Offline Signature Verification using Feature Point Extraction

S.N. Gunjal
Computer Engg. Dept
SRES's College of Engineering,
Kopargaon-423603.
Maharashtra (India)

B.J. Dange
Computer Engg. Dept
SRES's College of Engineering,
Kopargaon-423603.
Maharashtra (India)

A.V. Brahmane
Computer Engg. Dept
SRES's College of Engineering,
Kopargaon-423603.
Maharashtra (India)

ABSTRACT

Signature verification is one of the most widely used biometrics for authentication. The objective of the signature verification system is to discriminate between two classes: the original and the forgery, which are related to intrapersonal and interpersonal variability. Firstly, there exists great variation even between two signatures of the same person. They never start from the same position and neither do they terminate at the same position. Also, the angle of inclination of the signatures, the relative spacing between letters of the signatures, height of letters, all vary even for the same person. Hence it becomes a challenging task to compare between two signatures of the same person. The proposed an offline signature verification system to take care of that, which is based on depth for segmentation of signature image into different parts, after that geometric center of the each segment is find out as the feature point of that segment. The number of feature points extracted from signature image is equivalent to the number segment of the signature image that is produce by specifying value of depth. The classification of the feature points utilizes two statistical parameters like mean and variance. Our proposed model has three stages: image pre-processing, feature point's extraction and classification & verification. The user introduces into the computer through scanned signature images, our technique modifies their quality by image enhancement and noise reduction techniques, to be followed by feature extraction and finally used Euclidean distance model to classification of signature either genuine or forgery. The proposed offline signature verification system used "GPDS360 signature database".

General Terms

Euclidean distance model, signature verification and recognition,

Keywords

Offline signature verification, geometric centre, feature point, forgeries, FAR(False Acceptance Rate), FRR (False Rejection Rate), CCR (Correct Classification Rate), image processing.

1. INTRODUCTION

Signatures are composed of special characters and flourishes and therefore most of the time they can be unreadable. Also intrapersonal variations and interpersonal differences make it necessary to analyze them as complete images and not as letters and words put together [1]. As signatures are the primary mechanism both for authentication and authorization in legal transactions, the need for research in efficient automated solutions for signature recognition and verification has increased in recent years. Recognition is the identification of the signature owner. Verification is the decision about whether the signature is genuine or forgery [9]. In this decision phase the forgery images can be classified in three groups:

(i) Random Forgery (ii) Unskilled Forgery (iii) Skilled Forgery [2].

(1) Random Forgery: This is also known as simple forgery and is very easy to detect. The signer creates a signature in his own style by just knowing the name of an individual whose sign is to be made.

(2) Unskilled Forgery: The signer creates a signature after observing the signature once or twice without any prior experience.

(3) Skilled Forgery: The signer may be a professional in copying signatures. He creates a signature after having a good practice over it. Such signatures are most difficult to detect [1].

It is due to the fact that biometric characteristics of every person are unique and cannot be lost, stolen or broken. There are two types of biometrics: Behavioral and Physiological. Handwriting, speech etc. come under behavioral biometrics. Iris pattern, fingerprint etc. are part of physiological biometrics. Distinguish in two different categories of verification systems depending on acquisition of signatures: online signature verification, for which the signature is captured during the writing process and making the dynamic information available, and offline, for which the signature is captured once the writing process is over and thus, only static information is available. The objective of the signature verification system is to discriminate between two signature classes: the genuine and forgery signature is called Inter Personal Variation [2]. A lot of work has been done in the field of off-line signature verification [6], [7], [8]. Although a large number of works is focused on random and simple forgery detection, more efforts are still needed to address the problem of skilled forgery detection [3].

The proposed offline signature system is automated method for recognition and verification of signature by extracting feature points that characterizes the signature. This approach starts by scanning images into the computer, then modifying their quality through image enhancement and noise reduction, followed by feature point's extraction, threshold selection and finally verifies whether a signature is original or fake.

In this paper, all experiments have been performed on skilled forgeries. The paper is organized as: Section-2 describes signature database, Section-3 describes the proposed idea of this paper which includes signature acquisition, signature preprocessing, feature point's extraction and verification technique. In Section IV, experimental results have been presented including the results of the proposed technique and comparison with existing techniques. Section V concludes the technique discussed in the paper.

2. SIGNATURE DATABABE

This research has been conducted using GPDS360 signature database" offered for those doing research in the field of signature recognition at the *Universidad de Las Palmas de Gran Canaria,* Spain. The database contains data from 300 individuals, including 24 genuine signatures from individuals, and 30 forgeries. The 24 genuine specimens were collected in a single day of writing sessions. The forgeries were produced from the static images of the genuine signatures. Each forger was allowed to practice the signature for as long as she/he wished. Each forger imitated 3 signatures of 5 signers in a single day writing session. The genuine signatures were shown to each forger and were chosen randomly from the 24 genuine ones. Therefore, for each genuine signature, 30 skilled forgeries were produced by 10 forgers, from 10 different genuine specimens. Access to the mentioned database was authorized after a document of compliance was signed, that ensures the use of these signatures for research and academic work only, and not for commercial use [7].

3. PROPOSED SYSTEM

From previous studies, it has been observed that an Offline signature verification process consists of following
Steps:

1. Signature Acquisition
2. Signature Preprocessing
3. Feature Point's Extraction
4. Signature Verification

3.1 Signature Acquisition

Normally a scanner scans a page as an image. Here the problem is to identify the signature from a scanned image. It is difficult because there can be many other text and patterns in that scanned image. Hence it is required to set a signature area, which will help to identify the exact boundary of the signature in that scanned image. If the signature area had been identified (where users endorse their signature) bounded with a border color (like RGB-239, 228, 176) then it would help to identify the exact position of that signature in that scanned image. The image containing the user signature has been scanned using the following algorithm 3.2.1:. Fig.1 shows some sample signatures from database on which proposed technique has been tested.

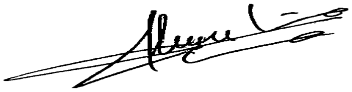


Fig 1: Original signature

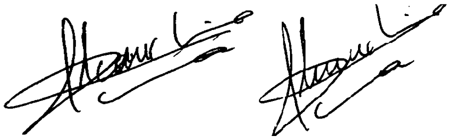


Fig 2: Forgery signature

3.2 Signature Preprocessing

To verify a signature correctly, preprocessing of acquired signature is required. The acquired signature image as shown in Fig.1 may sometimes contain noise (extra pen dots other than signature). It is necessary to remove these extra pixels from acquired image to verify the signature correctly. This can be done by using median filters. Following are the preprocessing steps:

Algorithm to calculate the rectangular signature area:

```
Step.1 Set st = 0, sl = 0, sh = 0, sw = 0 (sl is signature left
position an st is top sh is Height and sw is width)
Step.2 Set x = 0
Step.3 Set y = 0
Step.4 Scan the color from the scanned image and store it in
pixel color
Step.5 Check if pixel color = RGB-239,228,176 then goto
Step 6 else goto Step 8
Step.6 Check if sw = 0 then Set sl = x AND if sh = 0 then Set
st = y
Step.7 Set sw = x; sh = y; y = y + 1
Step.8 Check if y < image.height - 1 then Repeat from Step 4
to Step 7
Step.9 Set x = x +1
Step.10 Check if x < image.width - 1 then Repeat from Step 3
to Step 9
Step.11 End
```

3.2.1 Removing noise and normalize the color

Color and noise removal is very important because a signature may compose of many colors and may be affected by noises after scanning. Hence we have to eliminate all the noise present across the signature regional area to get the exact signature. After noise elimination the image had been converted to a black & white image. This can stored in the database as a sample signature or can be used to compare with a sample database signature. The advantage is that it decreases the size of the image and it is required to compare only two colors. In order to solve this problem the following methods have been proposed.

3.2.1.1 Color normalization method

To make the rectangular area black and white we need to scan all the rectangular area using the below algorithm and find the color of each pixel. If the color is (RGB (239,228,176)) then converts white and if the color is white then there is no conversation. Otherwise change the pixel color to black. This is so easy and fast technique to make a black and white picture. The technique of this is given below as algorithm.

- **Algorithm to make the image black and white:**

```
Step.1 Scan the color from the scanned image and store it in
pixelcolor

Step.2 If pixelcolor = color (rgb (239,228,176)) then Set
pixelcolor = color (white)

Step.3 If pixelcolor = color (white) then goto Step 4 else Set
pixelcolor = color (black)

Step.4 Repeat from Step 1 to Step 3 while image not scanned
completely

Step.5 End
```

3.2.2 Noise resolution method

In the next step of the proposed work the noise has to be removed. In this phase the noise associated with the image which has been created by the scanner while the image has been scanned need to be removed. After scanning that signature and normalizing its color some small single pixels of black color has been found, which is not the part of that signature. Following method has been introduced to remove the noise in the image.

- **Algorithm to remove noise**

```
Step.1 Scan the color from the scanned image and store it in pixel color
Step.2 If pixel color = color (black) then go to Step 3 else go to step 4
Step.3 If pixel color is not same wit adjoin pixel color then set pixel color = color (white) and go to Step 1 else go to  step 4
Step.4 Repeat step 1 to step 3 while image not scanned completely
Step.5 End
```

3.2.3 Adjust its property

After the previous step a binary image (only black and white) is obtained. After that it is required to locate the exact position of the signature in the image to perform the signature verification, because signature can be anywhere inside this rectangular area. And it can sign in different angles and sizes.

Now the first problem is to find the exact position of the signature from the rectangular boundary area and the second problem is to find the angle and size of the signature. After that the necessary correction need to be done in the context mentioned above.

3.2.3.1 Finding the exact position of the signature in the signature box

To solve the first problem a solution has been proposed. The proposed solution is based on the identification of edges of the signature in the signature box. It scan the rectangular area using the above algorithm 3.2 from its four (i.e. top, bottom, left, right) sides. And after that the actual signature area has been extracted.

3.2.3.2 Angular problem solutions

Another important task is that the angular detection of a user signature can change from time to time. A signature can be written in different angles. To compare the signature signed in different angles that is stored in database. The main goal is to change the property of scanned signature such that it can be compared with the sample database signature. Sometime same signature can be written in an angle and to solve this angular problem some mathematical formula is used that has been described below.

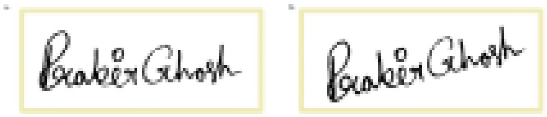


Fig 3 : Original Signature b) Sample Signature

We have used co-ordinate geometry to find the angle and to rotate the image accordingly. An angular signature which can be described by Fig.4 can be considered. In this picture two points ‘a’ and ‘b’ are the two end points of the signature.

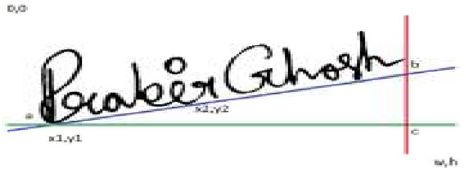


Fig 4 : An Angular Signature

Now we get x1, y1 co-ordinates from point ‘a’ and x2, y2 co-ordinates from point ‘b’. If we draw a straight line to ‘w’ and ‘h’ from point ‘a’ and ‘b’ respectively then we can see that the two lines from ‘a’ and ‘b’ cuts each other at point ‘c’.And now from the co-ordinate geometry, the angle is

$$\tan\theta = \frac{changes\ of\ Y}{changes\ of\ X}$$

$$\therefore\ \theta = \tan^{-1}\frac{changes\ of\ Y}{changes\ of\ X}$$

Change of Y =Y2-Y1,Change of X =X2-X1

In order to measure the angle properly the angle θ has been scaled 100 times. Then the image has been rotated by the angle θ to make it horizontal to the x axis.

3.2.4 Re-sizing image with database image

After rotation the signature as the signature size gets enlarged hence re-sizing of the signature is required before the two signatures get compared.

3.2.5 Threshold & Binarization

In computer vision and image processing, Otsu’s method is used to automatically perform histogram shape-based image thresholding or the reduction of a gray level image to a binary image. The algorithm assumes that the image to be threshold contains two classes of pixels (e.g. foreground and background) then calculates the optimum threshold separating those two classes so that their combined spread (intra-class variance) is minimal. Binarization means black and white version of the input (RGB) signature [11].

Thinning: Thinning is basically a morphological operation which is applied to binary image to obtain one pixel run of signature or skeleton of a signature shown in Fig 5.

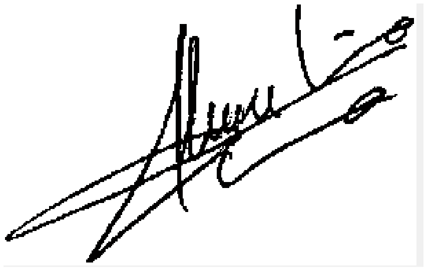


Fig 5: Original Signature image

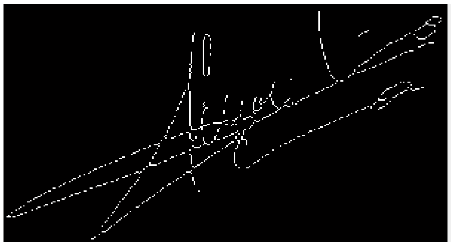


Fig 6: Binarization, thinning, resize operation

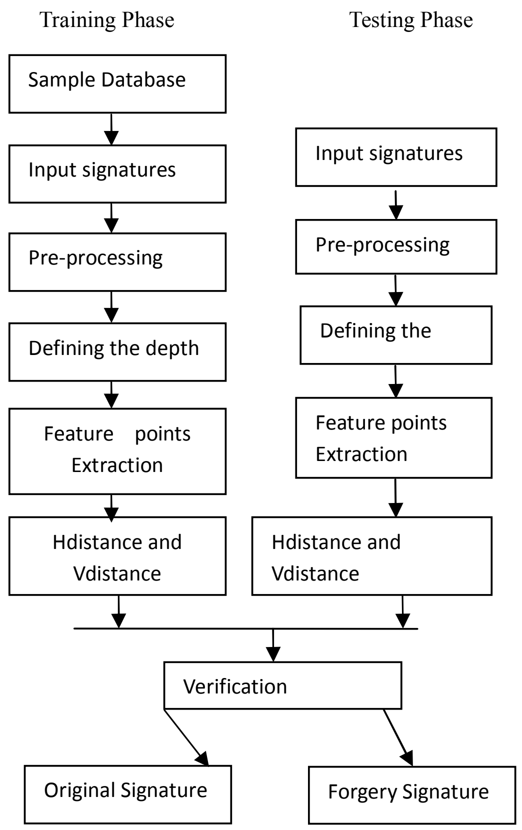


Fig 7: Flow Diagram of Offline Signature verification

3.3 Feature Point's Extraction

In feature extraction step, the necessary features points are extracted from the sample preprocessed signature. The numbers of feature point's to be extracted are based on the application and vary from system to system. The number of feature point's selection is depends upon the depth as per value of depth the number of feature point get selected.

Determine the number of feature points: In proposed method extraction of total number feature points from the signature is based upon the depth as value of depth changes the total number of feature point's changes. The more number of feature points extraction improve the accuracy of forgery signature detection.

a) Depth=1(4 feature points)

b) Depth=2(4fp+4*2fp=12fp)

c) Depth=3 (4fp+4*2fp+8*2fp=28fp)

d) Depth=4(4fp+4*2fp+8*2fp+16*2 fp=60fp)

e) Depth=5(4fp+4*2fp+8*2fp+16*2fp+32*2fp =124fp)

The geometric features are based on two sets of points in 2-dimensional plane. If depth is specified as 4 the vertical splitting of the image results thirty feature points (v1, v2,v3,…,v30) and the horizontal splitting results thirty feature points (h1, h2, h3,.…,h30).These feature points are obtained with relative to a central geometric point of the image. Here the centered image is scanned from left to right and calculate the total number of black pixels. Then again from top to bottom and calculate the total number of black pixels. Then divide the image into two halves w.r.t. the number of black pixels by two lines vertically and horizontally which intersects at a point called the geometric centre. With reference to this point 60 features are extracted.30 vertical and 30 horizontal feature points of each signature image.

Processing of the Signature: The geometric feature points of proposed system based on two sets of points in two dimensional planes. Each set having thirty feature points which represent the stroke distribution of signature pixels in image. These 60 feature points are calculated by considering the geometric center of each segment. Vertical Splitting and Horizontal Splitting are two main steps to retrieve these feature points.

3.3.1 Feature points based on vertical Splitting:

Thirty feature points are obtained based on vertical splitting w.r.t. the central feature point. The procedure for finding vertical feature points are given below
Algorithm:
Input: Static signature image, depth=4.
Output**:** Vertical feature points: v1, v2, v3, v4 …v29,V30.
The steps are:

1. Input the value of depth =4
2. Split the image with a vertical line passing through the geometric centre (v0) which divides the image into two halves: Left part and Right part.
3. Find geometric centers v1 and v2 for left and right parts correspondingly.
4. Split the left and right part with horizontal lines through v1 and v2 to divide the two parts into four parts: Top-left, Bottom-left and Top-right, Bottom right parts from which we obtain v3, v4 and v5, v6.
5. We again split each part of the image through their geometric centers to obtain feature points v7, v8, v9….v13, v14.
6. Then we split each of the parts once again to obtain all the thirty vertical feature point's v15, v16, v17, …,v29, v30.

3.3.2 Feature points based on Horizontal Splitting

Thirty feature points are obtained based on horizontal splitting w.r.t. the central feature point. The procedures for finding horizontal feature points are given below.

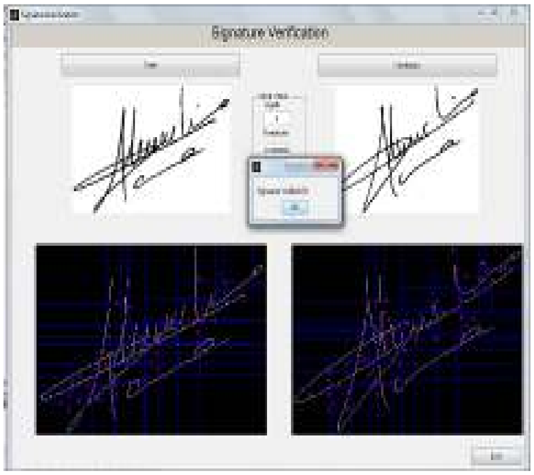
Algorithm:
Input: Preprocessed signature image, depth=4.
Output**:** Horizontal feature points: h1, h2, h3,h4….h29, h30.
The steps are:

1. Input the value of depth =4
2. Split the image with a horizontal line passing through the geometric centre (h0) which divides the image into two halves: Top part and Bottom part.
3. Find geometric centers h1 and h2 for top and bottom parts correspondingly.
4. Split the top and bottom part with vertical lines through h1 and h2 to divide the two parts into four parts: Left-top, Right-top and Left-bottom, Right bottom parts from which we obtain h3, h4 and h5, h6. We again split each part of the image through their geometric centers to obtain feature points h7, h8, h9,……, h13, h14.

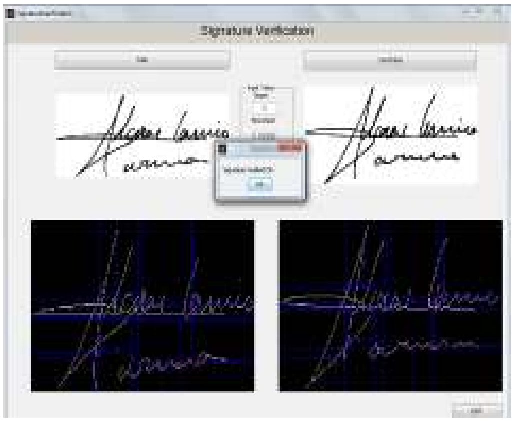
5. Then we split each of the parts once again to obtain all the thirty horizontal feature points h15, h16, h17,………, h29, h30.

Observations-:

a) **Experiment 1: Depth Level -4 & Threshold Value=0.2**



b) **Experiment 2: Depth level =3 & ThresholdValue=0.2**



4. SIGNATURE VERIFICATION

In this study features are based on geometric properties. So we use Euclidean distance model for classification [12]. This is the simple distance between a pair of vectors of size n. Here vectors are nothing but feature points, How to calculate distance between the training and testing sample using Euclidean distance model is described in the following Section.

Euclidean distance model:

Let A (p_1, p_2…..p_n) and B(q_1,q_2….q_n) are two vectors. In threshold calculation these distances are useful.

$$d(p,q) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2} \text{.....................(1)}$$

In our application, vectors are the feature points on plane. So d is simple distance between points. Here p_1,p_2,p_3,…,p_n are the mean feature points of the multiple training signatures and q_1,q_2,q_3,…….q_n are the feature points of the test signature.

Let n signatures are training signature from each person. There are 60 feature points from each original signature, 30 are taken by vertical splitting and 30 are taken by horizontal splitting. Mean vertical feature points based on vertical splitting is calculated as shown below:

$V_{mean, 1}$ = mean ($v_{1,1}$, $v_{2,1}$,..., $v_{n,1}$)

$V_{mean, 2}$ = mean ($v_{1,2}$, $v_{2,2}$,..., $v_{n,2}$)

$V_{mean, 3}$ = mean ($v_{1,3}$, $v_{2,3}$,..., $v_{n,3}$)

$V_{mean, 4}$ = mean ($v_{1,4}$, $v_{2,4}$,..., $v_{n,4}$)

....

……………………………………

$V_{mean,30}$ = mean($v_{1,30}$,$v_{2,30}$,...,$v_{n,30}$)

Where $v_{i,1}$,$v_{i,2}$,……,$v_{i,14}$ are vertical splitting features points of i^{th} training signature sample. Now the same procedure applied to calculate the mean horizontal feature points using the horizontal splitting.

$H_{mean,1}$ = mean($h_{1,1}$, $h_{2,1}$,..., $h_{n,1}$)

$H_{mean,2}$ = mean($h_{1,2}$, $h_{2,2}$,..., $h_{n,2}$)

$H_{mean,3}$ = mean($h_{1,3}$, $h_{2,3}$,..., $h_{n,3}$)

$H_{mean,4}$ = mean($h_{1,4}$, $h_{2,4}$,..., $h_{n,4}$)

....

$H_{mean,14}$ = mean($h_{1,30}$, $h_{2,30}$,..., $h_{n,30}$)

Where, $h_{i,1}$, $h_{i,2}$,……, $h_{i,14}$ are horizontal splitting features of i^{th} training signature sample. Feature points based on vertical splitting are v_1,v_2,v_3,v_4,…v_{14}, v_{28} and feature points based horizontal splitting are h_1,h_2,h_3,h_4,…… h_{13}, h_{14}.

Distance between mean feature points of training signatures and test signature features points are shown below:

$$\text{Distance} = \sum_{i=1}^{14} (\text{Hmean(i)-H(i)})^2 + (\text{Vmean(i)-V(i)})^2 \text{...} \quad (2)$$

Depending upon the value of Distance, identification of signature is performed as original or forgery.

5. RESULT

To train the system, a subset of GPDS database was taken comprising of 5 genuine samples taken from each of the 5 different individuals and to test the system 10 genuine samples taken from each of the 5 different individuals. The following table 1 shows the result of system in the form of Correct Classification Rate (CCR) and False Rejection Rate

(FRR) for original signatures. The CCR and FRR calculated as.

$$\text{FRR}=\frac{\text{Number of original signature rejected}}{\text{Number of original signature tested}} \text{..................................(4)}$$

$$\text{CCR}=\frac{\text{Number of samples correctly recognised}}{\text{Number of samples tested}} \text{.........................(3)}$$

Table 1: CCR and FRR for Original signature

Training Signatures	Testing Original Signatures	Threshold	CCR (%)	FRR (%)
25	50	0.1	76	24
		0.15	86	14

		0.2	90	10
		0.25	96	04
		0.3	98	02
		Average	89.2	9.2

The following table 2 shows the result of system in the form of Correct Classification Rate (CCR) and False Acceptance Rate (FAR) for forgery signatures. The CCR and FAR calculated as

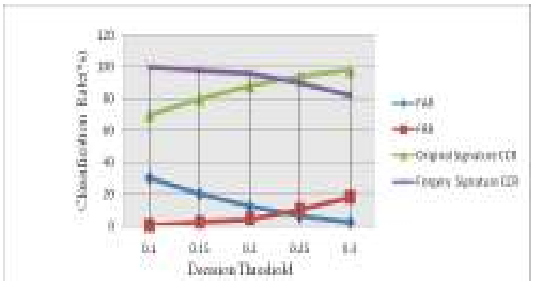
$$\text{FAR}=\frac{\text{Number of forgery accepted}}{\text{Number of forgery tested}}*100\ldots\ldots\ldots\ldots(5)$$

.

Table 2: CCR and FAR for Forgery signature

Training Signatures	Testing Forgery Signatures	Threshold	CCR (%)	FAR (%)
25	50	0.1	98	02
		0.15	96	04
		0.2	94	06
		0.25	88	12
		0.3	78	22
		Average	90.8	10.8

The Following graph 1 shows the FRR, CCR for original signature and the FAR, CCR for forgery signature at different threshold levels.



Graph 1: FAR, FRR, CCR for different threshold in Depth-3

Comparing the result of proposed method with other signature verification method such as Grid feature extraction & Signature Verification using Neural Network. The result is shown in Table 3.

Table 3: Comparative performance of system

Classification Rate	Grid Feature Extraction	Neural Network	Proposed Method (28FP)
FRR (%)	17.9	20	9.2
FAR (%)	9.7	14.66	10.8

6. CONCLUSION

This research was taken up with an objective of developing depth based signature segmentation algorithms for verification of Genuine and Forgery signatures. Signature verification in offline systems is completely based on the images of the signatures. It contains less discriminative information as it is often contaminated with noise either due to scanning hardware or paper background. A novel approach for off-line signature verification is proposed and implemented so that preprocessed signature i.e. resized, binarized and thinned signature is segmented into different parts specifying the depth value. From each individual segment, value of feature point is extracted by considering the signature pixels values. For verification, feature points of training signature images and test images have been compared using the Euclidean classification model and the test signature is then classified accordingly. So, the proposed system provides better result in terms of FRR, CCR for genuine signature and FAR, CCR for skilled forgery in different depth levels. The purposed algorithm result the FAR which is very much less as compared to the FAR of the previously existing techniques. So it is important for a user to sign his signature with at most care so that there is not a large variation of his signature. Otherwise there is a probability of rejection of an original signature.

7. ACKNOWLEDGMENTS

We wish to thank all the people who have helped us by obliging to give sample signatures for testing our technique. We also want to acknowledge the support of our family and Friends and my guide without which this paper would not have been a reality.

8. REFERENCES

[1] Swati Srivastava and Suneeta Agarwal, “Offline Signature Verification using Grid based Feature Extraction”, International Conference on Computer & Communication Technology (ICCCT)-2011 IEEE.

[2] Suhail M. Odeh and Manal Khalil, “Off-line signature verification and recognition: Neural Network Approach”, 2011 IEEE.

[3] Ashwini Pansare and Shalini Bhatia, “Handwritten Signature Verification using Neural Network” , International Journal of Applied Information Systems (IJAIS).

[4] L. Ravi Kumar and A.Sudhir Babu, “Genuine and Forged Offline Signature Verification Using Back Propagation Neural Networks”, (IJCSIT) International Journal of Computer Science and Information Technologies, 2011.

[5] B H Shekar and R.K.Bharathi, “eigen-signature: A Robust and an Efficient Offline Signature Verification Algorithm” IEEE-International Conference on Recent Trends in Information Technology, ICRTI- 2011 IEEE.

[6] Tai-Ping Zhang, Bin Fang, Bin Xu, Heng-Xin Chen, Miao Chen and Yuan-Yan Tang, “Signature Envelope Curvature Descriptor For Offline Signature Verification”, Proceedings of the 2007 International Conference on Wavelet Analysis and Pattern Recognition,-2007 IEEE.

[7] Dakshina Ranjan Kisku, Ajita Rattani, Phalguni Gupta and Jamuna Kanta Sing, “ Offline Signature Verification using Geometric and Orientation Features with Multiple Experts Fusion”, International Conference on Electronic Computer Technology (ICECT)-2011 IEEE.

[8] Alberto Martin and Sabri Tosunoglu, “Image Processing Techniques For Machine Vision”, Conference on Recent Advances in Robotics (ICRAR).

[9] Miguel A. Ferrer, Francisco Vargas, Carlos M. Travieso and Jesus B. Alonso, " Signature Verification using Local Directional Pattern (LDP) "International Conference on computer security technology(ICCST)-2010

[10] Siti Norul Huda Sheikh Abdullah and Khairuddin Omar, "State- of-the-Art in Offline Signature Verification System", International Conference on Pattern Analysis and Intelligent Robotics(ICPAIR)-2011 IEEE.

[11] Dr. S. Adebayo Daramola and Prof. T. Samuel Ibiyemi, "Offline Signature Recognition using Hidden Markov Model (HMM) ", International Journal of Computer Applications..

[12] Debnath Bhattacharyya, Rahul Ranjan, Farkhod Alisherov A, and Minkyu Choi, "Biometric Authentication: A Review", International Journal of u- and e- Service, Science and Technology, 2009.

[13] Igor Bohm and Florian Testor, "Biometric Systems", Department of Telecooperation University of Linz 4040 Linz, Austria,2006