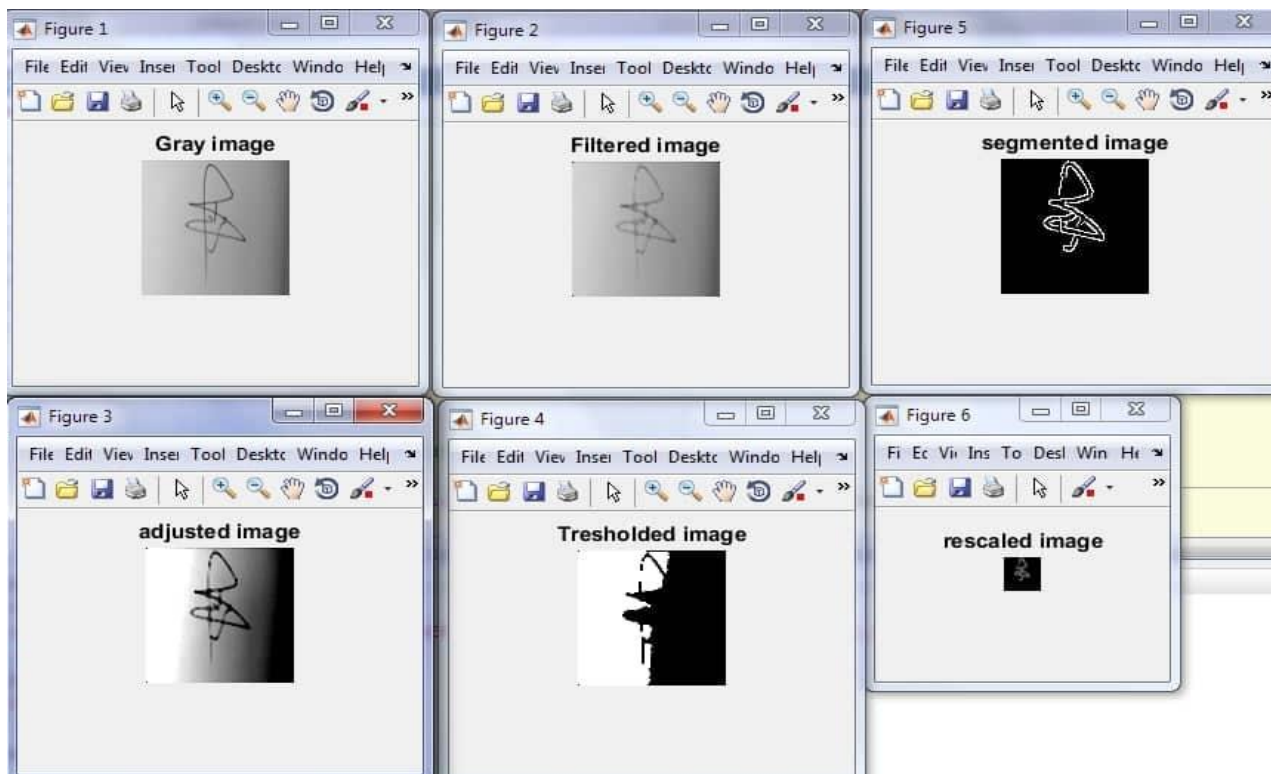
```
1  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Processing one image
2  - f=imread('s52.jpg');
3  - f=rgb2gray(f);
4  - f=imresize(f, [100 100]);
5  - figure, imshow(f), title('Gray image');
6  - pause
7  - f=medfilt2(f, [3 3]);
8  - figure, imshow(f), title('Filtered image');
9  - pause
10 - %f=imresize(f, [256 256]);
11 - g2=imadjust(f,[0.5 0.75], [0 1]);
12 - figure,imshow(g2), title('adjusted image');
13 - pause
14  %% threshold the image
15 - level = graythresh(g2);
16 - bw = im2bw(g2,level);
17 - bw = bwareaopen(bw, 50);
18 - figure,imshow(bw), title('Treshholded image');
19 - pause
20  % Labeled the component(s), and plot the centroid on the original image
21  % Canny
22 - S5 = edge(g2, 'canny', 0.25);
23 - figure,imshow(S5), title('segmented image');
24 - pause
```

```

22 - S5 = edge(g2, 'canny', 0.25);
23 - figure,imshow(S5), title('segmented image');
24 - pause
25 - m3=S5;
26 - T = blkproc(m3, [4 4], @mean2);
27 - figure,imshow(T), title('rescaled image');
28 - pause
29 - close all

```

Figure 5.2 :-matlab code for image processing in a signature



5.3 :-result image processing code

5.3 Experimental Setup & Performance Evaluation

In this section, the experimental setup, handwritten signature image database, different features, and different number of nodes in the hidden layer will be discussed

A high performance system will be used for processing, there are two feature extraction techniques, i.e. Canny edge detection (high pass filter) and averaging filter (low pass filter).

Performance Evaluation From Table 5-1, for the **canny edge detection**, as the number of hidden layers decreases the recognition rate also decreases, and as the number of hidden layers increases the recognition rate also increases. It shows that it **can achieve 100 % recognition rates** when the hidden layers are **1000**. It also shows that the type of filter used (feature extraction) also plays an **important role as it will improve the quality of the image and recognition rate** and will be increasing. However, for the averaging filter, it is not showing a consistent result. Sometimes, a higher number of hidden layers shows a lower recognition rate. **In conclusion, the number of hidden layers and types of filter will influence the performance and recognition rate of the system.** There are many variations in the handwritten signature database due to time differences and writing tools. However, the **canny edge detection has been found to be the most suitable feature as it gives a high recognition rate**[13]

<u>N_{hidden}</u>	Features Extraction	Training Time (seconds)	Recognition Rate(%)
1000	Canny edge detection	637.5	100
500		68.7	84
100		3.8	72
80		0.9	56
800	Averaging filter	155.1	64
500		135.2	68
100		4.1	64
80		3.1	76

Table 5-1:- The effect of number of nodes in the hidden layers and different features on the training time and recognition rate

5.4_Results Comparison

As mentioned in the literature review section, many studies were conducted for the purpose of recognizing signatures using intelligent systems. Each work was based on different image processing techniques. However, our proposed work suggested a new technique for the signature recognition which is to use the pattern averaging algorithm for the extraction of features from the signature image after it is segmented using thresholding and Canny edge detection algorithms. The developed system includes the use of backpropagation neural network as an intelligent classifier to learn the extracted features

The table 5-2 shows the results comparison of our signature recognition system with some other systems that used different databases but same classifiers. It can be seen that the developed system performs well in the generalization phase since its accuracy is either equal or higher than the other proposed researches[14]

Paper Title	Authors	Methods used	Recognition Rate
Biometric Signature Processing & Recognition Using Radial Basis Function Network	Ankit Chadha, Neha Satam, and Vibha Wali (Chadha et., 2013)	Signature Processing and Radial Basis Function Network	80 %
Handwritten Signature Verification using Local Binary Pattern Features and KNN	Jadhav and Chavan	Local Binary Pattern Features using K-Nearest Neighbor algorithm	75.34%
		Local Binary Pattern Features using Support Vector Machine algorithm	90.36%
Signature Recognition System	Salmin Mohamed	Canny edge detection , Pattern averaging and Backpropagation neural network	Depend on # Of Hidden Layers
Signature Recognition based on Discrete Wavelet Transform	Sivana Salahaldin Muhamed , Muzhir Shaban Al-Ani	Signature Recognition based on Discrete Wavelet Transform (DWT)	100%

Table 5-2:- Results comparison with other works

Chapter 6

Discussion

6.1 Completing the proposed system

This project aims to solve the weaknesses and problems that people face in forging signatures and impersonating characters. There are many techniques that are used to verify the authenticity of the signature, some of them are mentioned in Table (5-2).

In our project, a modern and accurate technique was chosen, which is the edge technique using an artificial neural network.

The Canny edge detector is considered as one of the best currently used edge detectors since it provides good noise immunity and detects the true edges or intensity discontinuities while preserving a minimum error.

6.2 SWOT Analysis

6.2.1 Project Strength

Some of the strengths of the Signature of ANN by used canny edge method effective technique of segmenting the signatures and also the extraction of features of the image which facilitates the learning stage of the neural network.

Canny edge methods

- provides good noise immunity
- detects the true edges or intensity discontinuities while preserving a minimum error.
- maximize the signal-to-noise ratio of the gradient.
- ensure that the detected edge is localized as accurately as possible.
- To minimize multiple responses to a single edge.
- the most popular method for edge detection and is a form of highpass filter
- the higher the number of nodes, the higher the recognition rate, although the training time is also higher. (achieve 100 % recognition rates when the hidden layers are 1000).

back propagation algorithm

- uses the theory of error minimization and gradient descent to find the least squared error
- uses a feed forward process, a back propagation updating method, and supervised learning topology.
- A back propagation network with a given hidden layer of elements can simulate any function to any degree of accuracy.

6.2.2 Project Weaknesses

- Back propagation method is costly in terms of processing requirements for learning.
- 2_ the recognition by this method predictable to take from 2 mins , which if we increase the number of the hidden layers, we get a higher recognition rate ,reach to 100%, but it is take time longer .

6.2.3 Project Opportunities:

This project supports the process of verifying the validity of signatures, and this is considered one of the most important ways to protect the security of information, especially financial and banking, so it will be one of the most important ways to use it in banks, whether it is to prove the identity of the owner inside the bank, or even to approve it to verify the signatures on bank checks, which will be easy to perform the signature scanning process as a single part of the check, and then enter it on the AI network.

Also, the process of verifying the authenticity of a person's signature is one of the most important methods that help the police to detect and investigate police crimes to maintain security.

6.2.4 Project Threats:

- 1) There are some expected threats to this project, perhaps the most important of which is the existence of alternative methods that done the recognition process in less time .

2)Because of our choice of a project based on artificial neural networks, which is a topic with a little background for us, first, we encountered some obstacles in finding the algorithm and mythologies suitable for our project, as we adopted our current method after a detailed search for more than one method for the same topic, but we chose this method for specific goals and explained it in the introduction .

3) Finally ,The time for our work on the first graduation project came, the continuation of the Corona virus Covid 19 pandemic, which constituted an obstacle for us at various parties, including the difficulty and lack of face-to-face and interactive meetings between group members and the teacher, despite that ,there were ways to communicate such as zoom, email and Facebook .

Chapter 7

Conclusion and Future Work

7.1 Conclusion

Mark or signature is an exceptional instance of handwriting which incorporates extraordinary characters and twists. Numerous signatures can be ambiguous. They are a sort of masterful handwriting objects. However, a mark or signature can be taken care of as a picture, and consequently, it can be perceived utilizing computer vision and counterfeit neural system strategies.

The point of the thesis is to develop a system that is capable of reading images and using acquired experience (during training) to determine the human identity of a signature found in the image.

It is important that the task of recognizing signatures is not one that is easily achieved in computer vision, as some critical image constraints such as object scale, translation, rotation, illumination, noise, and difference in signatures of the same individual, etc. make the task quite difficult. Thus, our system start by processing the images in order to remove noise first, and then segment the signature to make it easy for the network to learn since what is needed in the image is the signature only. Moreover, the system uses a very efficient feature extraction technique for the same purpose. This technique reduces the size of signature by taking only the important features to be fed into a backpropagation neural network.

In fact, our proposed signature recognition system is simple as compared to other proposed systems. However, the results obtained after testing it proved that it is more effective than the other related systems. This is due to effective technique of segmenting the signatures and also the extraction of features of the image which facilitates the learning stage of the neural network. Note that the feature extraction using pattern averaging also helped in distinguishing the human handwriting signatures used for training and testing the network.

Finally, it can be stated that a simple signature recognition system was developed in this thesis. Regardless of the simplicity of the system it is efficient enough to be compared to more complex and advanced related recognition systems. This is due to the effective image proceeding and feature extraction techniques used for segmenting and extracting the signatures.

7.2 Future Works

We used a method off line signature recognition, the first part of which has been applied in matlab image processing, provided that the training of the data base will completed in graduation project database 2 , and we will study the methods of each of the CNN && DWT && as other methods in the training process to find any methods bypassing the problem of delay in time and give the highest accuracy .and recommendation for future work is the exploration of support vector machine (SVM) for the recognition task. It is conceived that using SVM can help eliminate the classifier retraining as obtains in the BPNN; since SVM is a maximum margin classifier that always converge to the same local minima

We will also discuss the process of verifying the authenticity of signatures using online recognition using its application on smartphones.

CHAPTER 8

References

- [1] (T. S. Gunawan, I. Z. Yaacob, M. Kartiwi, N. Ismail, N. F. Za'bah, and H. Mansor, "Artificial Neural Network Based Fast Edge Detection Algorithm for MRI Medical Images," Indonesian Journal of Electrical Engineering and Computer Science, vol. 7, pp. 123-130, 2017.)
- [2] S. MOHAMED NEAR EAST UNIVERSITY 2016
(T. S. Gunawan, I. Z. Yaacob, M. Kartiwi, N. Ismail, N. F. Za'bah, and H. Mansor, "Artificial Neural Network Based Fast Edge Detection Algorithm for MRI Medical Images," Indonesian Journal of Electrical Engineering and Computer Science, vol. 7, pp. 123-130, 2017.)
- [3] Kröse, B., Krose, B., van der Smagt, P., & Smagt, P. (1993). An introduction to neural networks. *Procedia Computer Science. Educational Technology & Society*, 12(2), 176-175
- [4] Drouhard, J-P., Robert Sabourin, and Mario Godbout. "A neural network approach to off-line signature verification using directional PDF." *Pattern Recognition* 29.3 (1996): 415-424.
- [5] Drouhard, J-P., Robert Sabourin, and Mario Godbout. "A neural network approach to off-line signature verification using directional PDF." *Pattern Recognition* 29.3 (1996): 415-424.
- [6] Oz, C. (2005, May). Signature recognition and verification with artificial neural network using moment invariant method. In *International Symposium on Neural Networks* (pp. 195-202). Springer Berlin Heidelberg.
- [7] <https://ch.mathworks.com/help/signal/ref/medfilt1.html>
- [8] <https://ch.mathworks.com/help/images/ref/imresize.html#d123e170797>
- [9] <https://ch.mathworks.com/help/images/ref/imadjust.html>
- [10] <https://ch.mathworks.com/help/images/ref/im2bw.html>
- [11] <https://ch.mathworks.com/matlabcentral/fileexchange/46859-canny-edge-detection>
- [12] <https://ch.mathworks.com/help/images/ref/imresize.html#d123e170797>

[13]

publication/323649751_Development_of_offline_handwritten_signature_authentication_using_artificial_neural_network Gunawan, T. S. and M. Kartiwi (2018). "On the Use of Edge Features and Exponential Decaying Number of Nodes in the Hidden Layers for Handwritten Signature Recognition." Indonesian Journal of Electrical Engineering and Computer Science 12(2): 722-728.

[14] Automated Bank Cheque Verification Using Image Processing and Deep Learning Methods Prateek Agrawal · Deepak Chaudhary · Vishu Madaan · Anatoliy Zabrovskiy · Radu Prodan · Dragi Kimovski · Christian Timmerer Prateek Agrawal University of Klagenfurt, Austria Lovely Professional University, India