Offline Signature Verification and Forgery Detection Based on Computer Vision and Fuzzy Logic

Gautam S. Prakash, Shanu Sharma
Computer Science & Engineering Department, ASET, Amity University, Noida, Uttar Pradesh, India.

Article Info

Article history:

Received Jul 4, 2014
Revised Nov 10, 2014
Accepted Nov 21, 2014

Keyword:

Artificial Neural Network ANN
Fuzzy Logic
Computer Vision
Forgery detection
Signature verification

ABSTRACT

Automated signature verification and forgery detection has many applications in the field of Bank-cheque processing,document authentication, ATM access etc. Handwritten signatures have proved to be important in authenticating a person's identity, who is signing the document. In this paper a Fuzzy Logic and Artificial Neural Network Based Off-line Signature Verification and Forgery Detection System is presented. As there are unique and important variations in the feature elements of each signature, so in order to match a particular signature with the database, the structural parameters of the signatures along with the local variations in the signature characteristics are used. These characteristics have been used to train the artificial neural network. The system uses the features extracted from the signatures such as centroid, height – width ratio, total area, Ist and IInd order derivatives, quadrant areas etc. After the verification of the signature the angle features are used in fuzzy logic based system for forgery detection.

*Copyright © 2014 Institute of Advanced Engineering and Science.
All rights reserved.*

Corresponding Author:

Gautam S. Prakash,
Computer Science & Engineering Department, ASET,
Amity University, Noida, Uttar Pradesh, India
Email : Gautamsprakash@gmail.com

1. INTRODUCTION

Forgery is a process by which, identity documents of a person are copied or modified by such a person who is not authorized to do so, or are involved is modification for the purpose of deceiving those who view the document about the identity of the status bearer [1].

Signature, from the Latin word "Signare" meaning "Sign" is a stylized handwritten representation of a person's name or an identification mark that a person writes on documents/texts. For many centuries, signatures have been used as an important element in authentication of any person's identity, who is signing the document [2]. The unique characteristics of a person’s signature represent the person's identity and the person's consent for the terms of the document/text. The field of signature authentication is very important and hence the problem of verification and forgery detection is of the utmost importance. Handwritten stylized signatures vary largely from person to person. They differ in their sizes and shapes, and the variations are so much, that for a human being, just by having a glance at the signature, it is very difficult to separate out a genuine signature from a one that is forged.

An automatic signature verification system can either be online or offline. In an online verification system, as the person signs the document/text, the person's signatures are recorded. The merit of such a system is that, a person's dynamic information characteristics can also be accounted. But the problem withsuch a system is that, in reality, most of the documents are already pre-signed [3]. Hence to deal with such situations, an offline verification system is used, which only accounts for the static features of a signature.

Image Processing has found number of applications in the field of forensic examination. Image processing has proved to be very effective tool to analyze thousands of signatures in the database, and apply techniques for detailed analysis such as fuzzy logic and artificial neural network to decrease the amount of time, and increase the effectiveness of the system [4].

For better understanding of further studies, it is important to be acquainted with the basic common concepts such as computer vision technology, and the need for automated signature verification. A brief explanation about them is given below.

1.1. Computer Vision Technology

Computer Vision Technology is used for automating the vision perception process. Computer vision is a field that includes methods for acquiring, processing, analyzing, and understanding images and, in general, high-dimensional data from the real world in order to produce numerical or symbolic information, *e.g.*, in the forms of decisions [5]. Computer vision covers the core technology of automated image analysis which is used in many fields [6]. As a scientific discipline, computer vision is concerned with the theory behind artificial systems that extract information from images. The image data can take many forms, such as video sequences, views from multiple cameras, or multi-dimensional data from a medical scanner. As a technological discipline, computer vision seeks to apply its theories and models to the construction of computer vision systems [5].

1.2. Need of automated Signature Verification

Signature verification is very important in realizing tele-banking and tele-networking systems, where signatures can be used to identify and authenticate a subscriber. An automated verification process would enable banks and other financial institutions to significantly reduce check and money order forgeries, which account for a large monetary loss each year. Reliable signature verification can be of great help in many other application areas such as law enforcement, industry, security control and so on. Handwritten signatures appear on many types of documents such as bank checks and credit slip etc [7][12]. The large volume of such documents makes automatic signature verification desirable. A system for signature verification requires high reliability.

The rest of this paper is organized as follows: - In section II some basic techniques of signature verification and already developed systems are summarized. Section III describes the proposed algorithm, section IV presents the experimental results and section V concludes the paper.

2. RELATED WORK

The problem of signature verification and forgery detection of documents has long been an area of interest in the field of image processing. Many studies have been done till now in order to develop offline signature verification systems using computer vision technology and soft computing techniques [7]. Many researchers are still working on design, development and implementation of an automatic system for fast and much more effective as well as reliable signature verification system. Some already developed system in the problem area are explained below

Rameez Wajid et al. [8] have evaluated the performance of various classifiers for offline signature verification based upon the local binary patterns (LBP) feature set. They have performed the feature vector by dividing the signature images into twelve local regions and forming a code matrix by their LBPs. The authors have investigated the performance of seven classifiers on The FUM-Persian Handwritten Signature Database (FUM-PHSDB) comprising of 20 classes of genuine and forged signatures of depth 20 and 10 respectively. The classifiers considered by them are Support vector Machines (SVM), Least Squares-Support Vector Machines (LS-SVM), Distance Likelihood ratio Test (DLRT), Artificial Neural Network (ANN), Fisher's Linear Discriminant (FLD), Logistics Discriminant and Naive Bayes. Their experimental findings depict that LS-SVM performs the best among the seven classifiers, achieving the Equal Error Rate (EER) of 13%.

Muhammad Imran Malik et al. [9] have evaluated the impact of two state of the art offline signature verification systems which are based on local and global features respectively. The authors have investigate the performance of automated systems on disguised signatures. The systems were evaluated upon the publically available datasets from signature verification competition. The ICDAR 2009 Offline Signature Verification Competition dataset and the ICFHR 2010 4NSignComp datasets were considered. The offline signature verification systems considered for evaluation were Local Features combined with Gaussian Mixture Models (GMMs) and Global Features combined with k-Nearest Neighbour (kNN). In their experiments it was observed that global features are capable of providing good results if only a detection of genuine and forged signatures is needed. Local features are much better suited to solve the forensic signature verification cases when disguised signatures are also involved.

Juan Hu et al. [10] have presented an offline signature verification system using three different pseudo-dynamic features, two different classifier training approaches and two datasets. Three separate pseudo-dynamic features based on gray level: Local Binary Pattern (LBP), Gray Level Co-occurrence Matrix (GLCM) and Histogram Oriented Gradients (HOG) have been used. The classification is performed using the writer dependent Support Vector Machine (SVMs) classifier and Global Real Adaboost method. In their experiments, the results of the Equal Error Rate (EER) of skilled forgery test using the writer-dependent approach obtained were 11.73% for LBP, 11.54% for GLCM and 9.83% for HOG. The combination of the three resulted in EER of 7.66%. The results of EER of skilled forgery test using the writer-independent approach obtained were 13.09% for LBP, 19.33% for GLCM, 13.18% for HOG and combination of all three resulted in EER of 9.94%.

Md. Iqbal Quraishi et al [1] have proposed in their paper an Artificial Neural Network approach which implements an Automated Signature Verification and Authentication system. Their method comprises of various transformation techniques from the spatial as well as frequency domain. It also implements the use of Riplet-II transformation to extract the region of interest. To enhance the image, further it implements the use of Log Polar Transformation. They have implemented a Feed Forward Back Propagation Neural Network for the verification and authentication. They have considered 30 neurons in the hidden layer of the ANN The system proposed by the authors, has the accuracy of 96.15%, with the forgery detection rate of 92%. The False Acceptance Rate (FAR) is found to be 5.28%, and False Rejection Rate (FRR) of 2.56%. The authors have compared their system with other existing system and have found that their proposed system has better performance as compared to others. The drawback is that the test needs to be trained before the implementation, which is time consuming. There can be further improvement is the system with better performance rates.

Othman o-khalifa et al. [2] has reviewed offline signature verification schemes in their paper. They have considered the Artificial Neural Network Technique, and have compared various offline signature verification approaches and their issues. For the pre-processing of the data acquired the have used techniques such as Background Elimination, Noise Reduction, Thinning and Width Normalization. For the purpose of feature extraction, they have considered the global, geometric, texture, mask and grid features. They have explained how the ANN approach works in the signature Verification and what steps are involved. The authors have also pointed out that the main concern of the signature verification system is to provide the high security to access any confidential things those are highly restricted.

3. PROPOSED METHODOLOGY

The proposed approach aims at developing automatic offline signature verification and forgery detection system. Fig. 1 shows the algorithm that is used in order to build the automated signature verification and forgery detection system. The proposed system has been divided into two parts namely:

[1] Training
[2] Testing

3.1. Training Phase:

In the training part of the system, the following steps are performed:

3.1.1. Image Database Creation: The images are collected for training and are stored in a database. The images are collected by scanning them from a physical paper source. The database used is a self-created database which contains signatures of three different people. The database consists of fifteen signatures belonging to each person, and summing up to be forty-five signatures in total. More signatures can be added to the database easily and also the number of signatures per person can also be increased or decreased.

3.1.2. Pre-Processing*:* In this step, each of the scanned signature goes through a series of pre-processing steps which include the following[15]:

[1] Image Resizing: The image is resized to a pre-defined size of 128 x 128 pixels.
[2] Binarization: After resizing the image, the image is binarized, i.e. it is converted to black and white [14].
[3] Thinning: After the process of binarization, the image goes through the process of thinning, i.e. the thickness of the strokes of the signature is thinned down to a single pixel. It is done is order to exclude the variations in thickness of signature which may occur due to the use of different types of pens.

[4] Rotation: The image is then rotated on the basis of the lower most pixels. When a person signs a document, depending on the writing style of the person, there is a certain angle to the signature in which it is done. This process straightens out the signature.

[5] Cropping of the image: After the image is rotated, the excess area around the signature is removed and the image is cropped to the outer most pixels in four directions, i.e. top, bottom, left and right.

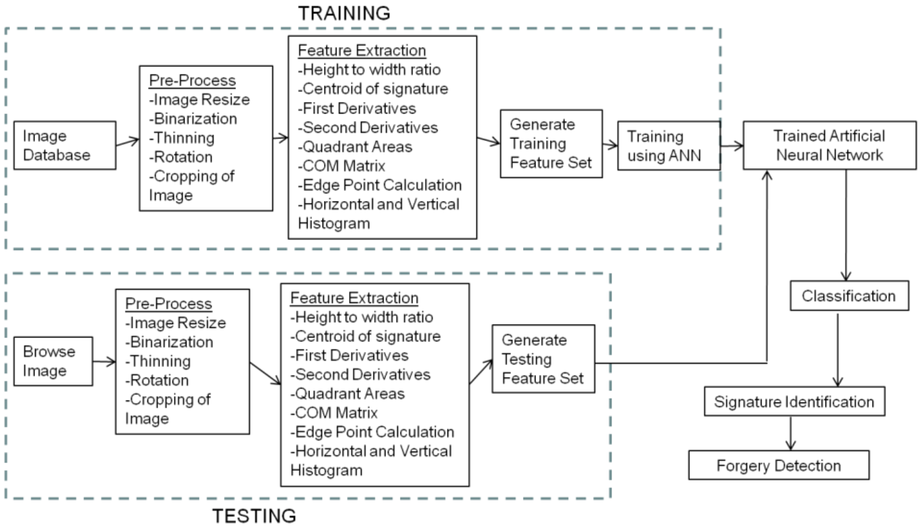


Figure 1. Proposed Algorithm

3.1.3. Feature Extraction: After the image has gone through the pre-processing, various features are extracted from the image. The extracted features out of each image are then stored in a MATLAB file.
Following unique features are extracted from each the images:

[1] Height-Width Ratio: After the image is cropped, height-width ratio of the signature is calculated.

[2] Centroid of Signature**:** The centroid or the barycentre of the image is calculated. The centroid gives the central point of the signature which is a unique signature characteristic. The signature is broken down vertically into two halves, and the centroid of the each half is calculated.

[3] First Derivatives: The first derivatives of the image matrix are calculated row wise as well as column wise.

[4] Second Derivatives**:** After the calculation of first derivatives, the second derivatives of the image matrix are calculated both row and column wise.

[5] Quadrant Areas: The image is broken down into four quadrants, and then the area of the signature pixels in each quadrant is calculated. This area is the area of strokes of the signature in that particular quadrant and does not include the area of the background.

[6] COM Matrix: COM Matrix or Co-Occurrence Matrix refers to the distribution of the co-occurring values at a given offset. It is used to measure the texture on the image. What is does is, as our image is in black and white after the pre-process, that means the image matrix has values either 0 or 1. It looks for pattern distribution of these values and looks where the patterns 00, 01, 11 and 10 occur. The co-occurrence matrix is also calculated for the signature.

[7] Edge Point Calculation: The number of edge points in the signature are calculated which gives a distinct characteristic about the signature.

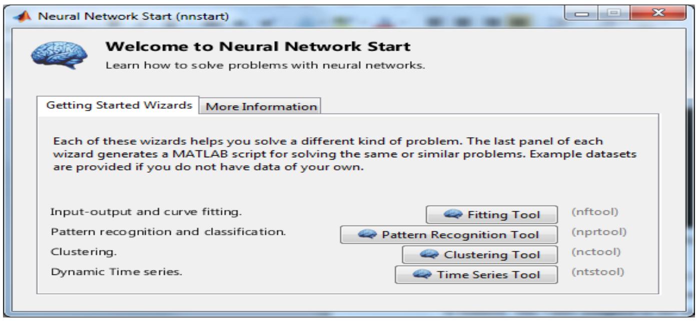
[8] Horizontal and Vertical Histogram**:** Each row and each column of the signature is gone through and the number of black pixels is calculated. The row and the column with the maximum number of black pixels is recorded and used as a feature. All these features give out unique characteristics about the signature and are used for classification of the signatures.

3.1.4. Generate Training Feature Set*:*

In this step, once all the features calculated is saved, then the required output is generated on the basis of which the Neural Network is trained. The vales assigned to the training images can be either 0 or 1. These values along with the values of the features are used to train the ANN.

3.1.5. Training Using ANN*:*

Once the feature values and output values of the images are decided, then the neural network can be trained using neural network tool box as shown in Fig 2.. The command to start neural network is “nnstart” in MATLAB which then opens the toolbox. After the training of system using ANN, the trained artificial neural network is obtained [13].



Neural Network Start (nnstart)

Welcome to Neural Network Start

Learn how to solve problems with neural networks.

Each of these wizards helps you solve a different kind of problem. The last panel of each wizard generates a MATLAB script for solving the same or similar problems. Example datasets are provided if you do not have data of your own.

Input-output and curve fitting. (nftool)

Pattern recognition and classification. (nprtool)

Clustering. (nctool)

Dynamic Time series. (ntstool)

Figure 2. Network Tool Box

3.1. Testing Phase:

This phase is used during the run time implementation of the system. It consists of following steps.

3.2.1. Browse Image:

Unlike the training part where the images are automatically read from the training database, in the testing part, the image is manually selected from the testing database.

3.2.2. Pre- Process:

In the testing part, the image selected goes through the same pre-process steps as in the training part.

3.2.3. Feature Extraction:

In testing part the features of the selected image are calculated and stored. These values are then later used for the classification step. The features extracted in the testing phase are the same as that of the training phase and follow the same process.

3.2.4. Generate Testing Feature Set:

After the features are calculated, they are stored and are used to generate the testing feature set. This feature set is then fed into the trained ANN system.

3.2.5. Signature Identification Using Trained Artificial Neural Network:

The testing feature set generated in the testing part is fed to the trained artificial neural network system that was obtained in the

training phase, selected signature is then classified [13]. Based on the classification done by the system, the signature belongs to which person is identified and is displayed on screen as a message window.

3.2.6. Forgery Detection:

Angle features have been used for the purpose of detection of forgery. The following algorithm is used for the purpose of forgery detection:

[1] Take the bottom left most point as reference point for the angle calculation.

[2] Calculate angle for each pixel of the signature from this reference point.

[3] Divide these angles into 18 categories (0-5, 5-10, ... , 85-90).

[4] Find the average of angles in these categories, thus giving us with eighteen features per signature.

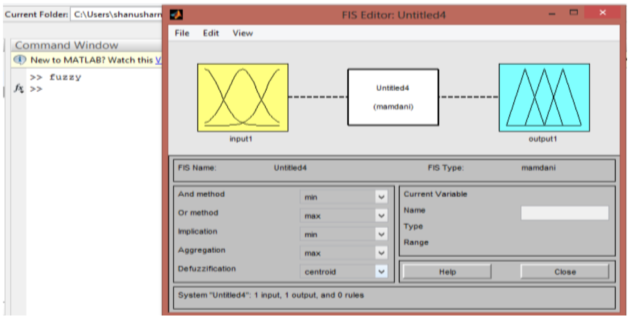
[5] Repeat these steps for 10 samples for each person.

[6] From the above data, a feature vector will be created, which will hold the minimum, maximum and average angle for each category.

Ten samples for each person are taken. Angles from these samples are calculated from the reference point. The angles so obtained are then divided into categories of interval of 5. The categories into which the angles are divided are 0-5, 5-10, 10-15, ... till 85-90. This provides us with the total of 18 categories. The average of angles of each category is taken providing us with total of eighteen features per signature. This data is then used to create a feature vector which holds the maximum, minimum and the average angles for each category.

Table 1. Angle feature vector for 10 samples of a person.

Angle Range	Min	Avg	Max
0--5	0.022175429	0.025656401	0.027006064
5--10	0.082017716	0.084699612	0.086908535
10--15	0.131457778	0.136645891	0.142601292
15--20	0.193417455	0.202624025	0.206745795
20--25	0.251191835	0.255595526	0.263375028
25--30	0.303061442	0.313633147	0.320997045
30--35	0.365394122	0.371463132	0.379375028
35--40	0.421999925	0.426335877	0.431094689
40--45	0.468604918	0.477416644	0.484985632
45--50	0.531222802	0.538884651	0.547915025
50--55	0.583422165	0.596400721	0.605556083
55--60	0.644628013	0.656271089	0.665773436
60--65	0.704056272	0.711330657	0.716529615
65--70	0.755028746	0.76822453	0.778926867
70--75	0.812659726	0.824126934	0.830193581
75--80	0.868147057	0.878395978	0.889457313
80--85	0.920436025	0.937960081	0.965923471
85--90	0	0.888797492	1



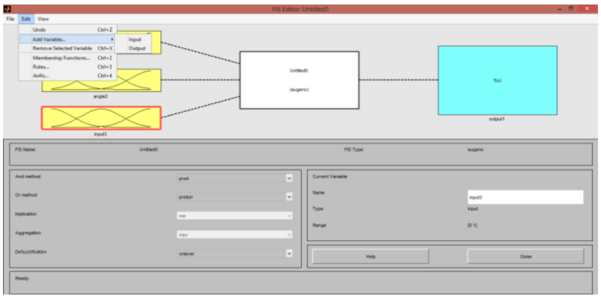
Current Folder: C:\Users\shanusharn

```
>> fuzzy
>>
```

FIS Name: Untitled4
FIS Type: mamdani
And method: min
Or method: max
Implication: min
Aggregation: max
Defuzzification: centroid

System "Untitled4": 1 input, 1 output, and 0 rules

Figure 3. GUI for Fuzzy Toolbox



FIS Name: Untitled5
FIS Type: sugeno
And method: prod
Or method: probor
Implication: min
Aggregation: max
Defuzzification: wtaver
Current Variable
Name: input3
Type: input
Range: [0 1]

Ready

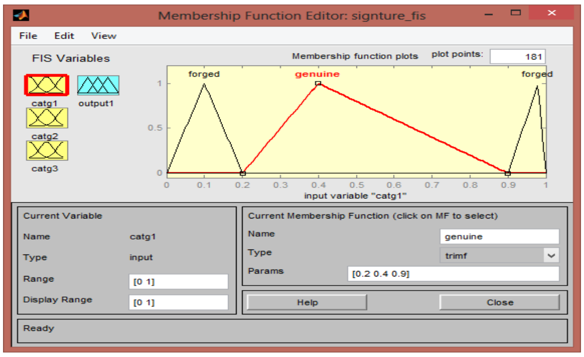
Figure 4. Creation of Angle features as an Input Variable

Once the feature vector file is created, we use this as an input for the Fuzzy Toolbox.

To open the fuzzy toolbox, we write “fuzzy” in the MATLAB command window [11]. Figure 3 shows the interface of the Fuzzy Toolbox.

Once the toolbox has opened, we use the Mamdani Fuzzy Model and add the angle features of the signature as variables. Fig 4 shows the addition of new variables to the toolbox.

Once we have added the variable to the fuzzy toolbox, we define the membership function to each variable according to the variable values. Figure 5 shows the membership function defined for a variable.



plot points: 181

Current Variable

Name catg1

Type input

Range [0 1]

Display Range [0 1]

Current Membership Function (click on MF to select)

Name genuine

Type trimf

Params [0.2 0.4 0.9]

Ready

Figure 6. Fuzzy Rule Generation

After the generation of the rules, the fuzzy model is saved with a name of “signature.fis”. This completes our fuzzy training phase.

For the testing phase, an input image is read and its angle features are extracted. Then the trained fuzzy model is read and is evaluated with the features extracted from the test image file. The value generated from the testing is used to classify the signature as Forged or Genuine.

The various testing results of the algorithm for the test image are presented in the next section.

4. RESULTS AND DISCUSSION

The system is developed with a user friendly GUI. The results at various stages of the application are discussed as below:-

4.1 . Image Selection

The images of two persons are used for testing the system. Total 5 genuine and forged images of each person are used for testing. The test image is selected using the select image button in the GUI as shown in Figure. 7.

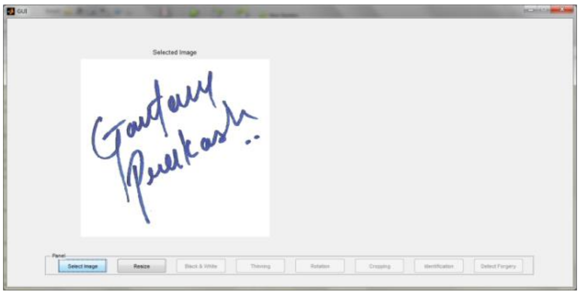


Figure 7. GUI for Testing Phase.

4.2. Pre-processing

The selected image will undergo the pre-processing steps as discussed in section III. The output of these steps can be shown using the Resizing, Binarization, Thinning and Cropping button of GUI. One of these steps is shown in Figure 8.

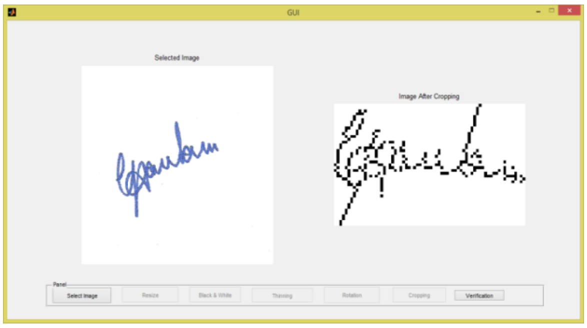
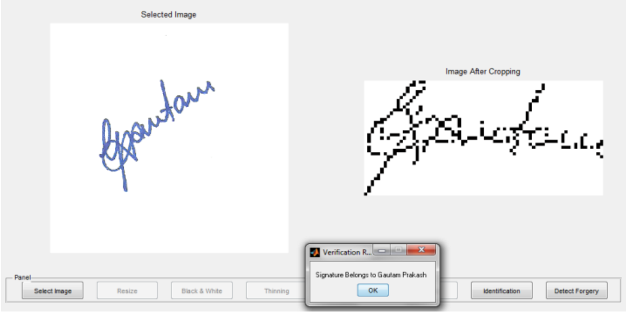


Figure 8. Image obtained after Cropping

4.3. Signature Identification

When the "Identification" button in the GUI is clicked, the already discussed steps of testing phased are done in the background, and the result is showed to the user. The result is a pop up message as shown in Figure. 9 that displays the name of the person to whom the signature belongs.

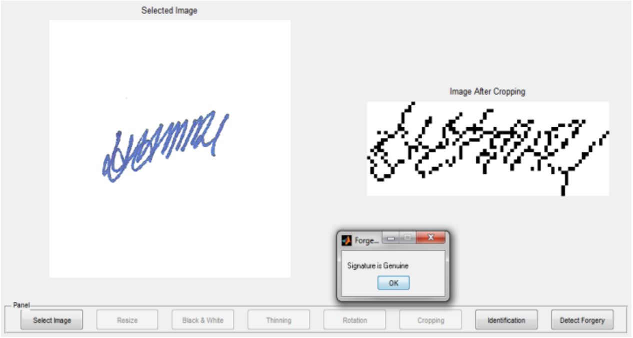


Signature Belongs to Gautam Prakash

Figure 9. Selected Image belongs to Gautam Prakash

4.4. Forgery Detection

Once the signature is identified that to whom it belongs, the next step is to detect forgery. When we click on the "Detect Forgery" button in the GUI, a message is displayed in a pop up window stating the signature is "Genuine" or "Forged". Fig 10 shows that the selected image is Genuine.

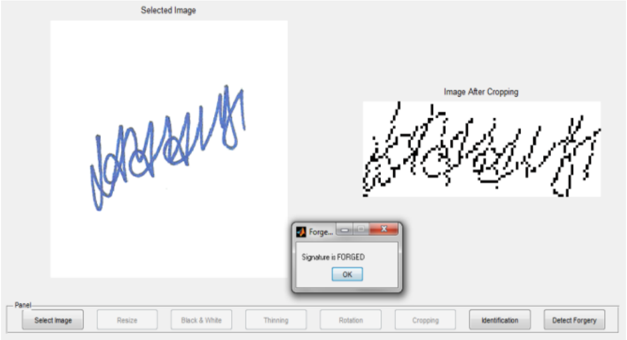


Signature is Genuine

OK

Figure 10. Selected Signature is Genuine

Another result is shown in Figure. 11 for the forged image of second person.



Signature is FORGED

OK

Figure 11. Selected Signature is Forged

5. CONCLUSION

The described system for Automatic Signature Verification and Forgery Detection have numerous applications in various fields like Bank-Cheque processing, ATM access, Document Authentications etc and can be used for the purpose of authenticating the signature.

Further, the variation in personality of signatures, because of age, sickness, geographic location and emotional state of the person actuates the problem. Another problem associated with offline signature verification is that, for security reasons, it is not very easy to make a signature dataset of real documents such as banking documents. These problems can be considered for improving the system

REFERENCES

[1] Md. Iqbal Quraishi, Arindam Das and Saikat Roy (2013), "*A Novel Signature Verification and Authentication System Using Image Transformation and Artificial Neural Netwrok*", Narula Institute of Technology, Kolkata.

[2] Othman o-khalifa, Md. Khorshed Alam and Aisha Hassan Abdalla (2013), "*An Evaluation on Offline Signature Verification using Artificial Neural Network Approach*", International Conference on Computing, Electrical and Electronic Engineering (ICCEEE).

[3] Rameez Wajid and Atif Bin Mansoor, "Classifier Performance Evaluation For Offline Signature Verification Using Local Binary Patterns", Institute of Avionics & Aeronautics, Air University, Islamabad, Pakistan.

[4] Muhammad Imran Malik, Marcus Liwicki and Andreas Dengel, "Evaluation of Local and Global Features for Offline Signature Verification", German Research Center for AI (DFKI GmbH).

[5] Juan Hu and Youbin Chen (2013), *"Offline Signature Verification Using Real Adaboost Classifier Combination of Pseudo-dynamic Features"*, 12th International Conference on Document Analysis & Recognition.

[6] Vaibhav Shah, Umang Sanghavi, Udit Shah, "Off-line Signature Verification Using Curve Fitting Algorithm with Neural Networks", Dwarkadas J. Sanghvi College of Engineering, Mumbai.

[7] M.Nasiri, S.Bayati and F.Safi, "A Fuzzy Approach for the Automatic Off-line Signature Verification Problem Base on Geometric Features", Azad University, Iran.

[8] Surabhi Garhawal and Neeraj Shukla (2013), "A Study on Handwritten Signature Verification Approaches", *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET),* Volume 2, Issue 8, August 2013.

[9] L B. Mahanta, Alpana Deka (2013), "A Study on Handwritten Signature", *International Journal for Computer Applications (0975-8887)*, Volume 79 - No. 2, October 2013.

[10] Pradeep Kumar, Shekhar Singh, Ashwani Garg and Nishant Prabhat (2013), "Hand Written Signature Recognition & Verification using Neural Network", *International Journal of Advanced Research in Computer Science and Software Engineering*, Volume 3, Issue 3, March 2013

[11] Ishita Sharma, Sakshi Goyal and Shanu Sharma, "Sign Language Recognition System for Deaf and Dumb People", *International Journal of Engineering Research & Technology (IJERT)* ISSN: 2278-0181, Vol 2, Issue 4,April-2013, pp. 382-387.

[12] R. Plamondon and S.N. Srihari, "Online and Offline Handwriting Recognition: A Comprehensive Survey", *IEEE Tran. on Pattern Analysis and Machine Intelligence*, vol.22 no.1, pp.63-84, Jan.2000.

[13] M. Blumenstein. S. Armand. and Muthukkumarasamy, *“Off-line Signature Verification using the Enhanced Modified Direction Feature and Neural based Classification,” International Joint Conference on Neural Networks,* 2006.

[14] Lal Chandra, Puja Lal, Raju Gupta, Arun Tayal,Dinesh Ganotra: Improved adaptive binarization technique for document image analysis. VISAPP (1) 2007: 317-321.

[15] Ved Prakash Agnihotri, “Offline Handwritten Devanagari Script Recognition”, *I.J. Information Technology and Computer Science,* 2012, 8, 37-42

BIBLIOGRAPHY OF AUTHORS

Mr. Gautam S. Prakash has done his B.tech in Computer Science and Engineering form CSE Department, Amity School of Engineering & Technology, Amity University, Noida in 2014. His research area includes Digital Image Processing & Computer Vision.

Ms. Shanu Sharma received her M.Tech Degree in Intelligent Systems from Information Technology Department, Indian Institute of Information Technology, Allahabad in 2010. She is currently working as an Assistant Professor in CSE Department, Amity University, Noida, Uttar Pradesh, India. Her Research area includes Digital Image processing and Content based image retrieval.