ISLAMIC UNIVERSITY OF TECHNOLOGY(IUT)

A Statistical Approach for Off-Line Signature Verification Using Local Gradient Features

Authors:

Ashikur Rahman (104401)

Golam Mostaeen (104404)

Supervisor:

Md. Hasanul Kabir, PhD

Associate Professor

Department of Computer Science and Engineering (CSE)

A thesis submitted to the Department of CSE

in partial fulfillment of the requirements for the Degree of

B.Sc. Engineering in CSE.

Academic Year: 2013-14

A subsidiary organ of the organization of Islamic Cooperation

Dhaka, Bangladesh

October 2014

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

A handwritten signature is the scripted name or legal mark of a person's identity, executed by hand and it is used for the purpose of authentication. Signature is an age-old distinguishing feature for individual's identification. The signature verification is used as a popular, cost effective authentication method and preferred among various biometrics as it is the widely accepted way to identify an individual. It is used in many areas of society related to automated banking transaction, electronic fund transfers, and document analysis and access control throughout the world. Hence, methods for automatic signature verification must be developed if authenticity is to be verified on a day to-day basis. Signature can be forged in many different ways. According to Maini et al. [1] some of them are-

- Traced Forgery
- Simulated or Copied forgery
- Practiced Forgery
- Spurious Forgery
- Transplanted Forgery
- Computerized Forgery
- Color Copy Forgery

The forgeries in handwritten signatures have been categorized based on their characteristic features (Suen et al., 1999) [2]. The three major types of forgeries are:

I. **Random Forgery**
The signer uses the name of the victim in his own style to create a forgery known as the simple forgery or random forgery. These forgeries represent almost 95% of all the cases generally encountered in fraudulent cases although they are very easy to detect even by the naked eye.

II. **Unskilled Forgery**
The signer imitates the signature in his own style without any knowledge of the spelling and does not have any prior experience. The imitation is preceded by observing the signature closely for a while.

III. **Skilled Forgery**
Undoubtedly, the most difficult of all forgeries is created by professional impostors or persons who have experience in copying the signature. For achieving this one could either trace or imitate the signature by hard way.

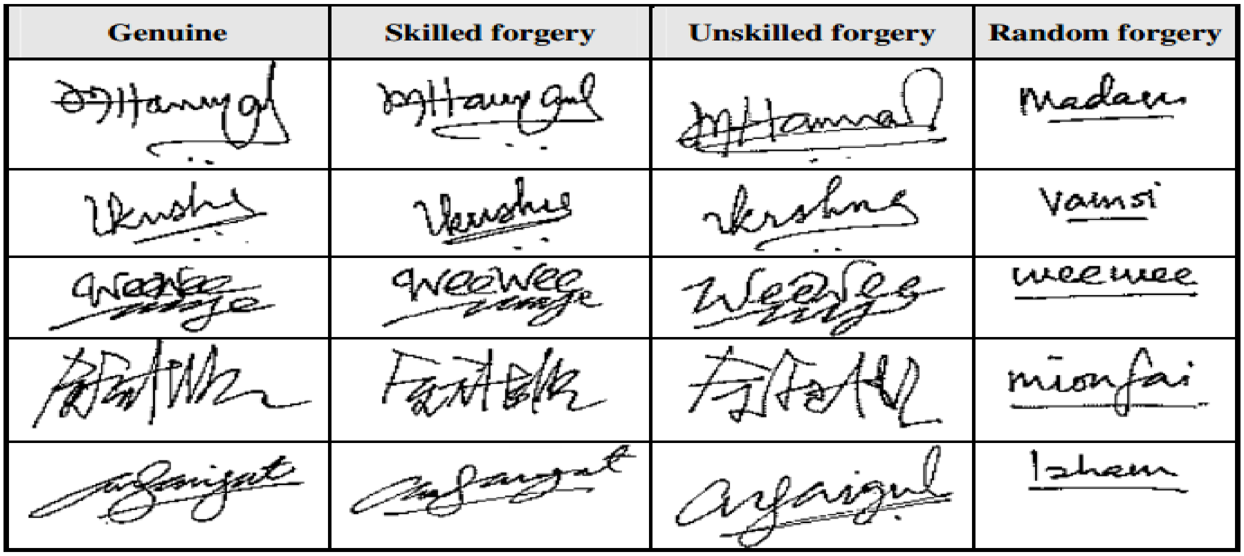


Figure 1: Different types of forgeries

1.2 Problem Statement

As the forgery becomes easy, there is a need to improve the methods for authenticating a person with a signature. Approaches for signature verification fall into two categories according to the acquisition of the data i.e. On-line and Off-line. On-line data records the motion of the stylus when the signature is produced, and includes location, and possibly velocity, acceleration and pen pressure, as functions of time. These dynamic characteristics are specific to each individual and sufficiently stable as well as repetitive. Off-line data is a 2-D image of the signature. Our focus is on Off-line signature verification.

1.3 Research Challenges

The difficulties in processing off-line signature are (i) the highly stylish and unconventional writing, (ii) the nature and variety of the writing pen (iii) the non-repetitive nature of variation of the signatures, because of age, illness, mood, stress levels, geographic location and perhaps to some extent the emotional state of the person

In Signature verification process there are some challenging situations. These all kinds of situation has not been overcome yet in the existing methods. The challenges are like below.

- **Variation of the sample**
  The genuine signature of the user varies because of age, illness, mood, stress levels, geographic location and perhaps to some extent the emotional state of the person

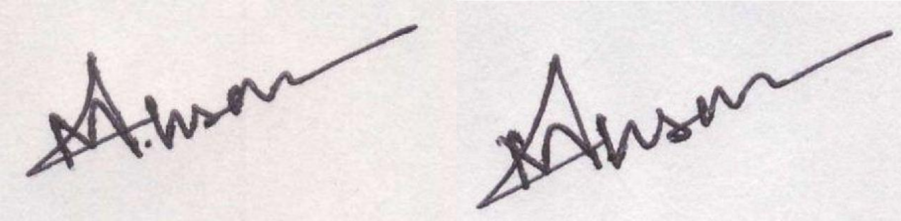


Figure 2: Signature of the same person

- **Different orientation**

  Signature can have different orientation in different time.

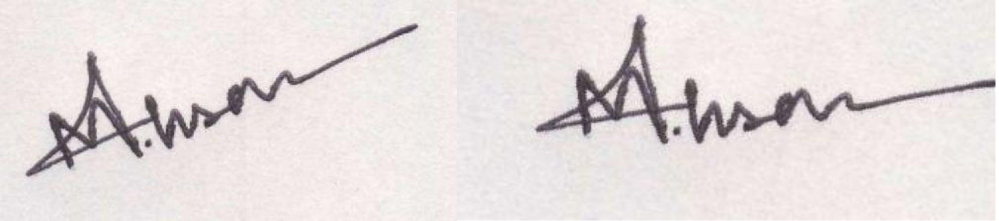


Figure 3: Same signature having different orientation.

- **Noisy Image**

  Input image can be noisy. In that case extracting feature from the image will be difficult.

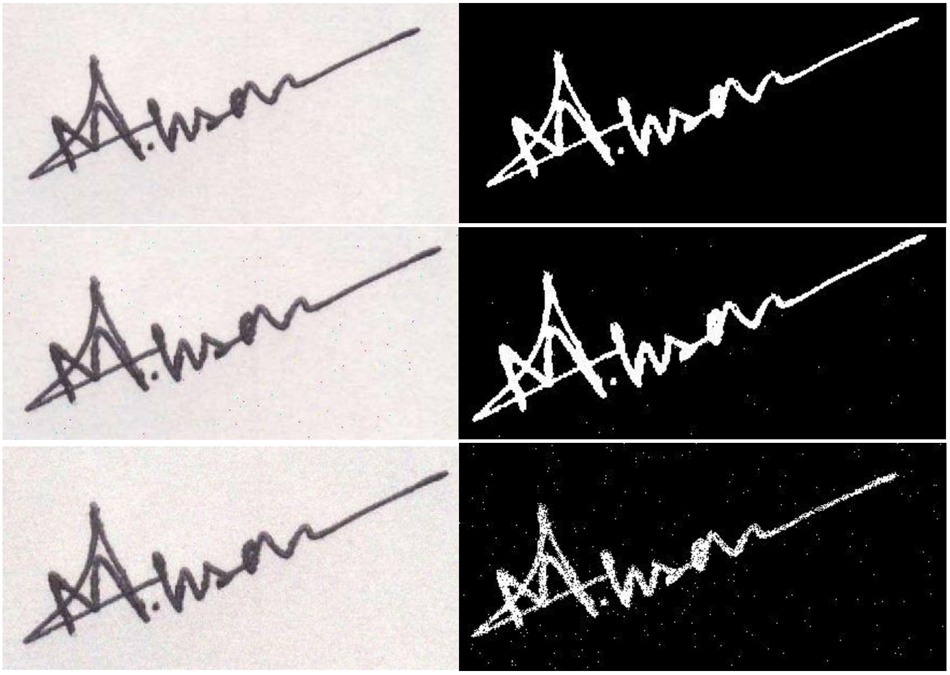
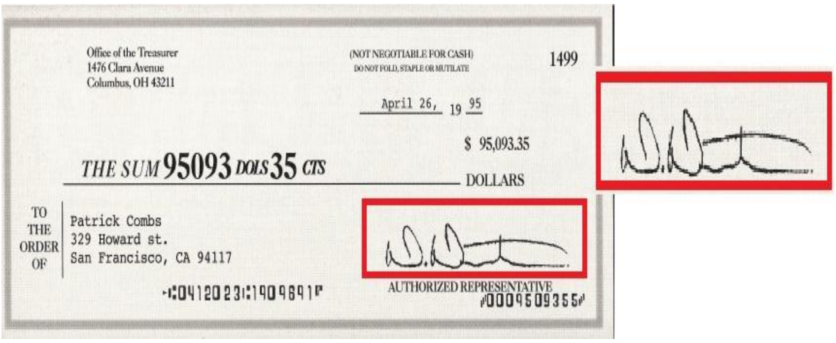


Figure: **a**)Noise less image **b**)Binary Response of **a**
c)Image with ‘salt & pepper’ noise **d**)Bianry Response **c**
e)Image with ‘gaussian’ noise **f**)Binary Response **e**

- **Sector of interest**

  The signature can be located with many redundant portion. Removing those redundant portion and getting the original signature is a big challenge.



Office of the Treasurer
1476 Clara Avenue
Columbus, OH 43211

(NOT NEGOTIABLE FOR CASH)
DO NOT FOLD, STAPLE OR MUTILATE

1499

April 26, 19 95

$ 95,093.35

THE SUM 95093 DOLS 35 CTS DOLLARS

TO THE ORDER OF

Patrick Combs
329 Howard st.
San Francisco, CA 94117

⑆041202 3⑆1909891⑈

AUTHORIZED REPRESENTATIVE

⑈0009509355⑈

Figure 4: Isolating the sector of interest.

- **Choosing Threshold**
  Threshold value should be taken wisely so that False accept and False Reject occur very less. If the threshold is not optimum, detection will result in error. And the threshold should be person specific as everyone has their own variance in signature.

- **Thickness of the line**
  Because of different nature of pen the signature line width vary. That's causes some error in detecting random forgery.

1.4 Thesis Objective

Now a days authenticating the signature in bank cheque has become so important because a large amount of transaction occurs daily by cheque. So, to improve this authentication is a necessity. Most of the existing methods failed to successfully verify signature in different challenging situations. That's why we want to develop a method to verify signatures successfully. We already discussed about the challenges in this field. And also claimed that no existing method can ensure different situations. Some methods are working well with approximate match but they fails on noisy situation and in the aspect of skilled forgery. Some methods work well, but take long time which is not suitable for banking sector. Moreover, the complexity of those methods are very high. So still there are a lot of scopes to develop the verifying method. We want to develop a method that will pass those challenges described and be very fast. Our thesis aimed at proposing a method which can verify signatures in case of different challenges like-

- Gaussian noise
- Salt & pepper noise
- Different orientation
- Time complexity
- Skilled forgery

1.5 Thesis Contributions

Already we have discussed about the research challenges in this field. And also we claimed that no existing method can ensure different situations. We are proposing a signature verifying method which will be robust to noises, different orientations and forgeries. The major contributions of our thesis are summarized below-

- We have presented a statistical approach for signature verification with person specific threshold which will be robust to noises, different orientations and forgeries etc.
- We have investigated several existing methods for signature verification identifying their weakness and tried to solve those with our method.
- Our method describe a statistical approach that works with local gradient features which is rotation invariant.
- In the comparison step we have compared the best math with the second best match up to a threshold ratio, which ensures good match and helps discarding bad match.
- For performance analysis, we implemented our method and figured out the performance using benchmark dataset.
- Comparative analysis of our method with others have also done for getting the comparative performance of our method.

1.6 Organization of thesis

The rest of thesis will be organized as follows:

In Chapter 2 we present the literature review of existing methods and their performance, strong point and weak points. In Chapter 3, we propose our verification method. There we discuss about the overall idea of our proposal and step by step implementation process. In Chapter 4, experimental set-up, experimental result and performance analysis of our method with various challenge is shown. Besides with other methods a comparative analysis is also shown. Lastly, in Chapter 5, we conclude our contribution in this thesis and shows the future scopes for further developing.

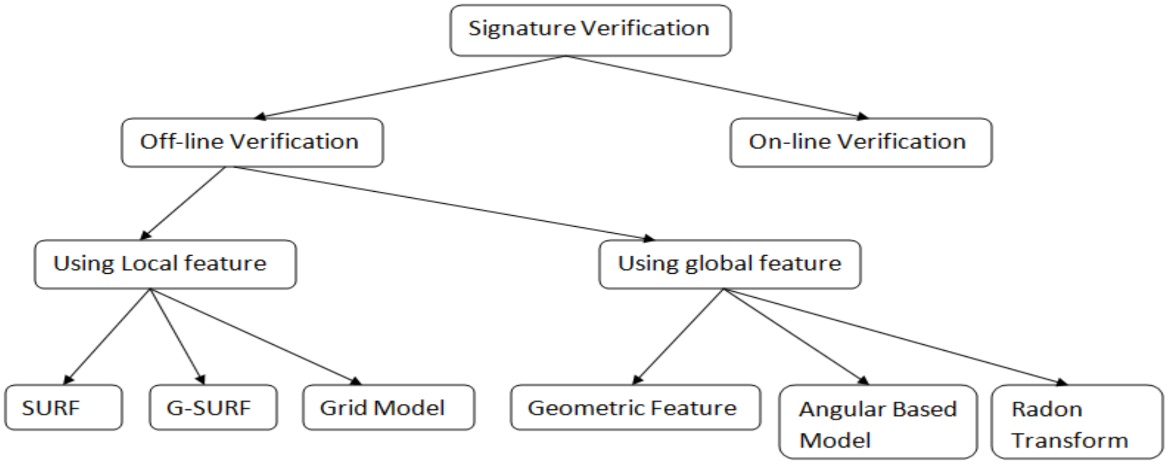
Chapter-2

Literature Review

Signature is the most popular tool to identify a particular person now a days. A signature authenticate whether any written paper, letter, writings or cheque is from that person as the characteristics of that signature is known in advance. Mostly banking sector is widely dependent on handwritten signatures. A large amount of money is transferred by relying on it. That's why this is the prime target of fraudulence. So, authenticating signature of the account holder is the most prior task in the cash section. Random forgeries are easily detectable. But, for the advancement of technology, forgery now a days became very skilled which is very tough to verify. That's why a digital automatic system is necessary in this sector. Many methods has been proposed for this task. But those methods are not capable to overcome all sorts of challenges. In this section we are going to discuss about existing methods.

2.1 Related Works

There are many methods exists now a days. Those method can be divided into two parts mainly on basis of their feature selection. One kind of them uses global features, which is calculated considering the whole image. Another one uses local features, which is calculated from particular points. The classification is given below-



2.1.1 Global features

In this method, features are extracted on basis of the total image. These features are size, ratio, height and width, total number of black pixels, center of gravity, mean, variance etc. Some of the methods that uses global features are discussed below.

2.1.1.1 Geometric feature

A.C. Verma et al [3] proposed this method that uses global and geometric features. Details of this method is given here-

- Geometric data (aspect ratio, center of gravity, baseline shift etc.) are considered as feature
- Mean of each feature calculated from the training data. This mean is regarded as the prototype of a user.

$$\mu = \frac{1}{n}\sum_{i=1}^{n} f_i * x_i$$

- Variance of the input signature from the mean is calculated using the following equation-

$$\sigma^2 = \frac{\sum(\mathrm{X}-\mu)^2}{\mathrm{N}}$$

- For the basis of comparison, Euclidian distance is calculated.

$$\delta = \frac{1}{\mathrm{n}}\sum_{\mathrm{i}=1}^{\mathrm{n}}\left[\frac{(F_i-\mu_i)}{\sigma_i}\right]^2$$

Benefits:

Very simple to implement and good for detecting random forgery.

Limitations:

- Cannot detect skilled forgery as the geometric value get closer
- False accept occurs more often

2.1.1.2 Angular Based Model

Prashanth and Raja proposed this method [4]. This method uses two processes- **1)** Detecting Random forgery **2)** Skilled forgery

1. For detecting random forgery, it calculates the average no. of Rows and columns in each sample. Then compares with the input image. If it is in acceptance range, then goes for second phase, otherwise reject.
2. For skilled forgery, it divides an image into two horizontal blocks on basis of the geometric center of the image. Geometric center is the point where the no. of black pixels is half of the total no. of black pixels. These points are regarded as the feature points. Then each and every block is divided vertically and horizontally until 128 points are found.

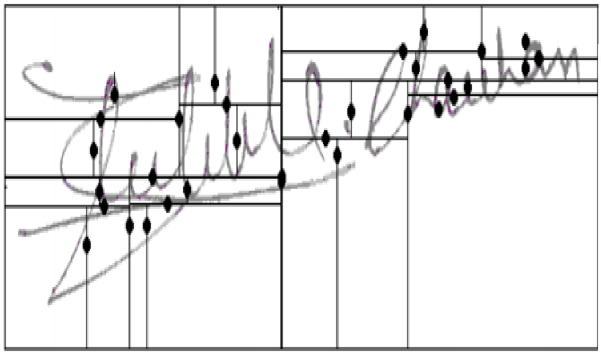


Figure 5: Image Splitting.

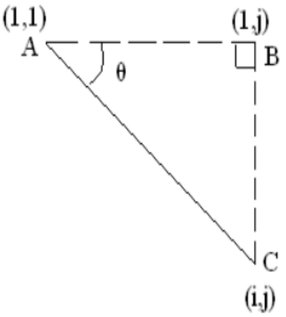


Figure 6: The angle and distance calculation.

Then for each point the distance and the angle from the point (1, 1) is calculated. These value describe the image. Testing image is compared with these values and on basis of a threshold this method validate.

Benefits:

Main benefit of this method that it handle the random forgery and skilled forgery separately.

Limitations:

- It depends on the global values of the feature, that can result in false accept.
- Different signature can have geometric center on the same point.

2.1.1.3 Radon Transform

- Computes projection sum of the image intensity along a radial line oriented at a specific angle with the formula-

  $$R(p,\theta)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}g(x,y)\delta(p-x\cos\theta-y\sin\theta)dxdy$$

  Where the δ(r) is Dirac function.

- Computation of Radon Transform is its projections across the image at arbitrary orientations θ and offsets ρ which is used as feature

Limitations:

False reject rate is little high for this method

2.1.2 Local features

Local features depends on particular portion of the input image. Mainly, some keypoints are detected from the input images and features are extracted for those particular keypoints.

2.1.2.1 SURF

To get better and faster output than SIFT, SURF [5] was proposed.

- First, key point is detected using fast Hessian Detector and Haar wavelet. The formula of Hessian matrix is-

  $$\mathcal{H}(\mathbf{x},\sigma)=\begin{bmatrix} L_{xx}(\mathbf{x},\sigma) & L_{xy}(\mathbf{x},\sigma) \\ L_{xy}(\mathbf{x},\sigma) & L_{yy}(\mathbf{x},\sigma) \end{bmatrix}$$

- Then SURF descriptor is extracted using assignment orientation and descriptor components.

Benefits:

- Rotation invariant
- Faster than other algorithms like SIFT.

Limitations:

- Though it is fast, now a days with faster computers we can beat the time.

2.1.2.2 G-SURF

S. Pal et al [6] proposed a method in 2012 using Gabor filter before SURF.

- Uses Gabor filter along with SURF algorithm
- A two dimensional Gabor Filter in spatial domain can be defined as follows-

$$G(x, y, \lambda, \theta, \psi, \sigma, \gamma) = \exp \{(x'^2 + \gamma^2 y'^2)/2\, \sigma^2 \} \cos(2\, \pi\, x' / \lambda + \psi)$$

Where ,

$$x' = x\cos\theta + y\sin\theta$$

$$y' = -x\sin\theta + y\cos\theta$$

Limitations:

- Though it overcomes the performance of SIFT and SURF, still need to be upgraded

2.1.2.3Grid Model

Madasu et al [7] described a grid model which split the image into a grid version.

- Image is partitioned into 8 partitions using equal horizontal density approximation method.

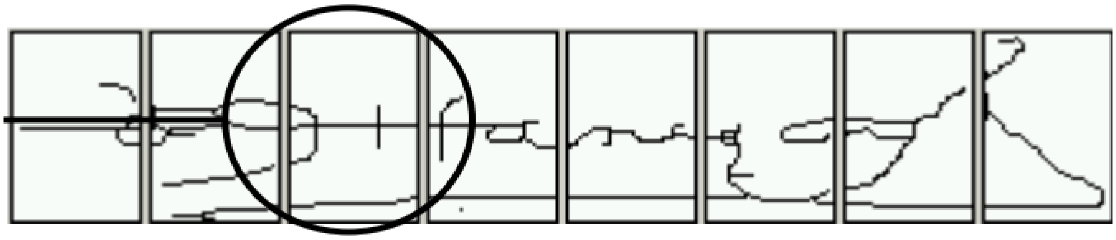


Figure 7: Images is split.

- Each Box portioned into 12 boxes (3 column, 4 row). This result in a total of 96 (8x12) boxes.

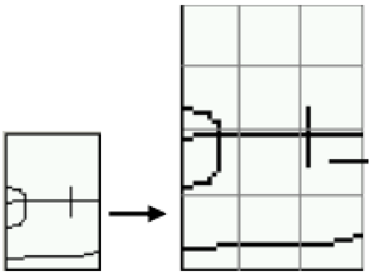


Figure 8: Box Partition

- Then for each block, angles of each set pixel from the lower left corner of that box is calculated and summed up. Each image will have 96 values as the feature vector.

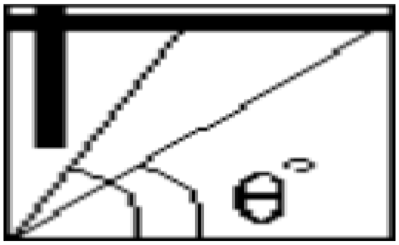


Figure 9: Angle from the lower left corner

Limitations:

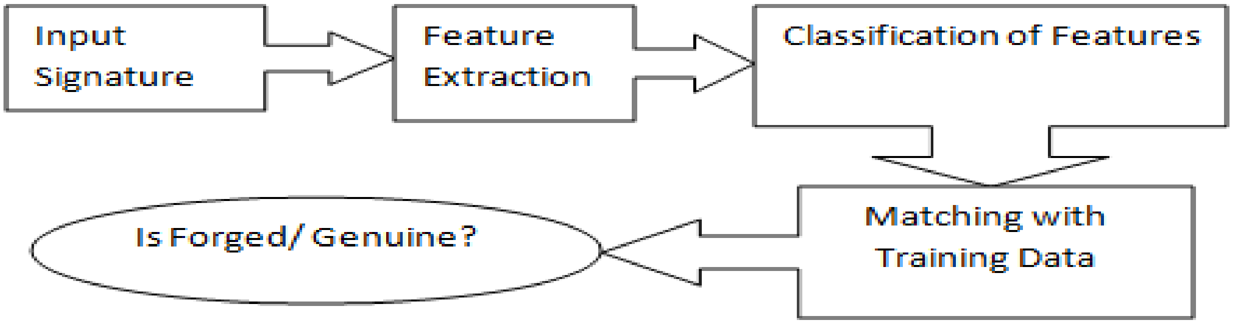
As it takes the value of each block, if the signature is little bit slant, the output will vary a lot.

2.2 Overall Detection Process

Most of the existing methods have same verifying criteria. First there is some preprocessing in every method. This is done to prepare the raw image for the later parts in which the verification is managed. Then feature is extracted from the input signature images. This part is very important as the whole process depend on this part. Each feature describes image. That's why features are also known as descriptors. After that, those features are classified into several classes. This part is optional. It varies from method to method. Then it goes for matching. In this portion threshold selection plays very important role. Some methods have extra one step as threshold selection to calculate the optimum threshold that gives the best result.

All the methods of signature verification undergoes the following steps:

- Feature is extracted (Varies from methods to methods)
- Features are classified
- The system is trained
- Matching



Chapter 3

Proposed Method

3.1 Skeleton of Proposed Method

Initially a set of signatures are obtained from the subject and fed to the system for learning. These signatures are pre-processed and the pre-processed signatures are used to extract the key-points. Feature descriptor is then found out for each of the key-points obtained. The training signatures are then matched with one another to get a certain pre-defined amount of match points. The threshold for obtaining that number of matches is updated accordingly. Thus the optimal threshold value is calculated. The mean and variance of the matches is also then calculated for the optimal threshold. In the next phase the test signature, which is to be verified is fed to the system. Feature extraction is done as similarly as the above steps for this test signature which is then matched with each of the training signatures for the learned threshold. The signature is classified to be genuine if the number of matches falls within the calculated variance otherwise the test signature is classified as forged one.

The following figure shows the overall steps of the proposed method:

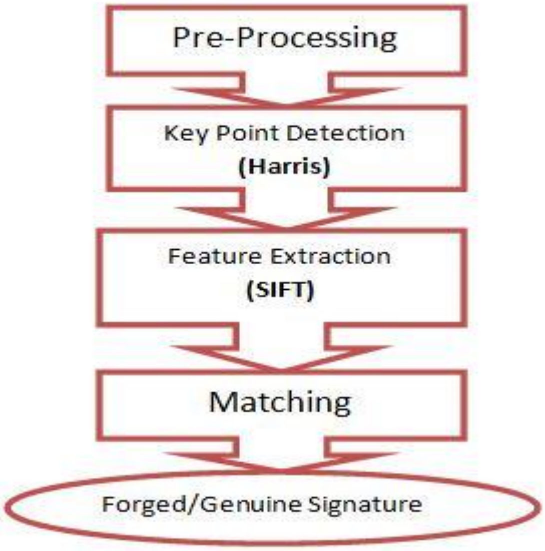


Figure 10: Basic steps of proposed method.

3.2 Pre-Processing

Signature pre-processing is a necessary step to improve the accuracy of Feature extraction, Classification and to reduce their computational needs. The purpose of pre-processing phase is to make signatures standard and ready for feature extraction. In our proposed method we apply some sort of pre-processing before going to the actual verification of the signatures. The preprocessing includes the following:

3.2.1 Scanning

For our implementation and testing of the proposed method a number of signatures from different persons are scanned. Some of them are used for training and others are used for testing. The signature sheets were scanned at 300 DPI (dots per inch).

3.2.2 Cropping

Signatures are then cropped from the signature sheet so that some of them can be used for training and others for testing separately. Cropping is done with respect to bounding box of the supplied signature by calculating the first foreground row, first foreground column, last foreground row and last foreground column.

3.2.3 Conversion to Gray-scale Image

Our proposed method work best with the gray scale images. But some signature may be given as color image. In that case for the better performance of the verification process we first convert the given color image to the gray scale image. For that conversion we just follow the following conversion equation:

$$\boldsymbol{I} = 0.299\boldsymbol{R} + 0.587\boldsymbol{G} + 0.114\boldsymbol{B}$$

3.2.4 Noise Reduction

Removal of random noises is one of the important steps in pre-processing. Simple “*Box Filter*” is used for the removal of random noises from the supplied signature. Considering the practical scenario supplied signature may contain a number of different noises; which are reduced in the pre-processing step. For example the following left image of signature shows the presence of ‘salt and pepper’ which has been removed in the right one by applying Median filter.

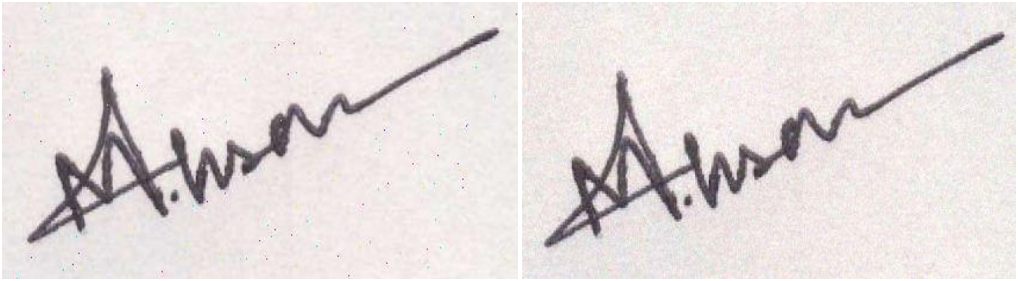


Figure 11: Result after noise reduction

3.3 Obtaining Keypoint using Harris Corner Detection

An interest point is a point in an image which has a well-defined position and can be robustly detected. This means that an interest point can be a corner but can also be, for example, an isolated point of local intensity maximum or minimum, line endings, or a point on a curve where the curvature is locally maximum.

The Harris corner detector algorithm [8] relies on a central principle: at a corner, the image intensity will change largely in multiple directions. This can alternatively be formulated by examining the changes of intensity due to shifts in a local window. Around a corner point, the image intensity will change greatly when the window is shifted in an arbitrary direction. Following this intuition the Harris corner detector uses the second moment matrix as the basis of its corner decision. From here the general equation of Harris can be shown as the following:

$$E(u,v) = \sum_{x,y} \underbrace{w(x,y)}_{\text{window function}} \, [\underbrace{I(x+u, y+v)}_{\text{shifted intensity}} - \underbrace{I(x,y)}_{\text{intensity}}]^2$$

As we said above Harris corner detector tries to quantify the local intensity changes at all the directions for each of the pixels. The figure below may be useful to demonstrate the basic idea:

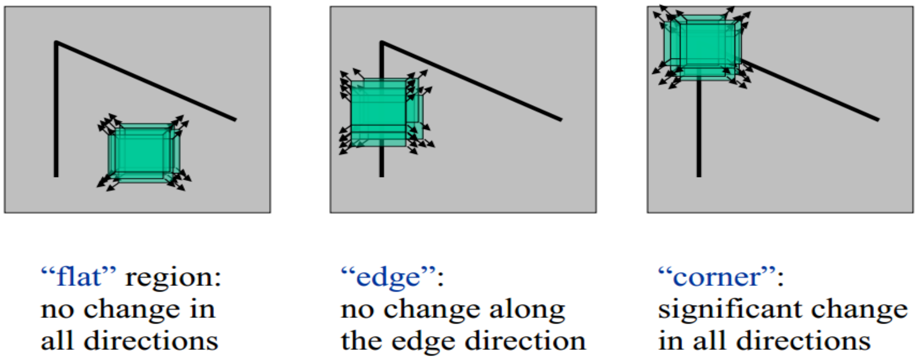


Figure 12: Basic of Harris corner detection method

So, in this step of our proposed method we find out the key-point of the supplied signature using Harris Corner Detection Algorithm. Those key-points define the exact skeleton of the supplied signature. The red dots of the following signature were found as key-point for the respective signature:

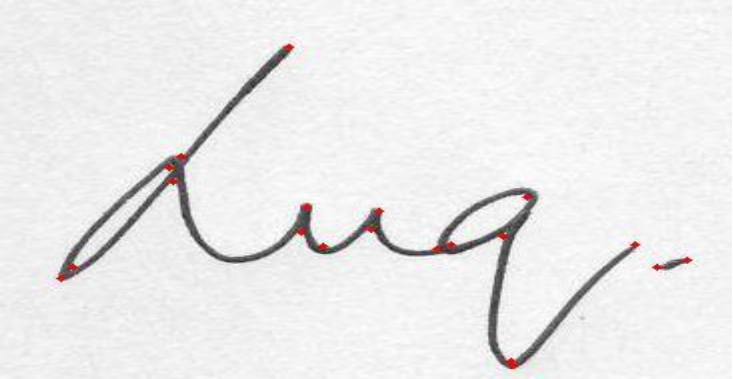


Figure 13: Keypoint detected (shown in red dots) in signature using Harris corner detection method

As we can see that the key-points of this respective signature defines the basic structure. The obtained key-point is supplied to the next step of our proposed method.

3.4 Feature Extraction of the Key-points

In this step the feature descriptor is calculated for each of the key-point found in the previous step. We used SIFT (Scale-Invariant Feature Transform) for calculating the descriptor so that descriptors are highly distinctive, invariant as possible to variations such as changes in viewpoint and illumination. Before calculating the descriptor for each of the key-point *Gaussian* weighting around center is done in order to provide some weight that is inversely proportional to the distance from the key-point. Next a 16x16 neighborhood around the key-point is taken. It is divided into 16 sub-blocks of 4x4 size. Gradient magnitude and orientation is calculated for each of those 16x16 points using following formulae-

$$m(x,y)=\sqrt{(L(x+1,y)-L(x-1,y))^2+(L(x,y+1)-L(x,y-1))^2}$$

$$\theta(x,y)=\tan^{-1}\left(\frac{(L(x,y+1)-L(x,y-1))}{(L(x+1,y)-L(x-1,y))}\right)$$

Where m(x,y) is the magnitude and θ is the orientation on that point. On basis of those two values, 8 bin orientation histogram is created for each sub-block. So a total of 128 bin values are available. The overall step may be shown as the following image:

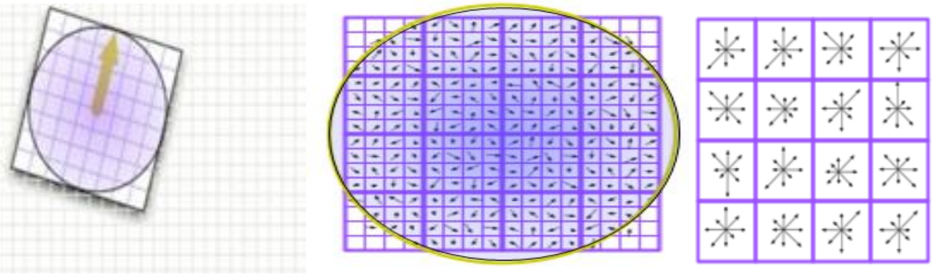


Figure 14: Feature (Descriptor) extraction

The first image shows that the Gaussian has been applied around the point of interest and rest two images show how the 128 bins are found as we said in the above. Finally the 128 bins value is represented in vector as key-point descriptor for the usage in next step of our proposed method.

3.5 Threshold Selection

In this step the system is trained with the signature of individuals. Now-a-days signature is used in number of sectors for checking the authentication. In most of those sectors a few for example 4 or 5 signatures are given for checking the authentication in reference to those later. Keeping that in mind we have designed our system in way that can learn from relatively smaller number of samples. For our testing we used only 3 signatures for training our system from individuals.

The key-point descriptor from the previous step for two signatures is matched to see to how extent they are similar. We count the two feature descriptor as similar if their ratio of vector angles from nearest to second nearest neighbor is less than a threshold. That is a comparative lower threshold means we are being strict in saying the two feature descriptor to be similar.

Let we are given 3 signatures $\mathbf{S_1}$, $\mathbf{S_2}$ and $\mathbf{S_3}$for training the system. For each of the pair possible we want to keep the number of matches in between to a pre-defined number **matches_to_find**. So the threshold is increased via some step until we find the number of matches in between the two signatures equal or greater than **matches_to_find.**

The following image shows that given two signatures we find the matches in between them. The number of matches should be greater or equal to **matches_to_find;** the corresponding threshold for those two signatures is increased in step accordingly.

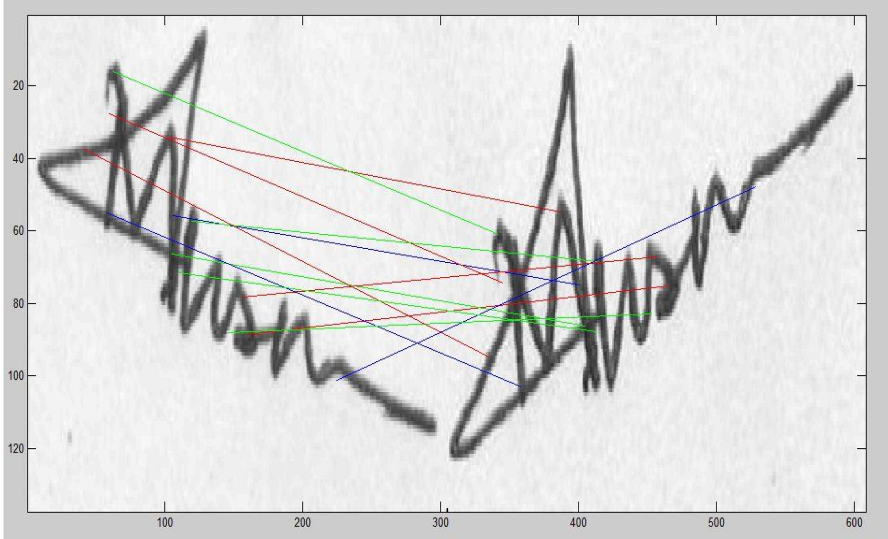


Figure 15: Matches found between two samples

This threshold is found out for each of the possible pairs (**S_1**, **S_2**), (**S_2**, **S_3**) and (**S_3**, **S_1**) of the given signatures. Let those thresholds be **$thres_1$**, **$thres_2$** and **$thres_3$** respectively. The mean and variance of the thresholds are calculated for the usage in the next step of the proposed model.

3.6 Matching

The calculated threshold is used in all the pairs of signatures (**S_1**, **S_2**), (**S_2**, **S_3**) and (**S_3**, **S_1**) and corresponding total number of matches **M_1**, **M_2** and **M_3** respectively for each them is calculated. Mean and variance of the matches are also calculated for the use in testing of the signature in the next phase of the proposed model. We have chosen the best threshold using ROC curve. That person specific threshold is stored in database. The training phase is thus completed for a class of signatures.

3.7 Testing

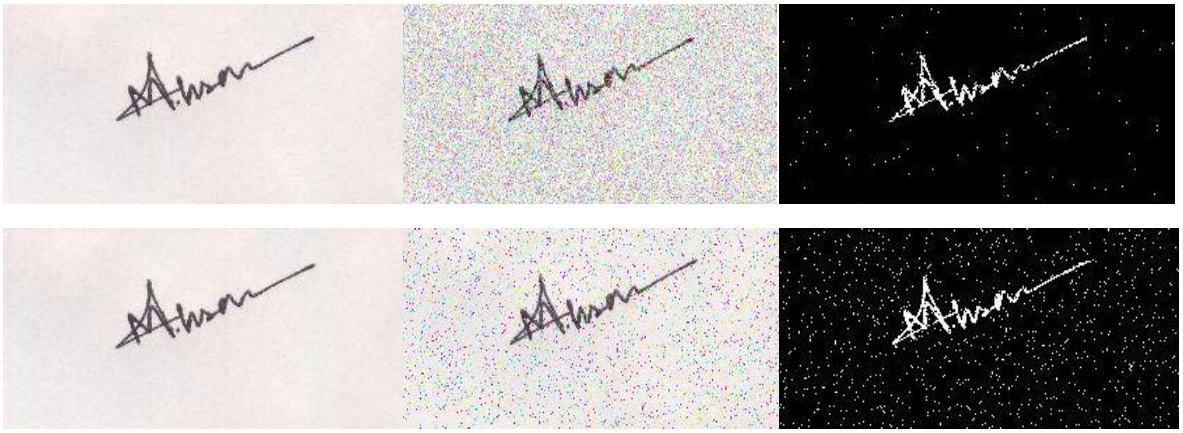
Now for the test signature **T_1**, the same way Feature descriptor is found out. That is the test signature same as the training signature is pre-processed, key points are calculated and then Feature descriptor is found out using SIFT. **T_1** is then matched with all training signatures using the pre-calculated threshold for that specific class of signature. The test signature is decided to be genuine if the mean of the number of matches found to be within the variance for the definite class of signature, otherwise it is decided as a forged signature.

Chapter-4

Experimental Result and Performance Analysis

4.1 Data Set and Experimental Setup

We have implemented the proposed model and compared with a number of existing approaches. All those implementation and simulation have been done using MATLAB 2013 on a personal computer of 2.13GHz processors with 2 GB main memory. For the comparison and testing of the proposed method we have used a number of signatures from 20 different persons. For each of the person we have used 6 signatures, 5 of them are genuine and the rest one is forged by someone else. Out of those 5 genuine signatures only 3 are used for training the system. And then the system is tested using remaining 2 genuine signatures and 1 forged signature. In practical scenario the signature verification can have a number of challenges for example the signatures provided can be noisy, the signature may be in different rotations etc. We have simulated some of those challenges to test the performance of the proposed method. From those view our whole dataset used can be divided into four groups. Our first section of dataset contains ideal signatures that are with negligible amount of noises and with no rotation. The second set of dataset contains 'salt and pepper' noise. The third set of dataset contains Gaussian noise with five different variances of 0.01, 0.02, 0.05, 0.1 and 0.2. Both of those noises are simulated using MATLAB codes. The fourth set of signature contains different rotation; that is the training and test signatures provided was of different rotations. Rotations are done with the help of Photoshop for different random angle values. A different signature of different dataset types has been shown in the following. This chapter contains the experimental result analysis for each of the mentioned dataset types.



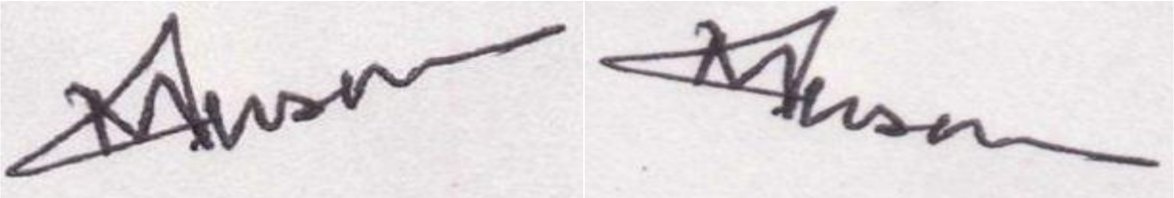


Figure 16: Dataset with Gaussian noise, ‘salt & pepper’ noise and different orientation.

4.2 Performance Measurement

We have measured the performance by calculating accuracy. Accuracy means the percentage of test signatures correctly classified. As, we have 20 classes and each class has 3 train data and three test cases: two genuine and one forged, we have 60 test cases. We also added 5 undefined signatures. That makes a total of 65 test cases. First, 40 genuine test cases needed to be assigned to correct person and the forged and last five test should be rejected as they are of no class. If it is done so, we say it correct classification, otherwise it is wrong. We calculate the accuracy by the following formula:

$$\text{Accuracy} = \frac{Number\ of\ signatures\ classified\ correctly}{Total\ number\ of\ test\ cases}$$

Higher accuracy means a less possibility of error. As in banking sector, financial transactions are related with signature verification, that’s why accuracy should be good enough. We discussed about various challenge in this sector in Chapter 1. Here we show the performance of our proposed method in those challenges.

4.2.1 Gaussian Noise

First we have noisy images. Noise is very common to the images due to the quality of scanner, impulse and other reasons. The most common of them is Gaussian noise. We added Gaussian noise to our dataset to measure performance for Gaussian noise. We used weinner2 filter of size 5x5 to eliminate Gaussian noise. This filter works well to eliminate noise upto a threshold level. The following figure shows the result of weinner2 filter in different noise level.

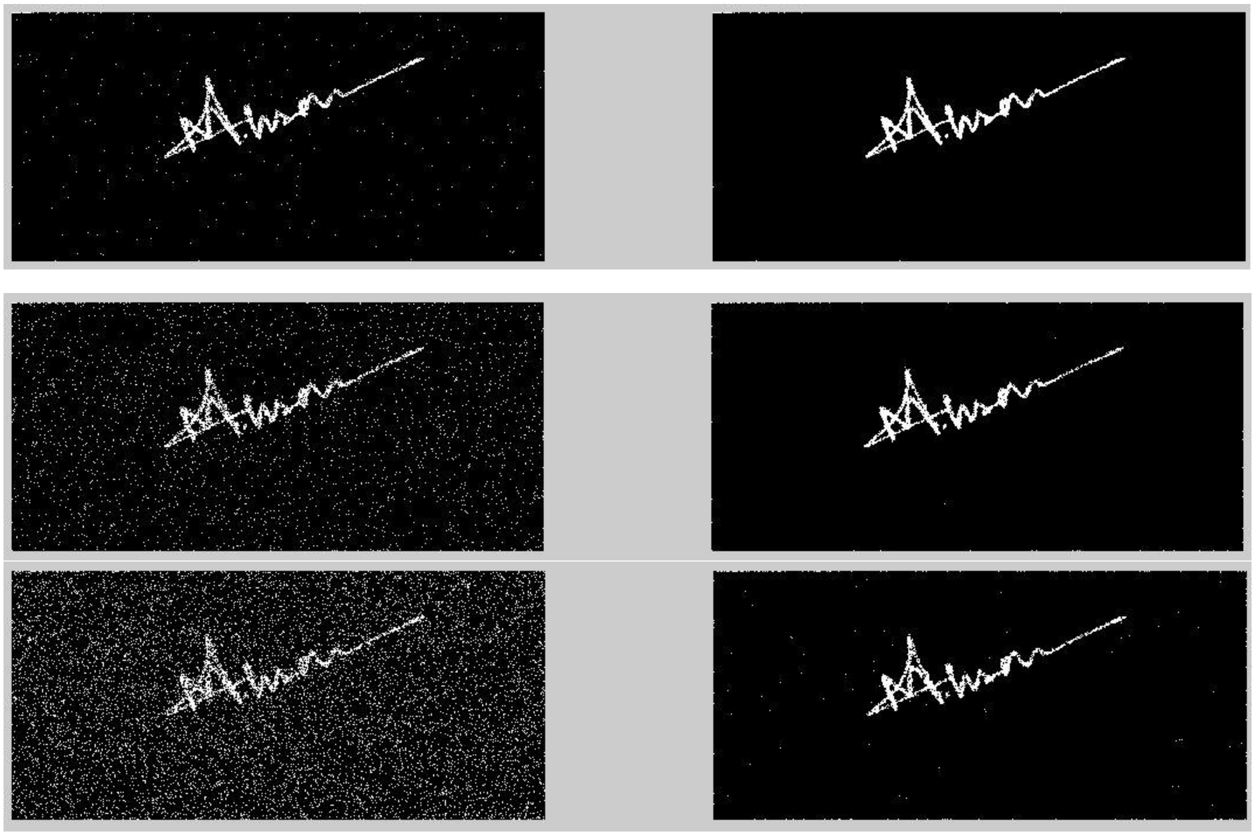


Figure 17: Output of weinner2 filter (noise level having variance 0.05, 0.1 and 0.2 accordingly)

After this we calculated features and find matches. The following table show the details of the performance of our method in Matlab:

Noise level (variance)	Accuracy (%)
0.01	93.84
0.02	93.84
0.05	92.30
0.10	87.60
0.20	80.00

.

4.2.2 ‘Salt & pepper’ noise

Another common form of noise is ‘salt & pepper’ noise. It happens very often. We used median filter of size 3x3.The output is-

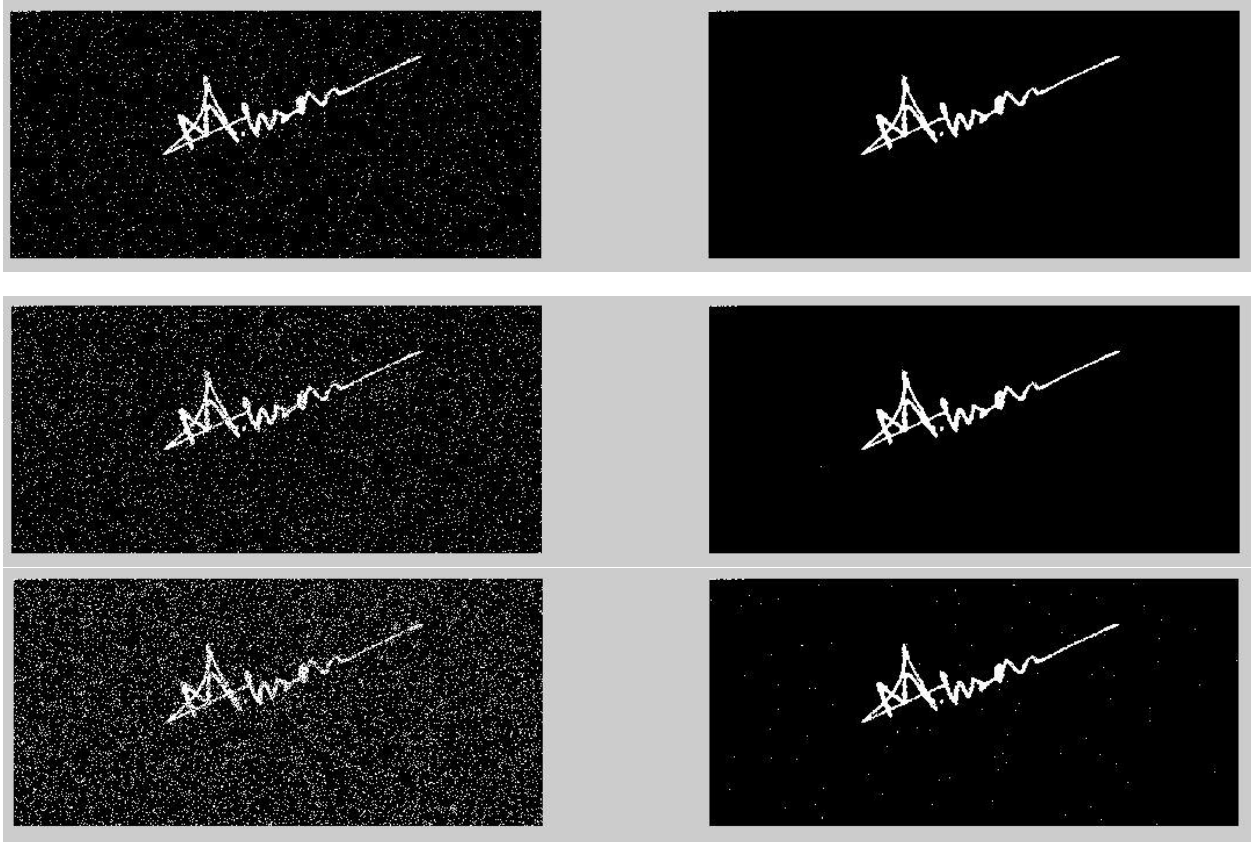


Figure 18: Output of median filter (noise level having variance 0.05, 0.1 and 0.2 accordingly)

Median filter can perform well if the noise is not more than the half of the pixels. Otherwise it fails to eliminate noise. In practical, the noise level is not that much high. So, median filter is an appropriate choice. The following table shows the performance for the dataset containing 'salt & pepper' noise.

Noise level (variance)	Accuracy (%)
0.02	93.84
0.05	93.84
0.10	90.70
0.15	89.23
0.20	86.15

4.2.3 Different orientation

Another common problem is with the orientation of signature. As signature are given in cheque in different situations, their orientation may vary. They can be in different angles. Whatever the angle is the signature should be classified correctly. As we selected local features to describe our samples, they have the benefit that the feature descriptor remain unchanged in spite having different angles. The following image shows the matches found between signatures in different angles. Here, only few matches are shown for better understandings.

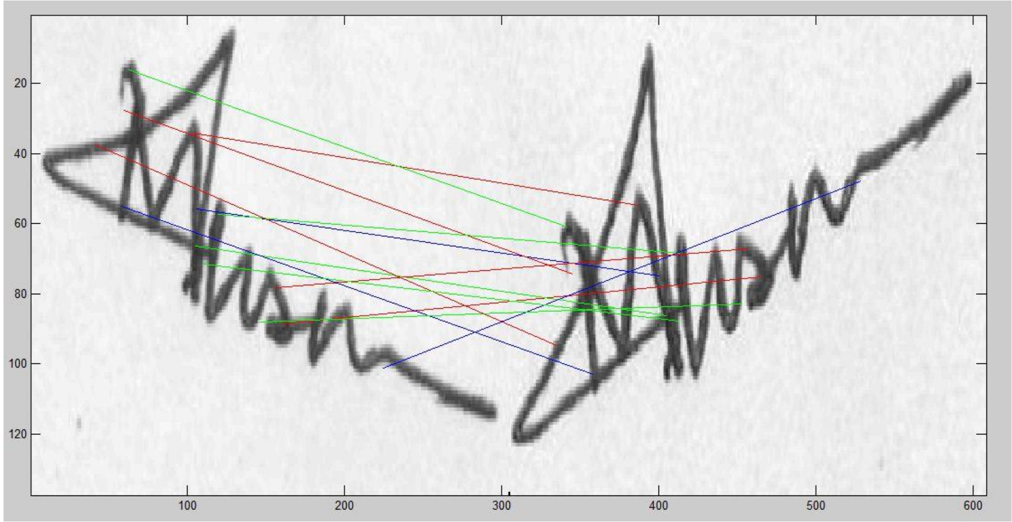


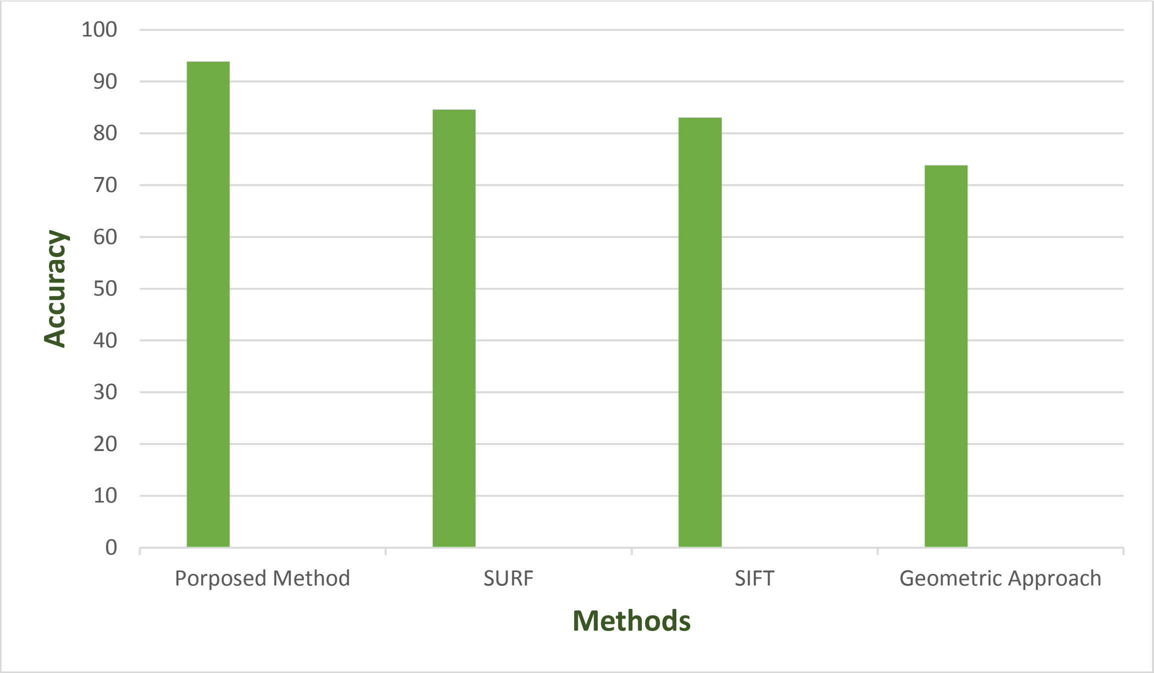
Figure 19: Matches found in signatures with different orientation.

4.3 Comparative Analysis

We have shown that how effectively our method works in different challenging situations conducting our experiment in different dataset. Now, we will compare our method with the existing methods. We implemented SURF, SIFT and one of the geometric feature based method and compared the outcome in different challenging situation with our proposed method. We can see that, existing methods cannot give a good performance in different challenging situations. Specially, they fail to detect the skilled forgeries. They lag behind in speed as those do some unnecessary calculations. The details of those comparison are discussed in the following portion.

4.3.1 Without Modification

At first, we tested the dataset images which contain no modification and the result of accuracy is shown in the following diagram along with the performance of other three existing methods. Although most of the methods accuracy is good enough, our proposed method gives a little better result amongst the others. The following chart shows that:

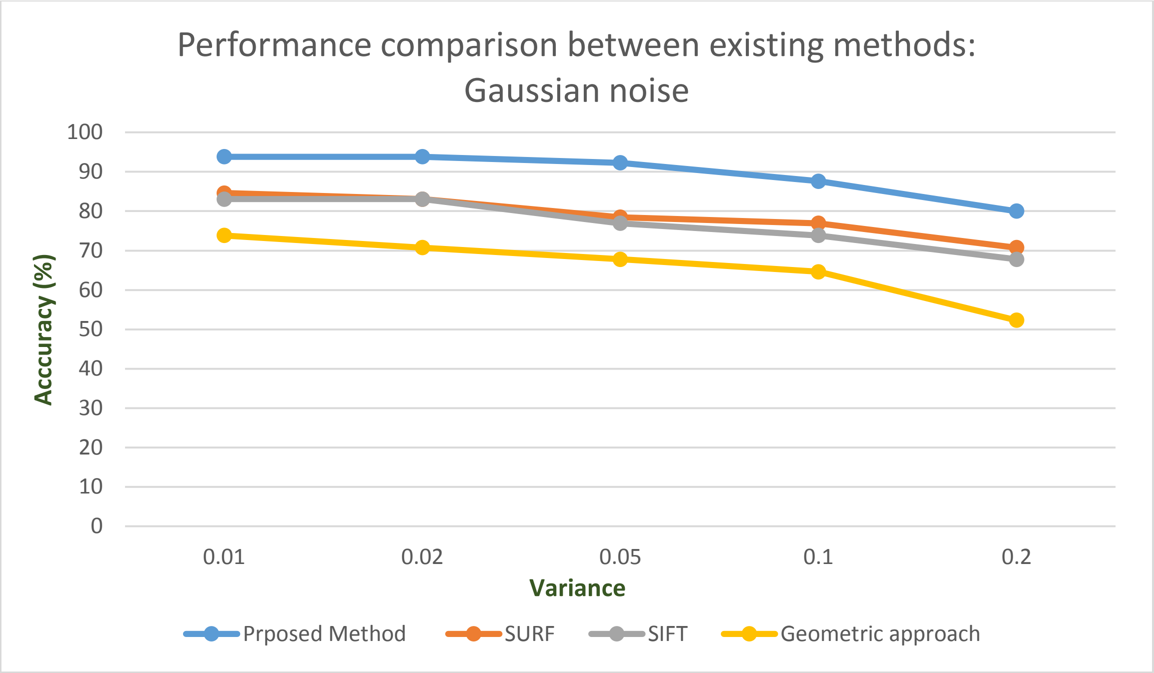


4.3.2 Noisy image

Noise is a common challenge in image processing. We compared how the existing methods response in different noise level. We took two most common noise: Gaussian noise and ‘Salt & pepper’ Noise.

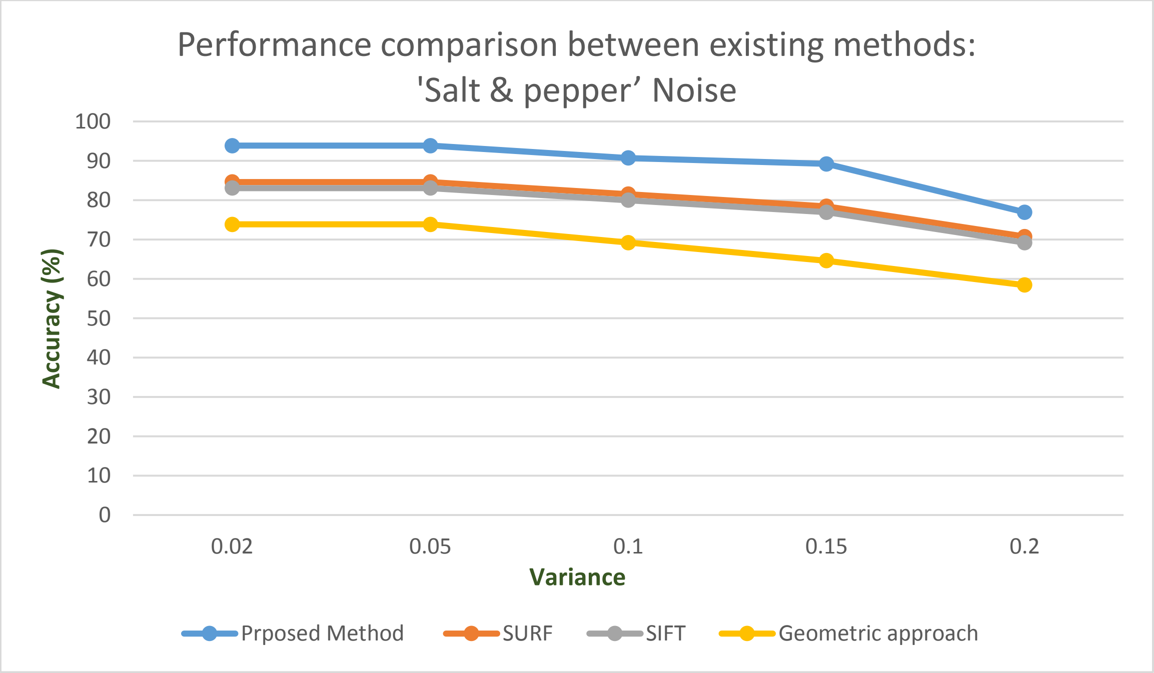
4.3.2.1 Gaussian noise

Gaussian noise is also known as white noise. We can see that, as the variance of noise increases the accuracy decreases. But, the performance of our proposed method is still better than the others. Normally, Gaussian noise does not occur having variance more than 0.2. Practically it is very less than that. Following chart show the performance comparison.



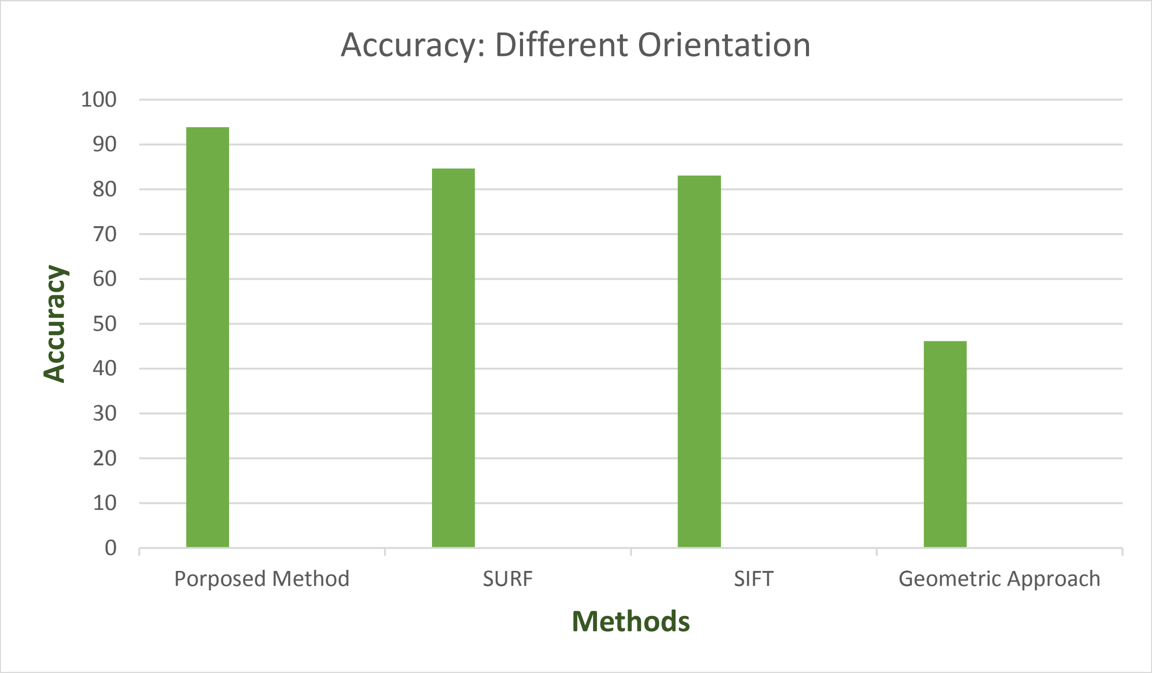
4.3.2.2 'Salt & pepper' Noise

Performance regarding 'Salt & pepper' Noise is almost same as Gaussian noise. Our proposed method perform much better than the existing methods.



4.3.3 Different orientation

Another very important challenge in signature verification is the different orientation. We have already shown you that our proposed method can easily find match even they are rotated in different angles. So, the accuracy is same as the unmodified dataset. For other methods this takes a little difficulty. As SURF and SIFT uses local features, they are also rotation invariant but problem is with the Geometric approach. It fails to perform better in different orientations.



Chapter-5

Conclusion

5.1 Summary of Contributions

Here we have presented an investigation of the performance of off-line signature verification. There are a number of existing methods that extract different features to train and test the system. In most of them a global decision line is used for deciding if a test signature is genuine or forged. But from practical scenario some signatures are too complex where some are not. Besides some signature undergoes a lot of changes, while even given by the actual person based on different environment or mental conditions. So always using the same decision line for every person's signature does not work the best in real life. Our proposed method uses different decision line for different person. Our system learns from the training signature's variations and nature to select an optimal decision line that works efficiently in practical day to day scenario. Besides signature are not expected to be given in the same rotation always. But a number of existing methods fail to handle this type of rotation in the supplied signature. We have proposed a system that works fine with any rotation of the training or test signature. Most of the existing methods for signature verification need a huge amount of training data for being usable for verification; which is not effective for practical scenario. This proposed method can learn from comparatively lesser amount of training data thus making it suitable for usage in practical scenario.

5.2 Limitations and Future Work

According to our proposed method we find key-points and then calculate feature descriptor for each of the points. Both of them relatively time consuming task. Besides the proposed method takes more time for learning the person specific optimal threshold. However we have speeded up the overall process by finding all of them only once and then storing. This way we do not to calculate feature descriptor time and again which leads faster testing of the signatures. But in this way we need comparatively more data storage for each of the person. Our future work thus is to find an optimal approach that takes relatively less time and less data storage.

Bibliography

[1] H. A. Raman Maini, "Study and Comparison of Various Image Edge Detection Techniques," in *International Journal of Image Processing (IJIP), Volume 3, Issue 1*, 2010.

[2] C. Suen, "Automatic recognition of handwritten data on cheques – Fact or Fiction?," in *Pattern Recognition Letters, 20, 1287-1295.*, 1999.

[3] A. C. Verma, D. Saha and H. Saikia, "Forgery Detection in Offline Signature Verification using Global and Geometric Features," in *International Journal of Computer and Electronics Research*, 2013.

[4] C. R. Prashanth and K. B. Raja, "Off-line Signature Verification Based on Angular Features," in *International Journal of Modeling and Optimization*, 2012.

[5] T. T. a. L. J. V. G. H. Bay, "Surf: Speeded up robust features," in *ECCV (1)*, 2006.

[6] S. Pal, S. Chanda, U. Pal, K. Franke and M. Blumenstein, "Off-line signature verification using G-SURF," in *12th International Conference on Intelligent Systems Design and Applications*, 2012.

[7] V. K. Madasu and C. L. Brian, "An Automatic Offline Signature Verification and Forgery Detection System," in *Pattern Recognition Technologies and Applications: Recent Advances (pp. 63-89). Hershey, PA: Information Science Reference. doi:10.4018/978-1-59904-807-9.ch004*, 2008.

[8] C. Harris and M. Stephens, "“A combined corner and edge detector," Manchester, UK, 1988.

[9] M. S. a. M. I. Marjaneh Safaei, "Social Graph Generation & Forecasting using Social Network Mining,”," in *33rd Annual IEEE International Computer Software and Applications Conference(COMPSAC'09)*, 2009.

[10] P. Beaudet, "Rotationally invariant image operators.," in *4th Int. Joint Conf. Patt. Recog.*, 1978.

[11] H.-T. C. Wei-Ting Lee, "Histogram-based Interest Point Detectors," in *IEEE*, 2007.

[12] K. R. I. K. T. Jasmine Pemeena Priyadarsini, "Bank Cheque Authentication using Signature," in *International Journal of Advanced Research in Computer Science and Software Engineering*, 2013.

[13] J. Malik, R. Dahiya and G. Sainarayanan, Harris Operator Corner Detection using Sliding, International Journal of Computer Applications (0975 – 8887), 2011.

[14] F. Nielsen, "'Harris-Stephens' combined corner/edge detector," 2009.

[15] J. CANNY, "A computational approach to edge detection," in *IEEE Trans. Pattern Anal. Mach. Intell.*, 1986.

[16] C. R. Prashanth, K. B. Raja, K. R. Venugopal and L. M. Patnaik, "Intra-modal Score level Fusion for Off-line Signature Verification," in *International Journal of Innovative Technology and Exploring*

Engineering (IJITEE), 2012.

[17] V. Kiani, R. Pourreza and H. R. Pourreza, "Offline Signature Verification Using Local Radon Transform and Support Vector Machines," in *International Journal of Image Processing (IJIP)*, 2010.

[18] I. S. ABUHAIBA, "Offline Signature Verification Using Graph Matching," in *Turk J Elec Engin*, 2007.

[19] D. Lowe, "Object recognition from local scale-invariant features," in *The Proceedings of the Seventh IEEE International Conference on Computer Vision.*, Kerkyra, 1999.

[20] P. Rajarajeswari and N. Uma, "Offline Signature Verification using Pixel Based Fuzzy Method," in *Journal of Innovative Research and Solutions(JIRAS)*, 2013.