A PROJECT REPORT ON

**Deep Learning Based Offline Signature Verification-A Comparative Study Between Writer Dependent & Writer Independent Methods**

Presented to the faculty of Engineering,



Institute of Engineering & Management, Kolkata.

In partial fulfilment of requirements for the degree of

B.Tech (I.T. Engineering)

Submitted by,

**Pratik Patra**

ROLL NO: 10400214075

**&**

**Rajdeep Mukherjee**

ROLL NO: 10400214080

Under the Guidance of

**Prof. Mr. Tapan Kumar Hazra**

**Department of Information Technology, IEM**

ACKNOWLEDGEMENT

We propose our sincere thanks to our esteemed project guide Prof. Tapan Kumar Hazra for providing us with the right guidance at crucial stages and showing us the right way ahead.

We extend our sincere and heartfelt thanks to the efforts of our HOD madam, Prof. Dr. Mohuya Chakraborty, Department of Information Technology and all the other faculty members of the department of Information Technology of Institute of Engineering and Management(IEM), Kolkata, for encouraging us and allowing us to use all the available facilities during the course of our work.

|  |  |
| --- | --- |
|  |  |
| Pratik Patra (B.Tech IT) | Rajdeep Mukherjee |
| Dept. of Information Technology | Dept. of Information Technology |
| Roll No. 10400214075 | Roll No. 10400214080 |
| Reg No. 141040110281 | Reg No. 141040110286 |



Abstract

The objective of this report is to provide a brief introduction to signature verification, its various applications and types. The discussion will be followed by an overview of deep neural networks and their application in machine learning problems. Signatures are still necessary means of identification for several professional transactions. The existing process of verification of signature requires human intervention and is prone to human error. A predefined signature dataset has been used for both training and testing purposes. The proposed system uses training of deep neural networks in features of genuine signatures and using the knowledge to classify them as genuine or forged. The procedure consists of a stage of pre-processing where the signature image is processed into the required format. The image is opened in grayscale and is then normalized with standard size. Thresholding is then used for binarization of the image which will scan the entire image and check whether the pixel value is greater than the threshold value. The binary image is then eroded to thin the pixels along the edges. Signature images are processed into this form and then features will be extracted and stored. The important features of the pre-processed signatures are then extracted from the images. The features that are considered for the system include both global and local features. The global features that have been extracted are baseline slant angle and aspect ratio. The local features that have been used are skewness, entropy and center of gravity. The dataset has labels for genuine and forged signatures. A multilayer perceptron has been modelled for the training purpose. The model is a sequential model with 1 input layer, 1 output layer and 1 hidden layer. The trained network is then tested on a different dataset and the accuracy of the model is then computed. The training and testing will be done in two separate modes- writer dependent and writer independent. In writer independent mode, the identity of the signer is not known and the neural network will classify the signatures as genuine or forged without taking into consideration the unique features of each signature. In writer dependent mode, the identity of the signer is known and the dataset will comprise of genuine and forged signature samples of the same identity and is tested accordingly. The relative performances of these modes are studied. The evaluation has been done based on the following parameters- False Acceptance Rate, False Rejection Rate and Correct Classification Rate. The results and their conclusions are subsequently summarized in the end.

**Contents**

[Chapter 1 INTRODUCTION 1](#_Toc512339864)

[1.1 Overview 2](#_Toc512339865)

[1.2 Problem Statement 7](#_Toc512339866)

[1.3 Objective 7](#_Toc512339867)

[Chapter 2 LITERATURE SURVEY 8](#_Toc512339868)

[Chapter 3 PROPOSED MODEL 11](#_Toc512339869)

[3.1 The Proposed Model 12](#_Toc512339870)

[3.2 Model Description 12](#_Toc512339871)

[3.3 Image Preprocessing 12](#_Toc512339872)

[3.4 Feature Extraction 14](#_Toc512339873)

[3.5 Features Used For The Model 14](#_Toc512339874)

[3.6 Dataset Description 15](#_Toc512339875)

[3.7 Types of Classification 16](#_Toc512339876)

[3.8 Evaluation Parameters 16](#_Toc512339877)

[Chapter 4 RESULTS AND IMPLEMENTATION 18](#_Toc512339878)

[4.1 Neural Network Structure 19](#_Toc512339879)

[4.2 Writer Independent Mode Results 22](#_Toc512339880)

[4.3 Writer Dependent Mode Results 27](#_Toc512339881)

[Chapter 5 CONCLUSION AND FUTURE SCOPE 32](#_Toc512339882)

[5.1 Conclusion 33](#_Toc512339883)

[5.2 Future Work 33](#_Toc512339884)

[Chapter 6 REFERENCES 34](#_Toc512339885)

**Table of Figures**

[Figure 1 : Raw Image 13](#_Toc512284638)

[Figure 2 : Pre-processed Image 13](#_Toc512284639)

[Figure 3 : Comparison between genuine and forged signature for same ID 16](#_Toc512284640)

[Figure 4 : Neural Network Structure (Multilayer Perceptron) 22](#_Toc512284641)

[Figure 5:Dataset for Writer Independent Classification 23](#_Toc512284642)

[Figure 6 : Writer Independent Mode Data 24](#_Toc512284643)

[Figure 7 : CCR with Test Size Vs Number of Iteration 25](#_Toc512284644)

[Figure 8 : Accuracy Vs Testing Set Size (100 iterations) 25](#_Toc512284645)

[Figure 9 : FAR with Test Size Vs No. of iterations 26](#_Toc512284646)

[Figure 10 : FRR with Test Size Vs No. of iterations 26](#_Toc512284647)

[Figure 11 : Dataset for Writer Dependent Classification 27](#_Toc512284648)

[Figure 12 : Writer Dependent Mode Data 28](#_Toc512284649)

[Figure 13 : CCR with Test Size Vs Number of Iteration 29](#_Toc512284650)

[Figure 14 : Accuracy vs Testing Set Size (100 iterations) 29](#_Toc512284651)

[Figure 15 : FAR with Test Size Vs Number of Iteration 30](#_Toc512284652)

[Figure 16: FRR with Test Size Vs Number of Iteration 31](#_Toc512284653)

Chapter 1  
INTRODUCTION

1. Overview

Biometric authentication is the process of verifying the identity of individuals based on their unique biological characteristics. It has become a ubiquitous standard for access to high security systems. Current methods in machine learning and statistics have allowed for the reliable automation of many of these tasks (face verification, fingerprinting, iris recognition). Among the numerous tasks used for biometric authentication is signature verification, which aims to detect whether a given signature is genuine or forged. Signature verification is an important biometric technique that aims to detect whether a given signature is forged or genuine. It is essential in preventing falsification of documents in numerous financial, legal, and other commercial settings. The task presents several unique difficulties: high intra-class variability (an individual’s signature may vary greatly day-to-day), large temporal variation (signature may change completely over time), and high inter-class similarity (forgeries, by nature, attempt to be as indistinguishable from genuine signatures as possible).

The existing methods for signature verification are manual and often time consuming since it is the job of a person to view and authenticate a signature based on a pre-existing sample in the database. If such a process is computerized, then the process will become faster and will also be less prone to errors to some extent. The way a human processes a signature is through intuition and experience and because of this property, it is difficult for a traditional computer algorithm to mimic the procedure. It is in this regard that neural networks are useful for this process. Neural networks naively mimic the way a human brain processes information and this property will make it easier to perform the signature verification process. Neural networks (NNs) have been a fundamental part of computerised pattern recognition tasks for more than half a century, and continue to be used in a very broad range of problem domains. The two main reasons for their widespread usage are: 1) power (the sophisticated techniques used in NNs allow a capability of modelling quite complex functions); and 2) ease of use (as NNs learn by example it is only necessary for a user to gather a highly representative data set and then invoke training algorithms to learn the underlying structure of the data).

Artificial neural networks are models motivated by the brain that is competent of machine learning and pattern recognition. They are usually presented as organization of interconnected "neurons" that can calculate values from inputs by providing information through the network. Neural networks are characteristically structured in layers. Layers are consisting of a number of interrelated 'nodes' which hold an 'activation function'. Patterns are available to the network by means of the 'input layer', which communicate to one or more 'hidden layers' where the concrete processing is done using a system of subjective 'connections'. The hidden layers then unite to an 'output layer' where the answer is final output of the system.

Signatures are of two forms-online and offline. Online signatures are obtained by making the signer sign on a specialized piece of hardware, like a tablet. Online signatures contain much more information than offline signatures, including the writing pressure, velocity of pen, etc. However, online signatures require specialized hardware and have less applicability since handwritten signatures are still not predominant in the digital space. Offline signatures are obtained by making the signer sign on a piece of paper and then scanning that image to obtain it in a digital format. Offline signatures are widely applicable since it is the most common means of attaining a signature. Offline signatures provide much less data compared to online signatures, since they are simply images and contain only pixel information. The offline signatures are thus used in this project because of their huge applicability.

The objective of the signature verification system is to discriminate between two signature classes the genuine and fake signature. A lot of effort has been applied in the field of off-line signature verification. Forgery is a crime that aims at misleading people. Since actual forgeries are difficult to obtain, the instrument and the results of the verification depend on the type of the forgery. There are basically three types of forgeries as

1) Random forgery - This can normally be represented by a signature sample. Forger has no information about the signature style and the name of the person

2) Simple forgery - This is a signature with the same shape or the legitimate writer’s name

3) Skilled forgery - This is signed by a person who has had access to a genuine signature for practice

Although a great amount of work is determined on random and simple forgery detection, more hard work is still needed to tackle the problem of skilled forgery detection. No verification algorithms are proposed which might be deal with skilled forgeries.

Using the raw images for the dataset will not be fruitful since the images might not all be in the same format or dimension. A standard format needs to be maintained in order to process the images. It is in this stage that the image pre-processing begins. The first step in pre-processing begins with changing the dimensions of the image into a standard format. The image is then converted into grayscale. The image is then converted into a binary image. A binary image is a digital image that has only two possible values for each pixel. Typically, the two colours used for a binary image are black and white. The colour used for the object(s) in the image is the foreground colour while the rest of the image is the background colour. In the document-scanning industry, this is often referred to as "bi-tonal". Binarization is a process that converts the image pixel values into exactly 0 or 1 depending on a threshold value. Thresholding is only one of the processes that is used for binarization. Thresholding is the simplest method of image segmentation. From a grayscale image, thresholding can be used to create binary images. The simplest thresholding methods replace each pixel in an image with a black pixel if the image intensity Ii , j is less than some fixed constant T (that is, Ii , j< T ), or a white pixel if the image intensity is greater than that constant. This results in the dark pen lines in the signature becoming completely black, and the white background becoming completely white. The binary image is then thinned into low pixel width through the process of erosion. Erosion is a process where the pixel value computed is minimum. The image is replaced under the anchor point with that minimum pixel value. With this procedure, the areas of dark regions grow in size and bright regions reduce. For example, the size of an object in dark shade or black shade increases, while it decreases in white shade or bright shade.

The image is now in a form that is suitable for feeding into the neural network. However, instead of directly feeding the images into the network, important features of the image are extracted and then fed into the network. Features are the information extracted from images in terms of numerical values that are difficult to understand and correlate by human. Suppose we consider the image as data the information extracted from the data is known as features. Generally, features extracted from an image are of much lower dimension than the original image. The reduction in dimensionality reduces the overheads of processing the bunch of images.In Feature extraction, the essential features are extracted from the original input signature. The features to be extracted are based on the application and fluctuate accordingly. Characteristic constraints are computed from the sorted out data and are used to characterize signature. The choice of a powerful set of features is crucial in signature verification systems. The features that are extracted from this phase are used to create a feature vector. We use a feature vector to uniquely characterize a candidate signature. The features extracted are of two types-global features and local features.

Global features are extracted from the whole signature, including block codes, Wavelet and Fourier series. The global features can be extracted easily and are tough to noise. But they only deliver limited information for signature verification. Global features offer information regarding shape like signature area, signature height-to-width ratio, slope & slope direction skewness of signature etc. Global features describe the image as a whole to the generalize the entire object whereas the local features describe the image patches (key points in the image) of an object. Global features include contour representations, shape descriptors, and texture features.

The global features used in our model are- baseline slant angle and aspect ratio. Baseline is the imaginary line about which the signature is assumed to rest. The angle of inclination of this line to the horizontal is called the Slant Angle ϴ. To deter-mine the slant angle the ratio of the maximum horizontal projection to the width of the projection is maximized over a range of values of angle of rotation Ф.

Local features are calculated to describe the geometrical characteristics such as location, tangent track, and curving. Local features provide affluent descriptions of writing shapes and are powerful for cultivated writers, but the extraction of consistent local features is still a hard problem. Local features offer information on the content of the image within key points, unlike global features, which are properties of the entire image. Local features are harder to extract since they require highly specialized algorithms for very specific purposes. Local features provide more information about the image since they are unique for a given class of images.

The local features used in our model are part of statistical features and geometric features. Features which describe the pixel distribution of the image are known as statistical features. For this model, we have used Skewness and Entropy.

Statistically, skewness measures the asymmetry of a real-valued random variable's probability distribution. A skew value less than zero says that the tail on the left side of the probability density function is longer and the bulk of the values (possibly including the median) lie to the right of the mean. A skew value above zero says that the tail on the right side is longer and the bulk of the values lie to the left of the mean. A zero skew value indicates that the values are typically evenly distributed on both sides of the mean, usually but not necessarily implying a symmetric distribution. A darker or glossier surface is typically more positively skewed than a lighter and matte surface. Hence, it is possible to use skewness in making judgments about image surfaces.

Entropy measures the amount of randomness or uncertainty is present in an image. It acts as a measure of how much information is provided by the image.

Geometrical Features are the features describing the geometric characteristics of the image and are less susceptible to distortion. For this model, we have used the center of gravity.

All the features extracted here are converted into a feature vector, each of them representing an individual signature. The feature vectors are combined in rows to form a signature feature dataset. The labelled feature vectors extracted from the pre-processed images are then fed into a deep neural network and is trained with genuine and forged signatures. The deep neural network is in the form of a multilayer perceptron. Multilayer perceptron (MLP) is a feedforward neural network with one or more layers between input and output layer. Feedforward means that data flows in one direction from input to output layer (forward). This type of network is trained with the backpropagation learning algorithm. Various training and testing ratios are utilised and the ratio with best accuracies are highlighted. The training and testing procedures will be performed in two separate modes-writer dependent and writer independent modes.

Writer dependent classification - Writer dependent classification involves training the dataset with another label which contains the id of the signer. Here the training is done in a way that will involve one id and several signature samples for the same id corresponding to a single signer. The signature samples consist of both forged and genuine signatures. During the testing phase, the id will be known to the network. The scope in this is limited since it is required that a sample be present for each signature in the system and it is assumed that forgeries of each class of signature is already available. However, since the problem space is smaller, the accuracy of predictions is higher in this approach.

Writer independent classification - Writer independent classification involves training the network with both genuine and forged signatures without taking into consideration the identity of signer. The classification is based on the network being able to differentiate between genuine and forged signatures based on previous training. This approach removes the requirement for an extra id and avoids the cost of labelling the data for an individual signer. This approach is universal since the identity of the signer is not required and the number of samples to be trained with is unconstrained. However, since the problem space is larger and since every signature has its own unique characteristics, the accuracy is significantly lower than writer dependent classification.

A comparison between these two aspects form the crux of the result section of this report. The comparison will be done on basis of three percentage parameters - Correct Classification Rate(CCR), False Acceptance Rate(FCR), False Rejection Rate(FRR). They are given by the following formulae-

1. Problem Statement

The process of signature verification is crucial in identification and authentication in several industries, especially in the banking and financial services (BFS) sector. In a lot of cases, signature verification can be prone to human error because a lot of this verification is still done by the human eye. A signature can be described as a set of features and a collection of these features can be used to uniquely identify a signature. Signature input can be of two types: offline and online. Signature verification can follow two methods: writer-independent and writer-dependent. In the writer-independent model, the model is not aware of the identity of the signer. In a writer-dependent model, the model is aware of the signer’s identity.

1. Objective

The objective of this project is to develop and train a deep learning model to verify offline signatures with a reasonably high rate of accuracy and to compare the accuracy levels of writer-independent and writer-dependent models of verification. For the verification process, both models shall use the same set of image features. By comparing the two, we can lay claim as to which model gives us a higher level of accuracy. The false classification rates will also be measured so that future work can provide a model with higher accuracy rates.

Chapter 2  
LITERATURE SURVEY

1. Existing work on online signature verification.

McCabe [1] et al in 2008 created their own signature database to work on online signature verification. The main problem with working on handwritten signature verification is the lack of standardized datasets. The researchers used a XGT Serial Digitizing Tablet to capture their data. Obtaining skilled forgeries was also an obstacle to working with online signature verification, because forgers had difficulty reproducing an offline signature on an online device. This paper presented a method for verifying handwritten signatures by using a NN architecture. Various static (e.g., height, slant, etc.) and dynamic (e.g., velocity, pen tip pressure, etc.) signature features were extracted and used to train the NN. Several Network topologies were tested by the researchers and their accuracies were compared. The most successful version of the neural network based system used a single multilayer perceptron with one hidden layer to model each user’s signature. It was trained using five genuine signatures and one hundred zero-effort forgeries. This approach resulted in an optimal error rate of 3:3%.

1. Existing work on offline signature verification.

Offline verification uses pixel data obtained from scanned images and can be used for a variety of purposes. In contrast, online signatures can only be used for a limited set of tasks because it requires specialized input devices. Alvarez [2] et al used a Convolutional Neural Network to verify offline signatures. They built upon the work of Khalajzadeh [3] et al and used the ICDAR Database’s Dutch and Chinese Signatures to train and test their network. For testing the network, they used both genuine and questionable signatures. They experimented with several variations on signature verification tasks. They showed that convolutional neural networks do an excellent job of verifying signatures when allowed access during training to examples of genuine and forged signatures of the same people whose signatures are seen at test time. They then conducted an experiment where they tested their network on the signatures of new people whose signatures had not been seen at all during training, resulting in performance little better than a naive baseline due to the inherent difficulty of this task. Finally, they proposed a novel architecture for the comparison of signatures which had promise for future work in signature verification, specifically in situations where a possibly-forged signature can be compared to known genuine signatures of a specific signer.

They did not utilize traditional feature extraction methods based on image processing, but instead opted for using Convolutional Neural Networks to create the feature vector automatically. They trained one network on signatures from all identities together. For each language, all the data for that language was split into training, validation, and test data randomly, meaning validation and test data consisted of new copies of signatures of the same people whose signatures were in the training set. For both Chinese and Dutch, they put 80% of the data in the training set and 10% in each of the validation and test sets. They achieved a validation accuracy of 97% for the Dutch dataset and 95% for the Chinese dataset.

Choudhary [4] et al designed an offline signature verification device for small computational devices. Their proposed model extracted a variety of local and global features for the verification system. A combination of genuine and fraud signature images was fed into the proposed model and ANN was used to classify these images. A set of parameters: learning rate, Momentum, Epoch and Accuracy were monitored for different combination to arrive at the best combination for a given data set. Features used for this study were global, statistical and local. The entire design was developed on a JAVA based platform. Experimental results suggested that it was able to deliver 95% of accuracy.

Karounia [5] et all used another traditional feature based system, but it did not use deep learning. After training, the best classification accuracies were achieved. The classification ratio exceeded 93%, although the threshold, the parameter deciding the genuineness of an image, was 90%. The algorithm they supported used simple geometric features to characterize signatures that effectively served to classify signature as exact or forged.

Chapter 3  
PROPOSED MODEL

1. The Proposed Model

The proposed model will employ deep learning to train a neural network on signature features. The type of work done by the network is known as ‘classification’ where it will try to understand whether the test signature is genuine or forged. The classification will be done based on the extracted image features which will be fed to the network in the form of a feature vector. The dataset will be split into the training test and the test set. The training set will be used to train the neural network and the test set will be used to test the accuracy of the network. Standardized evaluation parameters shall be used to calculate the accuracy of the classification.

1. Model Description

The model will first pre-process all the images of the dataset. The image pre-processing procedure involves conversion of raw images into a standard form. The signature dataset has images of different dimensions and properties; this step will standardize those images for easier processing. The next step is feature extraction. Extraction of the attributes (features) of the system is known as feature extraction and these features determine the accuracy of the verification process. The extracted features will be stored in a feature vector. Features can be of two major categories: local and global. These features are formulated into a feature vector, which is then used to train the neural network. Two types of classification are performed: writer-dependent classification and writer-independent classification. The accuracy of each classification is evaluated using three parameters: Correct Classification Rate (CCR), False Acceptance Rate (FAR) and False Rejection Rate (FRR).

1. Image Preprocessing

The image pre-processing procedure involves conversion of raw images into a standard form. The signature dataset has images of different dimensions and properties; this step will standardize those images for easier processing. The first step in pre-processing involves opening the image in grayscale. Since a signature is being used here, it is not necessary that the colour pixels of the image be present for our training processes. Conversion to grayscale also reduces the size of the image being processed, reducing the time required for the later steps. The image is then resized to the dimension of 200x500. This value is a trade-off between size reduction and image quality preservation. The next step is to perform binarization of the image. The binarization step converts the pixel values of the images into either 0 or 1 based on a threshold value. Whenever a pixel value greater than the threshold is found, it is taken as 1, otherwise it is taken as 0. Here, 1 refers to the colour white and 0 refers to the colour black. Thus, the resulting image will have pixel values of either black or white, with-out any other colour. This makes the image easier to read for the future steps. The image will then go through a process of erosion, where the black pixels will be thinned as much as possible without making them invisible. This step makes the pixels uniform.

**Conversion to greyscale** - Grayscale images are smaller and easier to process than color images. Since the images being processed are signatures, the colour of ink is irrelevant for feature extraction. The images are scanned in grey using equation (1)

Grey= (0.299\*Red) + (0.5876\*Green) + (0.114\*Blue) [6]

**Resizing the image** - Resizing will reduce the size of the image and the pixel dimensions are selected as a trade-off between speed of processing and its accuracy. The dimensions chosen are 200x500.

**Binarization** - The subsequent image is then converted into binary form where the pixel values are either 0 or 1, where 0 is black and 1 is white-in accordance with the RGB color theory. The procedure applied here is binary thresholding. The procedure involves scanning all the pixels of the image and then comparing it with a threshold value T. Here T is chosen to be (225,255) based on experimentation with the image characteristics. The formula used here is equation (2):

**Erosion** - The image is then eroded. The procedure is analogous to the real life phenomenon of erosion in the sense that the boundaries of the foreground object are thinned out so as to thin the lines that were thickened due to difference in a pen’s ink flow. A kernel slides through the image and a pixel in the original image will only be considered 1 if all the pixels under the kernel is 1, otherwise it is taken as 0. The kernel size employed here is 2x2 and it is passed over 2 iterations. The image produced is therefore thinned.

A comparison between a raw image (Fig. 1) and the pre-processed image (Fig. 2) (difference in dimensions is excluded) is shown below:

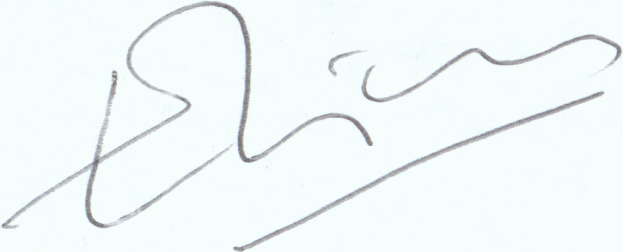


Figure 1 : Raw Image

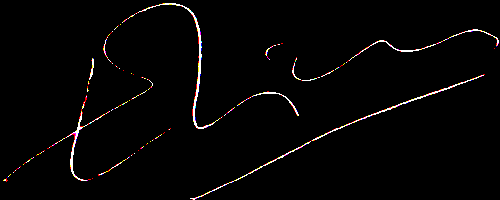


Figure 2 : Pre-processed Image

1. Feature Extraction

Extraction of the attributes (features) of the system is known as feature extraction and these features determine the accuracy of the verification process. The extracted features will be stored in a feature vector. Features can be of two major categories: local and global.

**Global Features** - Features such as width, length, etc. are examples of global features of an image. Global features describe the image in its totality, and are less susceptible to noise. They are also less susceptible to variation; hence they provide low accuracy for signature verification. In this model, we have used aspect ratio and base line slant angle.

**Local Features** - Local feature sets describe specific areas of the image in much greater detail. So, the information provided by local features is more useful and more precise than global features. However, computational cost of extracting these features is high. There are two categories of local features:

1. **Statistical Features** - Features which describe the pixel distribution of the image are known as statistical features. For this model, we have used Skewness and Entropy;
2. **Geometrical Features** - These features describe the geometric characteristics of the image and are less susceptible to distortion. For this model, we have used the center of gravity.
3. Features Used For The Model
4. **Global Features**
5. **Baseline Slant Angle** - Baseline is the imaginary line about which the signature is assumed to rest. The angle of inclination of this line to the horizontal is called the Slant Angle ϴ. To determine the slant angle the ratio of the maximum horizontal projection to the width of the projection is maximized over a range of values of angle of rotation Ф.
6. **Aspect Ratio** - The aspect ratio (A) is the ratio of width to height of the signature. The bounding box coordinates of the signature are determined and the width (Dx) and height (Dy) are computed using these coordinates as given by equation (3)

A = Dx/Dy

1. **Local Features**
2. **Statistical Features**
   1. **Skewness** - Statistically, skewness measures the asymmetry of a real-valued random variable's probability distribution. A skew value less than zero says that the tail on the left side of the probability density function is longer and the bulk of the values (possibly including the median) lie to the right of the mean. A skew value above zero says that the tail on the right side is longer and the bulk of the values lie to the left of the mean. A zero skew value indicates that the values are typically evenly distributed on both sides of the mean, usually but not necessarily implying a symmetric distribution. Mathematically skewness can be given by equation (4) [7]

A darker or glossier surface is typically more positively skewed than a lighter and matte surface. Hence, it is possible to use skewness in making judgments about image surfaces.

* 1. Entropy - Entropy measures the amount of randomness or uncertainty is present in an image. It acts as a measure of how much information is provided by the image. Entropy can be mathematically expressed as equation (5) [8]

1. Geometric Features
   1. Center of Gravity. Center of gravity (X, Y) given by equations (6) and (7)

X = j=0∑N-1 PV(j)\*j/Δ



Y = i=0∑M-1 PH(i)\*i/Δ

where PV and PH are the vertical and horizontal projections respectively. [9]

1. Dataset Description

The model uses the International Conference on Document Analysis and Recognition (ICDAR) 2011 Sig-Comp international signature verification competition Signature dataset. [10] This dataset consists of Chinese and Dutch signatures that are both online and offline. The offline dataset comprises PNG images, scanned at 400 dpi, RGB color. The dataset is split into a training set and testing set of non-overlapping IDs.

The testing set consists of two sections “referenced” and “questioned” set. The referenced set consists of known genuine signatures whereas the questioned set consists of both genuine and forged signatures. A sample of genuine and forged signature for the same id is shown in Figure 3.

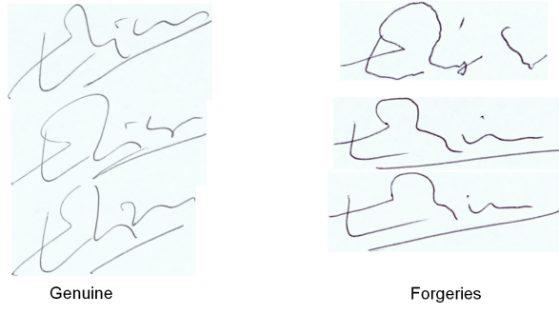


Figure 3 : Comparison between genuine and forged signature for same ID

1. Types of Classification
2. Writer dependent classification - Writer dependent classification involves training the dataset with another label which contains the id of the signer. Here the training is done in a way that will involve one id and several signature samples for the same id corresponding to a single signer. The signature samples consist of both forged and genuine signatures. During the testing phase, the id will be known to the network. The scope in this is limited since it is required that a sample be present for each signature in the system and it is assumed that forgeries of each class of signature is already available. However, since the problem space is smaller, the accuracy of predictions is higher in this approach.
3. Writer independent classification - Writer independent classification involves training the network with both genuine and forged signatures without taking into consideration the identity of signer. The classification is based on the network being able to differentiate between genuine and forged signatures based on previous training. This approach removes the requirement for an extra id and avoids the cost of labelling the data for an individual signer. This approach is universal since the identity of the signer is not required and the number of samples to be trained with is unconstrained. However, since the problem space is larger and since every signature has its own unique characteristics, the accuracy is significantly lower than writer dependent classification.
4. Evaluation Parameters

The model has been evaluated based on certain generalized evaluation parameters. The evaluation parameters are False Acceptance Rate (FAR), False Rejection Rate (FRR) and Correct Classification Rate (CCR). These parameters are computed using standard formulae

Each evaluation parameter was computed for every test size for every phase of iterations and were plotted. Some interesting trends were observed which have been detailed upon in the results section.

Chapter 4  
RESULTS AND IMPLEMENTATION

1. Neural Network Structure

Neural networks are a set of algorithms, modelled loosely after the human brain, that are designed to recognize patterns. They interpret sensory data through a kind of machine perception, labelling or clustering raw input. The patterns they recognize are numerical, contained in vectors, into which all real-world data, be it images, sound, text or time series, must be translated.

Neural networks help in clustering and classifying. They can be thought of as a clustering and classification layer on top of the data that is stored and managed. They help to group unlabelled data according to similarities among the example inputs, and they classify data when they have a labelled dataset to train on. Neural networks can also extract features that are fed to other algorithms for clustering and classification; so deep neural networks can be thought of as components of larger machine-learning applications involving algorithms for reinforcement learning, classification and regression.

The network has been trained with the feature vectors extracted from the images. The vector consists of individual columns for each feature and a ‘Genuine’ column which take values 0 or 1 depending on whether the signature is genuine or forged. This vector is then fed into the multilayer perceptron with proper ratios between forged and genuine signatures. The intention behind training this network is that after observing a certain number of genuine and forged signatures the network can figure out for itself when it next sees a signature’s features, that whether it is genuine or forged. Various training set and testing set ratios were used and their relative accuracies were studied.

The multilayer perceptron architecture used is described in this section. Multilayer perceptron (MLP) is a feedforward neural network with one or more layers between input and output layer. Feedforward means that data flows in one direction from input to output layer (forward). This type of network is trained with the backpropagation learning algorithm. An MLP consists of at least three layers of nodes. Except for the input nodes, each node is a neuron that uses a nonlinear activation function. MLP utilizes a supervised learning technique called backpropagation for training. Its multiple layers and non-linear activation distinguish MLP from a linear perceptron. MLPs are widely used for pattern classification, recognition, prediction and approximation. Multilayer Perceptron can solve problems which are not linearly separable.

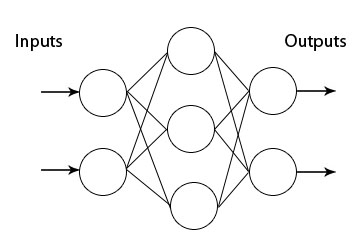


Figure : A general Multilayer Perceptron

The section will explain each term introduced and will define how it is implemented in the model.

**Skeletal Structure -** Skeletal structure refers to the basic underlying architecture of a network without taking into consideration its detailed features. The skeletal structure of the proposed network is a sequential structure which consists of a linear stack of layers. The layers used in this system are dense layers, meaning they are fully connected.

**Layers -** Layers in a multilayer perceptron refer to a collection of neurons in a single logical level of computation. There are three types of layers in any neural network-input layer, hidden layer and output layer. The input layer is the layer that is fed the inputs of the dataset. The hidden layers are the ones that are responsible for abstract-ing the type of input in the system. They are thus very important in classification problems. The output layer is the final layer in a linear stack which contains the out-put of the neural network. The layers in the proposed system will all be dense layers, meaning that they will be fully connected. For an n-classification problem, the output layer should in convention consist of n number of neurons.

**Input Layer -** The input layer consists of the layer that interacts with the input. There are three things to consider in this regard-the number of neurons in the layer, dimensionality of the layer and the activation function. The number of neurons used in this layer is chosen to be 16 while dimensionality while be number of features plus one, in this case it will be 6. Where the 1 is added for the label of ‘genuine’ or ‘forged’

The activation function chosen is ReLu (rectified linear unit) since it has the ad-vantage of being simple to compute and also due to its ability of not reaching saturation at high input values, meaning that their gradients don’t become 0 when the input value becomes very high.

***Number of hidden layers*** The universal approximation theorem provides the theory that neural networks with even a single hidden layer can be used to approximate any existing function when given the proper input parameters. Thus the inference is that a single hidden layer is able to solve a large variety of solvable problems.

Taking into consideration these factors, the proposed network has a single hidden layer. According to Heaton [11] a network with one hidden layer “can approximate any function that contains a continuous mapping from one finite space to another”. The reason behind choosing a single hidden layer is that the problem is a simple binary classification problem and hence complex representations are not required for such a problem. After sufficient experimentation it was found that a single hidden layer performed better than two hidden layers in terms of both speed and accuracy for the current problem.

Number of neurons in hidden layer. Choosing the number of neurons in a hidden layer is again crucial for training the network. Having less neurons makes it prone to underfitting while having too many neurons makes it prone to overfitting. Heaton proposed a rule of thumb for choosing the number of neurons. Following this rule of thumb, the values are chosen to be 16 for the first hidden layer.

***Activation function in hidden layer***. The activation function chosen for the hidden layers is ReLu for the same reason it was chosen for the input layer.

**Output Layer** - One neuron is used in the output layer since this is a binary classification problem. This is a convention and has neither been proved or disproved. The activation function in the output layer will be a sigmoid function which is since after the previous layers of abstraction the computable values will be reduced and a more accurate function can be used without getting affected by large computational values. The structure of the proposed neural network is shown below in Figure 5.

The structure shows fully connected layers with connections shown in only one layer (otherwise the diagram would look clunky). Each of the layers are shown with exactly the number of nodes each of them have. The final layer has a single node and provides the output of the network.

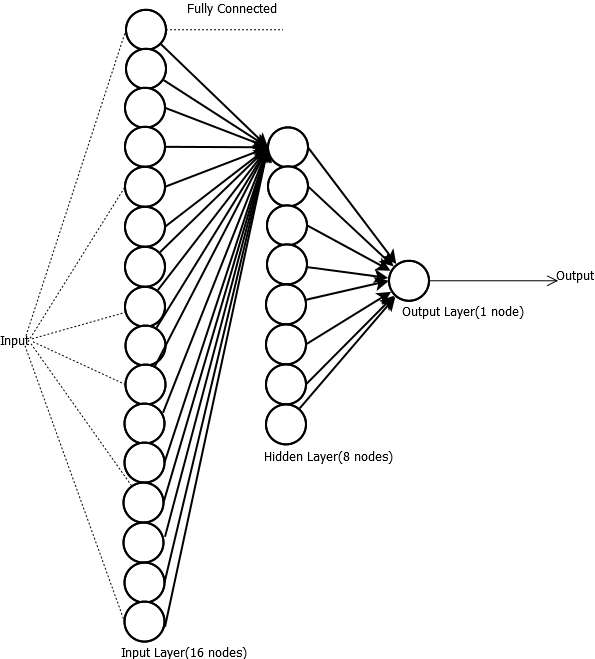


Figure 5 : Neural Network Structure (Multilayer Perceptron)

1. Writer Independent Mode Results

When the classification was performed in writer independent mode, the entire dataset was used, without taking into account the id of the signature. For this mode, all the features of all the signatures of the training set were fed to the network with labels indicating whether it is genuine or forged. The general idea of this mode was to show the network how a general genuine signature looks and how it differs from a general forged signature. A total of 2238 signatures were used for this purpose. A snapshot of the dataset structure is shown below: -

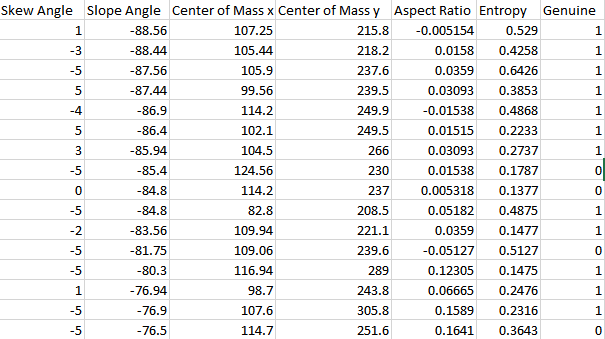


Figure 6:Dataset for Writer Independent Classification

Using this concept, several configurations of the neural network have been used and tested. The dataset was split into training set and testing set with varying ratios. Various values of test set size have also been tested. Separate number of iterations have also been used to test the accuracy of the network. The chart below cumulatively displays all the experimental results.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Set Size** | **CCR** | **FAR** | **FRR** | **No. of Iterations** |
| 0.25 | 68.82 | 38.06 | 18.18 | 100 |
| 0.25 | 67.95 | 41.01 | 15.15 | 105 |
| 0.25 | 68.82 | 34.71 | 23.78 | 110 |
| 0.25 | 67.6 | 34.45 | 28.1 | 115 |
| 0.3 | 67.49 | 34.57 | 28.38 | 100 |
| 0.3 | 67.78 | 27.52 | 42.85 | 105 |
| 0.3 | 67.2 | 33.54 | 31.22 | 110 |
| 0.3 | 70.99 | 27.82 | 31.73 | 115 |
| 0.35 | 67.5 | 35.22 | 27.2 | 100 |
| 0.35 | 71.25 | 32.9 | 19.76 | 105 |
| 0.35 | 66.5 | 33.65 | 33.21 | 110 |
| 0.35 | 69.125 | 38.33 | 16.48 | 115 |
| 0.4 | 67.5 | 36.66 | 24.09 | 100 |
| 0.4 | 67.5 | 41.5 | 15.28 | 105 |
| 0.4 | 67.61 | 31.77 | 33.67 | 110 |
| 0.4 | 66.84 | 30.69 | 38.3 | 115 |
| 0.45 | 68.09 | 33.87 | 28.08 | 100 |
| 0.45 | 65.85 | 33.09 | 36.3 | 105 |
| 0.45 | 65.75 | 35.32 | 32.06 | 110 |
| 0.45 | 68.28 | 35.06 | 24.77 | 115 |
| 0.5 | 68.56 | 32.42 | 29.41 | 100 |
| 0.5 | 66.9 | 37.28 | 25.31 | 105 |
| 0.5 | 66.63 | 36.82 | 26.38 | 110 |
| 0.5 | 71.27 | 34.32 | 17.02 | 115 |

Figure 7 Writer Independent Mode Data

From the data above it can be inferred that in writer independent classification, the number of iterations significantly affects the accuracy values for the same test set size. The highest accuracy value or CCR (Correct Classification Rate) (the terms accuracy and CCR will be used interchangeably from now onwards) within this mode of classification is surprisingly with a test size of 0.5 with an iteration value of 115 at 71.27%. The result is slightly surprising because within the same test size, increasing the number of iterations have reduced the accuracy, but when it is 115, the value jumps 3 % higher than the previous highest within the same test size. This increase in accuracy is high due to the significant reduction in the false rejection rate values. The result can be partially explained by the reduced training set size, which had more genuine signatures, as a result of which the extra number of iterations improved the performance.

The best general performance is associated with a test size of 0.35, which is acceptable since an optimal test size helps in improving the performance of a system. The highest accuracy here is at an iteration size of 105 at 71.25 %, which is only marginally less than the global highest of 71.27 %. Moreover, the highest accuracy was reproducible when the training and test sets where randomly shuffled.

The reason for the relatively low accuracy of writer independent classification stems from a very simple concept. Signatures vary significantly between individuals and it is not possible for a system to naively classify a signature as genuine or forged based on how a general signature looks. There is not much scope for generalization within a signature since it is a property unique to each person. Considering these aspects, a reproducible highest accuracy above 70 % is still acceptable because of the difficulty of the problem itself. Detailed relationship between the individual evaluation parameters and the test size and number of iterations are shown below.

Figure 8CCR with Test Size Vs Number of Iteration

As seen here and as discussed above as well, the accuracy peaks at 71.27 % and 71.25 % at separate points. The general trend as observed suggests that accuracy levels peak at a certain test set size and decreases gradually with a sudden, final increase at a test size of 0.5. The trends implied here for an iteration size of 100 have been represented in Figure 4 below.

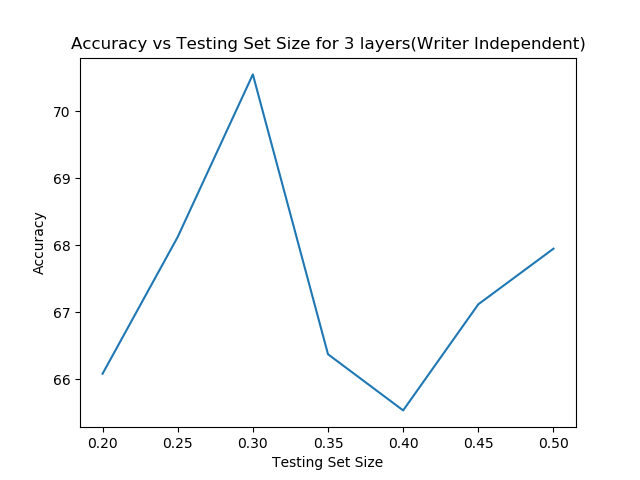


Figure 9 : Accuracy Vs Testing Set Size (100 iterations)

Figure 10 : FAR with Test Size Vs No. of iterations

The false acceptance rate has very high values when the number of iterations is 105. The values are somewhat lesser with number of iterations at 110. The value dips significantly at a test set size of 0.3 and number of iterations at 115. High FAR values are implicative of high leniency of the system where the system accepts forged signatures as genuine. Writer independent classification does not provide good performance in this regard.

Figure 11 : FRR with Test Size Vs No. of iterations

The false rejection rate values are uniformly low at 100 iterations and is uniformly high at 110 iterations. It peaks significantly at 105 iterations and a test size 0f 0.30. The lowest value is an acceptable one of 15 % at two points in test size of 0.25 and 105 iterations. High FRR values imply a stricter system which even classifies a genuine signature if it has some traits of a general forged signature. From the data, we can easily infer that the system performs slightly better in this regard when compared to FAR.

1. Writer Dependent Mode Results

When the classification was performed in writer dependent mode, the entire dataset was split by ids, taking into account the id of the signature. For this mode, all the features of all the signatures of the training set were fed to the network with labels indicating whether it is genuine or forged. Another label for ids was used. The general idea of this mode was to show the network how a genuine signature for a particular id looks and how it differs from a forged signature of the same person. A total of 2238 signatures were used for this purpose. A snapshot of the dataset structure for ID 1 is shown below: -

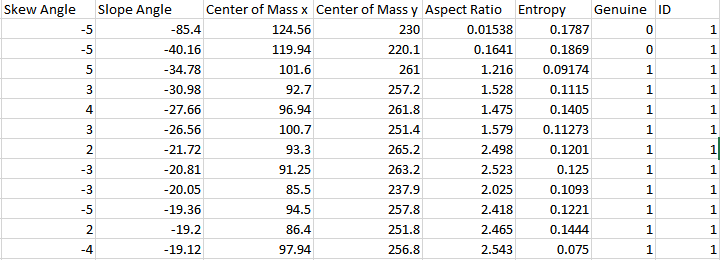


Figure 12 : Dataset for Writer Dependent Classification

Using this concept, several configurations of the neural network have been used and tested. The dataset was split into training set and testing set with varying ratios. Various values of test set size have also been tested. Separate number of iterations have also been used to test the accuracy of the network. The method in which writer dependent mode has been implemented is that for each ID, the network is fed a combination of genuine and forged signatures and trained. Then, a mix of genuine and forged signatures of the same ID are tested with it and the accuracy for that particular ID is noted. A system like this would be very useful in a banking structure since several genuine signature samples of each account holder is always available. Average accuracies of all IDs are then tabulated for each configuration. The chart below cumulatively displays all the experimental results.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Set Size** | **CCR** | **FRR** | **FAR** | **No. of Iterations** |
| 0.25 | 82.05 | 15.6 | 19.61 | 100 |
| 0.25 | 82.05 | 15.6 | 19.61 | 105 |
| 0.25 | 82.23 | 16.06 | 19.05 | 110 |
| 0.25 | 82.41 | 16.51 | 18.49 | 115 |
| 0.25 | 81.83 | 16.97 | 19.05 | 120 |
| 0.3 | 81.48 | 18.99 | 18.39 | 100 |
| 0.3 | 81.15 | 20.16 | 18.16 | 105 |
| 0.3 | 81.28 | 20.16 | 17.94 | 110 |
| 0.3 | 81.27 | 20.16 | 17.94 | 115 |
| 0.3 | 81.14 | 20.16 | 18.16 | 120 |
| 0.35 | 81.43 | 24.63 | 15.69 | 100 |
| 0.35 | 80.94 | 26.1 | 15.69 | 105 |
| 0.35 | 81.03 | 25.37 | 15.86 | 110 |
| 0.35 | 81.12 | 26.47 | 15.15 | 115 |
| 0.35 | 81.27 | 25.37 | 15.51 | 120 |
| 0.4 | 81.93 | 26.87 | 14.15 | 100 |
| 0.4 | 82.13 | 27.21 | 13.69 | 105 |
| 0.4 | 82.1 | 27.89 | 13.38 | 110 |
| 0.4 | 82.18 | 27.55 | 13.38 | 115 |
| 0.4 | 82.01 | 28.57 | 13.08 | 120 |
| 0.45 | 80.58 | 23.97 | 17.21 | 100 |
| 0.45 | 80.82 | 23.69 | 16.93 | 105 |
| 0.45 | 81.1 | 23.42 | 16.64 | 110 |
| 0.45 | 80.8 | 23.42 | 17.07 | 115 |
| 0.45 | 80.58 | 24.24 | 16.93 | 120 |
| 0.5 | 79.39 | 22.45 | 19.92 | 100 |
| 0.5 | 79.4 | 22.7 | 19.79 | 105 |
| 0.5 | 79.46 | 22.96 | 19.53 | 110 |
| 0.5 | 79.6 | 22.7 | 19.39 | 115 |
| 0.5 | 79.19 | 23.72 | 19.53 | 120 |

Figure 13 : Writer Dependent Mode Data

From the data above it can be inferred that in writer dependent classification, the number of iterations does not affect the accuracy values for the same test set size very significantly. The highest accuracy value within this mode of classification is surprisingly with a test size of 0.25 with an iteration value of 115 at 82.41 %. The result is not surprising because within the same test size, increasing the number of iterations have reduced the accuracy and 115 is the optimal value for the number of iterations. The best general performance is also associated with a test size of 0.25, with accuracy values within the range of 81-82 % for all the number of iterations which is acceptable since an optimal test size helps in improving the performance of a system. Detailed relationship between the individual evaluation parameters and the test size and number of iterations are shown below.

Figure 14 : CCR with Test Size Vs Number of Iteration

As seen here and as discussed above as well, the accuracy peaks at 82.41 %. The general trend as observed suggests that accuracy levels peak at a certain test set size and decreases gradually with a sudden, final increase at a test size of 0.5. The trends implied here for an iteration size of 100 have been represented in Figure 9 below.

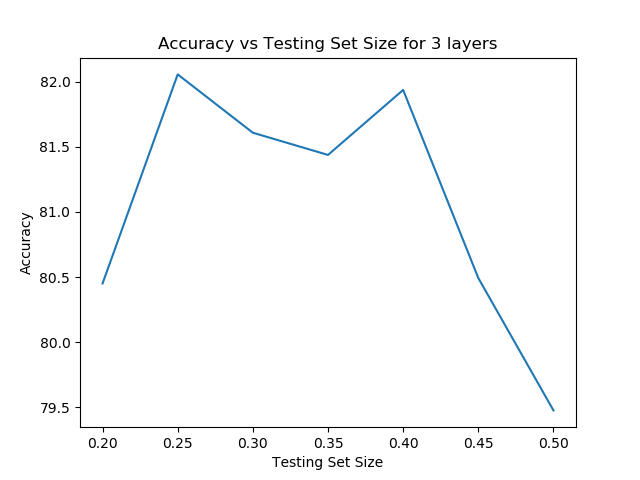


Figure 15 : Accuracy vs Testing Set Size (100 iterations)

The reason for the relatively higher accuracy of writer dependent classification stems from the idea that in this case, the problem space is smaller and the testing is constrained by the fact that genuine signature feature values will be relatively close to each other while forged signatures are further apart. When observing the accuracies of individual IDs, it was found that many of the individual IDs had accuracies as high as 100 %, while some (very few) had accuracies as low as 66 %, which brought down the total average accuracy of the system. The reason behind this relative difference in performance for separate signatures can be attributed to how the features extracted define each signature. For the cases with high accuracy, the features could fully define the signatures, while for lower values of accuracy the features could only partially define the signature.

Figure 16 : FAR with Test Size Vs Number of Iteration

The false acceptance rate value trends very uniformly. When the test size is high, for the same iteration number, the FAR value increases. The system is relatively more tolerant to false acceptance than the writer independent system. High FAR values are implicative of high leniency of the system where the system accepts forged signatures as genuine. Writer dependent classification provides acceptable performance in this regard because FAR peaks at 19.92 %, which is only present for a high testing set size.

Figure 17: FRR with Test Size Vs Number of Iteration

The false rejection rate values are uniformly low at low test sizes and peaks at a test size of 0.35 and then drops again. The trend is same for all iteration sizes. The lowest value is an acceptable one of 13 % at test size of 0.4 and 105 iterations. High FRR values imply a stricter system which even classifies a genuine signature if it has some traits of a general forged signature. From the data, we can easily imply that the system performs similarly in this regard when compared to FAR.

Chapter 5  
CONCLUSION AND FUTURE SCOPE

1. Conclusion

We can thus safely conclude that the writer-dependent classification shows a significantly better accuracy rate than the writer-independent model. However, the problem of false acceptance of forgeries and false rejection of authentic signatures still persists.

Writer dependent mode implemented with pre-known IDs would be very useful in a banking system where processing of cheques can be performed in an automated manner. The pre-requisite for such a system to work with high accuracy is to have more sample signatures per individual and to train the system for each ID with such samples. Once the system is trained with more data, it will be easier for the system to determine whether a signature for the particular person is genuine or forged. Writer dependent mode also showed promise in many cases where the accuracy was as high as 92-100 % for a significant number of IDs. Further research on these aspects can produce satisfactory results since knowing how the features extracted affect the system and how it can be improved will be a major driving force for increasing the accuracy of the system.

One of the major challenges of working on signature verification is the lack of standardized algorithms. Another major challenge that was encountered was the lack of standardized datasets. Because standardized input cannot be derived from any ordinary input device, the project had to rely solely on the limited data of the 2011 Sig-comp dataset. Because signature features vary depending on nationality, the project used only the Dutch signatures since they were more in number.

1. Future Work

Future work on this project would be to include more features for training and testing the neural network. One notable feature would the effect of pressure variance on the signature. [12]

The model could also be trained to work on signatures of Indian origin, which would have their own unique features. A standardized dataset would have to be built from scratch for that purpose which would require input devices. The lack of any preceding work on the field of Indian signatures makes this sphere of work a very promising one indeed.

Chapter 6  
REFERENCES

1. McCabe, Alan, Trevathan, Jarrod, Read, Wayne (2008) "Neural Network-based Handwrit-ten Signature Verification" DOI-https://doi.org/10.4304/jcp.3.8.9-22
2. Gabe Alvarez, Blue Sheffer, Morgan Bryant, "Offline Signature Verification with Convolutional Neural Networks
3. H. Khalajzadeh, M. Mansouri, and M. Teshnehlab, Persian Signature Verification using Convolutional Neural Networks, in International Journal of Engineering Research and Tech-nology, vol. 1. ESRSA Publications, 2012.
4. Savita Choudhary Siddanth Kaul Shridhar Mishra Dr. Arun J.B "Signature Verification Using Java – Python for Small Computational Devices"
5. Offline signature recognition using neural networks approach Ali Karounia, Bassam Dayab, Samia Bahlakba, Procedia Computer Science 3 (2011) 155–161
6. Savita Choudhary Siddanth Kaul Shridhar Mishra Dr. Arun J.B "Signature Verification Using Java – Python for Small Computational Devices"
7. Importance of Statistical Measures in Digital Image Processing Vijay Kumar, Priyanka Gup-ta IJETAE ISSN 2250-2459, Volume 2, Issue 8, August 2012)
8. Information Entropy Measure for Evaluation of Image Quality Du-Yih Tsai, Yongbum Lee and Eri Matsuyama Published: June 19, 2007 DOI:10.1007/s10278-007-9044-5
9. Prasad A.G. Amaresh V.M. “An offline signature verification system”..
10. Marcus Liwicki, Michael Blumenstein, Elisa van den Heuvel, Charles E.H. Berger, Reinoud D. Stoel, Bryan Found, Xiaohong Chen, Muhammad Imran Malik. "SigComp11: Signature Verification Competition for On- and Offline Skilled Forgeries", Proc. 11th Int. Conference on Document Analysis and Recognition, 2011
11. Heaton Research “The Number of Hidden Layers” [http://www.heatonresearch.com/2017/06/01/hidden-layers.html Last Checked-23/04/2018](http://www.heatonresearch.com/2017/06/01/hidden-layers.html%20Last%20Checked-23/04/2018)
12. A Dutta, P Bhowmick, S Ghosh, TK Hazra, BB Mallik "Extraction of Handwriting features on Pressure Variance"