Offline Handwritten Signature Verification System Using Random Forest Classifier

Maduhansi Thenuwara[#1], Harshani R. K. Nagahamulla[#2]

[#]Department of Computing and information Systems, Faculty of Applied Sciences, Wayamba University of Sri Lanka
Kuliyapitiya, Sri Lanka
[1]hansi.thenuwara@gmail.com
[2]harshaninag@yahoo.com

***Abstract*— This research was conducted to find a feasible solution to verify hand written signatures. The scope has been narrowed down to offline signatures which contains static inputs and outputs. Several classification methods such as Multinomial Naive Bayes Classifier (MNBC), Bernoulli Naive Bayes Classifier (BNBC), Logistic Regression Classifier (LRC), Stochastic Gradient Descent Classifier (SGDC) and Random Forest Classifier (RFC) were implemented to identify the most suitable classifier to verify hand written signatures. The classifiers were trained and tested using a signature database available for the public use. The best performance was obtained from RFC with and accuracy score 0.6. For an average, the system created has been successful in verifying signature images provided with a considerable accuracy level.**

***Keywords*— Offline handwritten signature, classification, algorithms, artificial intelligence, Random forest classifier.**

I. Introduction

Signature has been a distinguishing feature for personal identification through years. This domain includes a history of ages which starts from the Roman Empire. The first legalization of the handwritten signature has been declared by the British government in the 19th century. Signatures have been used for automatic clearing of cheques in the banking industry as well as confirming legality of any document regarding properties, real estate and agreements. Despite an increasing number of electronic alternatives to paper cheques and the other materials which are generally authorized by the handwritten signature, fraud detection has been a real time problem in every sector where the handwritten signatures are used. Since commercial banks pay little attention to verifying signatures on cheques and Debit and Shopping cards, a system capable of verifying whether a signature is a forgery or not will prove beneficial. As signature is the primary mechanism for authentication and authorization in legal transactions and documents, the need for an efficient automated solution for signature verification has increased. Therefore, developing a robust system that automatically authenticates the signatures based on the sample images of original signatures of the owner is the objective of this project.

II. Problem Specification

The area of handwritten signature verification has been broadly researched in the last decades and still remains as an open research problem. The importance is that the wide usage of the signatures in verification processes. Even though there are more advanced biometric verification areas such as iris, fingerprint and palm recognition, still handwritten signature is mostly accepted as it is used in the legal documentations for decades. Currently handwritten signatures are also used on the back side of the debit electronic cards provided by commercial banks which are used in shopping malls and billing machines. In these points the signature is verified with the signature which will be signed by the owner at the point of transaction. The verification is done manually by the retailer or the operator by investigating and comparing the two versions from the human eye. Therefore, this method of verification depends on the vision capabilities and the perception of the viewer. In addition to that the level of comparison will be decided by the viewer.

Other than electronic cards, handwritten signatures are frequently verified in banks when updating accounts and transactions and managing cheques. In most of the times the signatures are compared manually or they are scanned through an optical scanner and verified. The manual process, as mentioned earlier contains a lot of errors and less accuracy. On the other hand, optical scanners only compare the versions of images taken from the original signature and the testing signature as image pixels and it only checks whether the test image is a perfect reflection of the original one.

A handwritten signature can be varied with time when considering a specific owner [1]. This may be because of aging where the pen pressure, timing and stability of the hand movement could be different. Also when an owner gets familiar with a signature with the progression of time, it tends to be signed in a shorter time and with the experience details of the signature could be evolved. Due to these reasons a signature may vary with the time where it may not look as same as it was when the original sample was taken. Therefore, optical scanners are not an effective solution to verify handwritten signatures with their dynamic behavior as the scanning process checks for an exact match of the original signature.

According to the current situation there is no process to verify handwritten signatures with the dynamic behavior where the verification could be done with small variations and differences.

This study aims to find a feasible solution to verify handwritten signatures with a variation. This study will be focused on offline signature verification, characterized by the usage of static (scanned) images of signatures, where the objective is to discriminate whether a given signature is genuine which means produced by the claimed individual, or a forgery which means produced by an impostor.

978-1-5386-2444-9/17/$31.00 © 2017 IEEE

Several types of classifiers such as Multinomial Naive Bayes Classifier (MNBC), Bernoulli Naive Bayes Classifier (BNBC), Logistic Regression Classifier (LRC), Stochastic Gradient Descent Classifier (SGDC) and Random Forest Classifier (RFC) will be implemented and trained using the same dataset. There performances will be compared to identify the best performing classifier in handwritten signature verification.

III. Literature Survey

A signature verification system and the techniques used to solve this problem can be divided into two classes online and Off-line [2]. On-line approach uses an electronic tablet and a stylus connected to a computer to extract information about a signature and takes dynamic information like pressure, velocity, speed of writing etc. for verification purpose. Offline signature verification involves less electronic control and uses signature images captured by scanner or camera. An offline signature verification system uses features extracted from scanned signature image. The features used for offline signature verification are much simpler. In this only the pixel image needs to be evaluated. But, the off-line systems are difficult to design as many desirable characteristics such as the order of strokes, the velocity and other dynamic information are not available in the off-line case. The verification process has to wholly rely on the features that can be extracted from the trace of the static signature images only. [3]

In the terminology of machine learning, classification is considered an instance of supervised learning, i.e. learning where a training set of correctly identified observations is available. The corresponding unsupervised procedure is known as clustering, and involves grouping data into categories based on some measure of inherent similarity or distance. [4]

There are different classifiers that can be used for this application and the followings are the alternatives that was used in this study.

A. Multinomial Naive Bayes Classifier

MultinomialNB implements the naive Bayes algorithm for multinomial distributed data, and is one of the two classic naive Bayes variants used in text classification (where the data are typically represented as word vector counts, although tf-idf vectors are also known to work well in practice).

If a given class and feature value never occur together in the training data, then the frequency-based probability estimate will be zero. This is problematic because it will wipe out all information in the other probabilities when they are multiplied. [5]

B. Bernoulli Naive Bayes Classifier

BernoulliNB implements the naive Bayes training and classification algorithms for data that is distributed according to multivariate Bernoulli distributions; i.e., there may be multiple features but each one is assumed to be a binary-valued (Bernoulli, Boolean) variable. Therefore, this class requires samples to be represented as binary-valued feature vectors; if handed any other kind of data, a BernoulliNB instance may binarize its input (depending on the binarize parameter).

In the case of text classification, word occurrence vectors (rather than word count vectors) may be used to train and use this classifier. BernoulliNB might perform better on some datasets, especially those with shorter documents.

C. Logistic Regression Classifier

In the multiclass case, the training algorithm uses the one-vs-rest (OvR) scheme if the ‘multi_class’ option is set to ‘ovr’ and uses the cross-entropy loss, if the ‘multi_class’ option is set to ‘multinomial’. (Currently the ‘multinomial’ option is supported only by the ‘lbfgs’ and ‘newton-cg’ solvers) [6].

This class implements regularized logistic regression using the liblinear library, newton-cg and lbfgs solvers. It can handle both dense and sparse input. Use C-ordered arrays or CSR matrices containing 64-bit floats for optimal performance; any other input format will be converted (and copied).

D. Stochastic Gradient Descent Classifier

Stochastic Gradient Descent (SGD) is a simple yet very efficient approach to discriminative learning of linear classifiers under convex loss functions such as (linear) Support Vector Machines and Logistic Regression. Even though SGD has been around in the machine learning community for a long time, it has received a considerable amount of attention just recently in the context of large-scale learning [7].

SGD has been successfully applied to large-scale and sparse machine learning problems often encountered in text classification and natural language processing. Given that the data is sparse, the classifiers in this module easily scale to problems with more than 10^5 training examples and more than 10^5 features.

E. Random Forests

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of overfitting to their training set. [7]

Feature extraction and matching plays a very important role in this type of applications. Various methods for both feature extraction and matching are being researched regarding this problem and some of them are;

- Hidden Markov Models
- Spectral Analysis
- Altitude and Direction of Pen Movement
- Wavelets and Backpropagation Neural Networks
- Automatic Signature Verification using a Three-axis Force-sensitive Pen

Although researches has been conducted still there is no common agreement on benchmark databases and protocols in the research community. Researches are on a wide range but not focused on a particular approach. A few of such research are outlined here.

Dash, Nayak and Chattopadhyay used Adaptive Resonance Theory Nets and Associative Memory Net to verify forged signatures. [8] The algorithms were trained with the original and genuine signature and tested with a sample of twelve very similar-looking forged signatures. Their system managed to detect forged signatures with very high accuracy.

Rathi, Rathi and Astya developed a system to verify the offline/handwritten signatures by acquiring a signature using a scanner and taking a boundary of the entire signature and doing pixel wise comparison. [9] This method also managed to identify the signature with high accuracy.

Zhang used a Support Vector Machine technique to identify genuine signatures from forged ones [10]. He used a benchmarking database "Grupo de Procesado Digital de Senales" for training and testing the models. Satisfactory results were obtained with a False Rejection Rate of 2.5% and a False Acceptance Rate 2%.

IV. Methodology

Off-line verification only takes shape and pixel points as a static image. This is sufficient for the requirement and is more feasible because for the online verification, additional hardware requirements may arise such as pressure pads with pens and more powerful machines to process real time data. The aim should be to do the verification with the generic personal computers available in the banking counters and retail outlets. And this would be effective as existing training data can be used.

The operation of a general handwritten signature verification system is given in Fig 1.

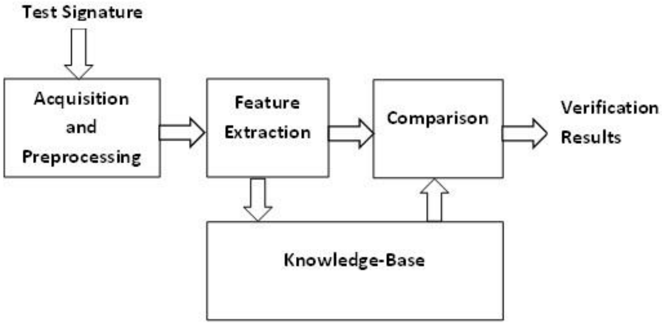


Fig. 1 The operation of a general handwritten signature verification system

The proposed system was developed according to the following steps.

- Process Images
- Get pixel arrays
- Save in data set
- Train the classifier
- Give testing images to identify

Fig 2 explains the process of classifying images used in the system. According to this the images has been converted into black and white and then the pixels were identified as binary outputs based on the black pixels in the image.

These binary values were converted in to a binary array which consists of two dimensions and used as training and testing data.

The MNBC, BNBC, LRC, SGDC and RFC classification methods were tested for the highest accuracy level. The accuracy was calculated from the percentage of correct predictions made by using the training and validation data sets.

The classifier with the best accuracy level was selected for the training and prediction. Using this classifier, the target values were predicted for the testing data and then verified by comparing with the expected output which is the owner's name or ID.

V. Implementation

A. Data Collection

A signature collection for training and testing the classifiers was obtained from ICDAR 2009 Signature Verification Competition (SigComp2009) from the IAPR-TC11: Reading Systems web site (http://www.iapr-tc11.org/mediawiki/index.php?title=ICDAR_2009_Signature_Verification_Competition_(SigComp2009)) [11]. The collection contains simultaneously acquired online and offline signature samples. For this study only the offline signature samples were used.

The signature collection contains: authentic signatures from 100 writers and forged signatures from 33 writers. In total it contains 1953 offline signature files. Out of these 20 signatures were selected containing 12 genuine samples and six forged samples. Fig 3 shows one such sample.

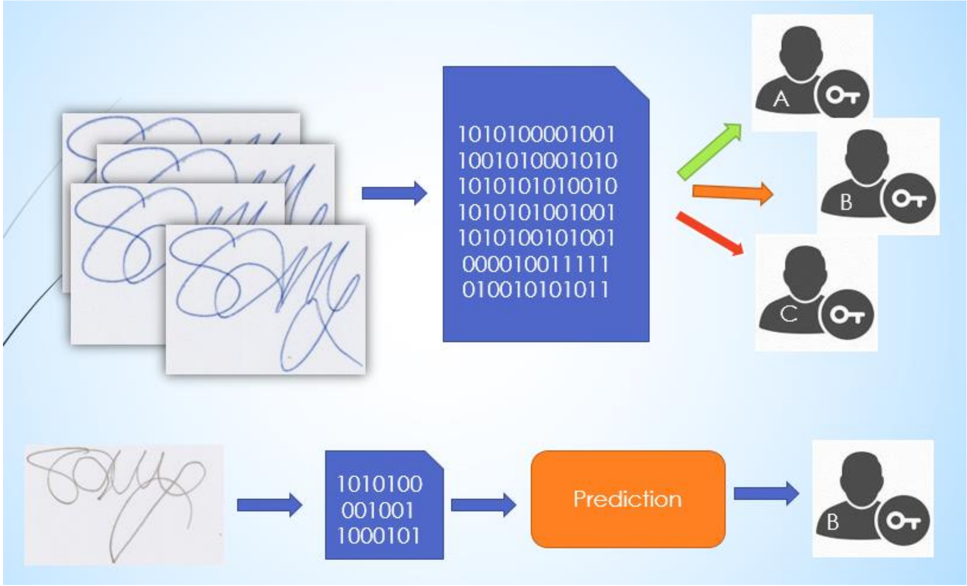


Fig. 2 The process of classifying images in the handwritten signature verification system

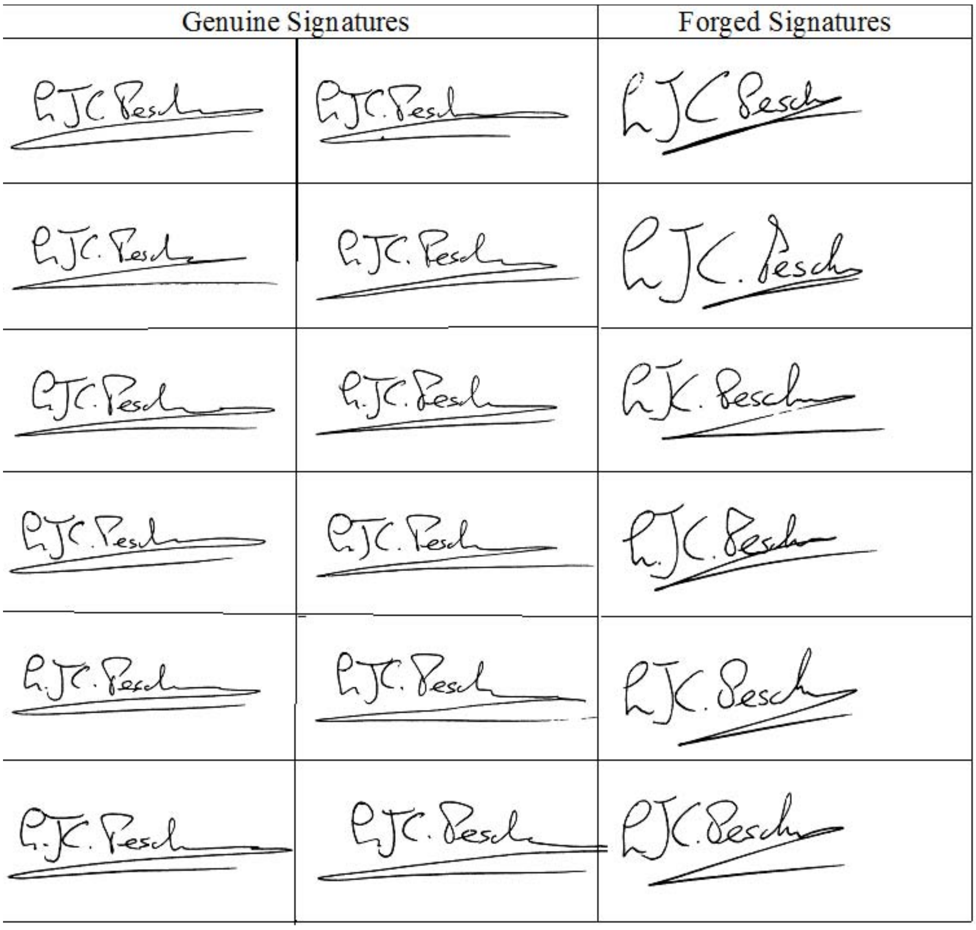


Fig. 3 Sample set of signatures including 12 genuine signatures and six forged signatures.

B. Prepocessing

The images were pre-processed with python image processing libraries. First they were converted into black and white images and then broken down into pixels where the colour of the pixels was taken into a two dimensional binary array.

These data were saved in csv files along with their target value which will be the owner's name in the training data set.

C. Training and Validation

Eight genuine signatures from each sample signature set was selected as training data set and two genuine signatures from each sample were selected as validation data set. The datasets were fed to the MNBC, BNBC, LRC, SGDC and RFC classifiers and they were trained.

D. Testing

The testing data set was fed in to each trained classifier object and the prediction was done by the classifier for the target value. This target value was saved in a csv file as before and then the target was read by the client application for the comparison with the given value by the user. The target output is compared with the predicted output and the probability of prediction which indicated how much the given input belongs to the predicted group is higher than the cut off probability which is defined by the end user. If both conditions were satisfied, the test input will be accepted as verified and if not the signature will be rejected as a forgery.

The accuracy of each method was compared with each other. A separate python program was written calculate and display the accuracy and select the best classifier among them. According to the result the Random forest classifier was selected for the model and the training was continued.

E. The client application

A client application was developed to facilitate the users to verify signatures. The following images will explain the behaviour and appearance of the application. Fig 4 shows the training panel of the client program. In the training panel the user can upload image location and process the images and start training of the data set. When training the system with a new signature the user can give the name or the ID of the owner of the signature.

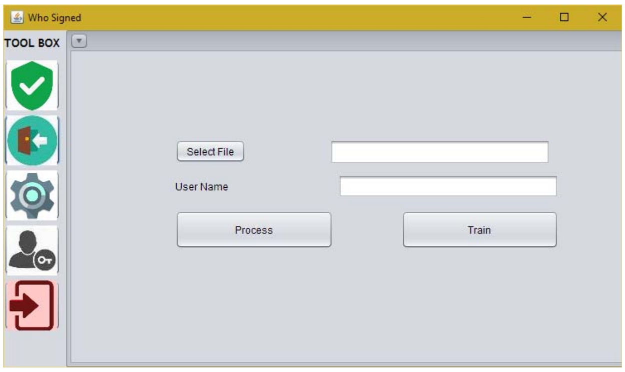
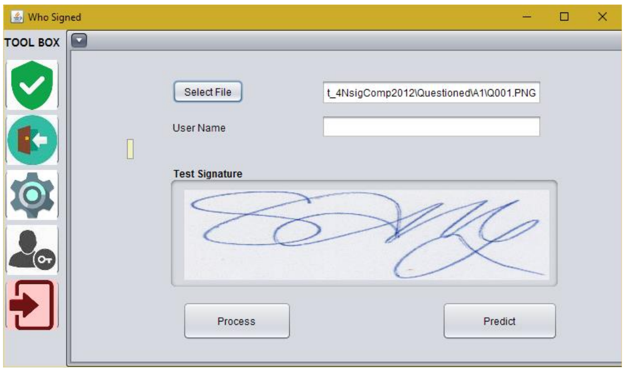


Fig 4 : Trainng Panel

Fig 5 shows the testing panel of the client program. The testing panel allows the user to verify a given signature by predicting the owner of the uploaded image.

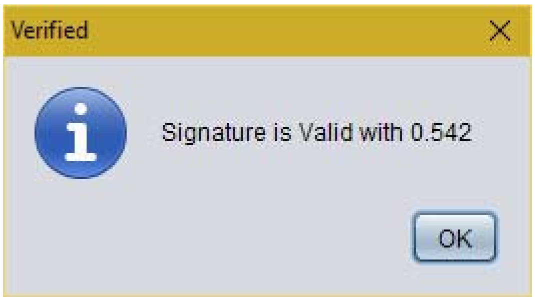


t_4NsigComp2012\Questioned\A1\Q001.PNG

Figure 5 : Verification panel

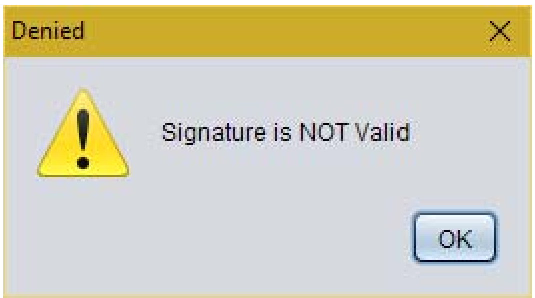
VI. Results

The result of the signature verification of the client program will be given as shown in Fig 6 and Fig. 7.



Signature is Valid with 0.542

Figure 6 : Signature accepted



Signature is NOT Valid

Figure 7 : Signature rejected

The MNBC, BNBC, LRC, SGDC and RFC classifiers were trained and tested with genuine and forged signatures. Their performances were compared with each other. Table 1 shows the accuracy of each classifier in identifying signatures.

TABLE I
Performance Comparison of Classifiers

Classifier	**Accuracy**
Multinomial Naive Bayes Classifier (MNBC)	0.40
Logistic Regression Classifier (LRC)	0.53
Stochastic Gradient Descent Classifier (SGDC)	0.67
Random Forest Classifier (RFC)	0.67

Compared with the performances of other classifiers the RFC and the SGDC classifiers show the best performance with an average accuracy of 67%. The client program was implemented by using RFC classifier.

VII. Conclusion

Verification of handwritten signatures can be done with the use of Random Forest Classifier or the Stochastic Gradient Descent Classifier with an accuracy of 67%. This system supports higher variation of signatures and is more flexible. The accuracy level can be controlled by the cut off probability parameter which can be changed by the user according to their needs. In addition to that this system enables the user to do the image processing and classification together in one application. The system was designed based on object oriented concepts and is designed for change. This system can be used as an integrated tool for different domains such as the internal system of a bank or an inventory and sales management system of a retail shop. Higher the number of samples in the training set, higher the accuracy in signature verification.

On the other hand, since the system is flexible and supports variations, forgeries might be promoted compared to the other methods. As signatures varies from time to time and can be causes of frauds since skilled forgeries can be made, there are higher error rates than other traits. Always the inputs of the system are affected by the physical and emotional state of the owner of the signature. In addition to that this system will contain a large temporal variation.

Acknowledgment

I would like to thank all the lecturers in Department of Computing and Information Systems for the discussions, extensive advice, helpful feedback, and constant support.

My gratitude also goes to my colleges at the university and colleagues. It was with their help that I was able to corporate with knowledge and experiences throughout the project.

And I also should be thankful to the non-academic staff of the Wayamba University. My gratitude goes to my parents and family as well who helped me throughout in many ways.

I was also benefited from the interactions with the references listed in the report as well.

References

[1] D. Joon, S. Kikon. An Offline Handwritten Signature Verification System - A Comprehensive Review. *International Journal of Enhanced Research in Science Technology & Engineering*, Vol. 4 Issue 6, June-2015

[2] S. S. Gharde, K. P. Adhiya, H. G. Chavan. Offline Handwritten Signature Verification Approaches: A Review, *International Journal of Computer Sci ence And Technology*, Vol. 3, Iss ue 2, April - June 2012

[3] Ashwini, S. Bhatia, Handwritten Signature Verification using Neural Network

[4] D. Michie, D.J.Spiegelhalter, C.C.Taylor, *Machine Learning, Neural and Statistical Classification*, 1994.

[5] S. Raschka, *Naive Bayes and Text Classification - Introduction and Theory,* 2014

[6] A. Smola, S.V.N. Vishwanathan *Introduction To Machine Learning* Cambridge University Press 2008

[7] S. Shalev-Shwartz, S. Ben-David. *Understanding Machine Learning: From Theory to Algorithms*, Cambridge University Press. 2014

[8] T. Dash, T. Nayak, S. Chattopadhyay, Handwritten Signature Verification (Offline) using Neural Network Approaches: A Comparative Study, *International Journal of Computer Applications* Volume 57– No.7, November 2012

[9] A. Rathi, D. Rathi, P. Astya. Offline handwritten Signature Verification by using Pixel based Method. *International Journal of Engineering Research & Technology*, Vol. 1 Issue 7, September – 2012

[10] B. Zhang. Offline signature verification and identification by hybrid features and Support Vector Machine, *Int. J. Artificial Intelligence and Soft Computing*, Vol. 2, No. 4, 2011

[11] IAPR-TC11: Reading Systems - http://www.iapr-tc11.org/mediawiki/index.php?title=ICDAR_2009_Signature_Verification_Competition_(SigComp2009)