International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol.9, No.8 (2016), pp.205-222
http://dx.doi.org/10.14257/ijsip.2016.9.8.18

An Offline Signature Verification Technique Using Pixels Intensity Levels

Abdul Salam Shah[1*], M.N.A. Khan[2], Fazli Subhan[3],
Muhammad Fayaz[4] and Asadullah Shah[5]

[1,2]*SZABIST, Islamabad, Pakistan*
[3]*National University of Modern Languages-NUML, Islamabad, Pakistan*
[4]*University of Malakand, KPK, Pakistan*
[5]*International Islamic University Malaysia (IIUM), Malaysia*
[1]*shahsalamss@gmail.com,*[2]*mnak2010@gmail.com,*[3]*fsubhan@numl.edu.pk,*
[4]*hamaz_khan@yahoo.com,*[5]*asadullah@iium.edu.my*

Abstract

Offline signature recognition has great importance in our day to day activities. Researchers are trying to use them as biometric identification in various areas like banks, security systems and for other identification purposes. Fingerprints, iris, thumb impression and face detection based biometrics are successfully used for identification of individuals because of their static nature. However, people's signatures show variability that makes it difficult to recognize the original signatures correctly and to use them as biometrics. The handwritten signatures have importance in banks for cheque, credit card processing, legal and financial transactions, and the signatures are the main target of fraudulence. To deal with complex signatures, there should be a robust signature verification method in places such as banks that can correctly classify the signatures into genuine or forgery to avoid financial frauds. This paper, presents a pixels intensity level based offline signature verification model for the correct classification of signatures. To achieve the target, three statistical classifiers; Decision Tree (J48), probability based Naïve Bayes (NB tree) and Euclidean distance based k-Nearest Neighbor (IBk), are used.

For comparison of the accuracy rates of offline signatures with online signatures, three classifiers were applied on online signature database and achieved a 99.90% accuracy rate with decision tree (J48), 99.82% with Naïve Bayes Tree and 98.11% with K-Nearest Neighbor (with 10 fold cross validation). The results of offline signatures were 64.97% accuracy rate with decision tree (J48), 76.16% with Naïve Bayes Tree and 91.91% with k-Nearest Neighbor (IBk) (without forgeries). The accuracy rate dropped with the inclusion of forgery signatures as, 55.63% accuracy rate with decision tree (J48), 67.02% with Naïve Bayes Tree and 88.12% (with forgeries).

Keywords*: Biometrics, Decision Tree, Forgery, k-Nearest Neighbor, Naïve Bayes, Offline Signature, Online Signatures, Preprocessing*

1. Introduction

Biometric systems have changed the way user identities are secured and verified in our daily lives. They are used to identify the unique attributes of humans and are a well-accepted form of personal identification where the authentication is a top priority. The handwritten signatures are well accepted throughout the world and people are well aware of this method of identification for legal, administrative,

*Corresponding Author

ISSN: 2005-4254 IJSIP

business, online banking, cheque processing and financial transactions in banks and verification of human identity.

Researchers are trying to find out the way to use signatures as biometrics [1] same as iris, thumb impression, facial expressions and so on. The verification of signatures can be divided into online and offline. In online signatures, verification method, the electronic pad is used to get the signature that makes verification easier due to the dynamic features like pressure, speed, the direction of writing and pointing and positions of pen tips. Whereas, in offline signatures, the dynamic features are not available.

The signatures, due to their importance, are extremely exposed, and often can be misused and be forged. Three types of forgeries are associated with the offline signatures: random forgery, simple forgery and skilled forgery [2]. However, from the forensic point of view, another class known as a disguised signature; belongs to the genuine author, but differs the intention of the author, is also important [3].

The signature images or metaphors are the combination of fuzzy lines, curves, and unique symbols, they cannot be considered as a combination of words and letters [4]. For signature, verification, character recognition techniques cannot be used [5]. Moreover, the genuine signatures carry natural variations well-known as intra variability, which makes the distinction of genuine and forgery quite difficult. For the verification of offline signatures, currently different techniques are used which can be categorized into the local features that describe the specific part, and global features describe the entire signature image or combination of both [6].

The handwritten signature has importance in banks, as currently they are using a manual system of signature verification for the authentication and authorization, there is need of an authentic signature verification system that can correctly differentiate between two classes of signatures *i.e.*, genuine and forgery [7]. This paper, will specifically focus on offline signatures. For the comparison of the accuracy rate of online signatures with the offline signature, three statistical classifiers are used. First is a decision tree (J48) [8], which generate decision based on information entropy. Second is Naive Bayes (NB tree) [8], works with the assumption that the value of a specific feature is independent of the value of any other feature, given the class variable and generate decision tree based on the probability, Third is k-Nearest Neighbor (IBk), calculates the distance between two objects and on the basis of that identifies the class of a new object. These statistical classifiers have achieved higher accuracy with medical images [9].

For signature, verification and identification purpose various techniques are used, among them the most prominent and successful reported in literature are Support Vector Machine, Neural Network and k-NN. These learning techniques (verification and identification) can be categorized into two main types, unsupervised and supervised techniques. Most of the researchers prefer supervised learning approaches for signature, verification problem. The supervised learning is the machine learning task of inferring a function from labeled training data [10]. The set of training examples is used as training data in supervised learning. The most successful supervised learning technique for the signature verification is k-NN which is a distance based approach in which the distance between two signature vectors is calculated and on the basis of resultant distance the class of signature is decided [11].

The Support Vector Machines (SVMs) also known as Support Vector Networks, uses supervised learning algorithm for the analysis of data to recognize the hidden patterns that are used for regression and classification [12]. The SVM has proven higher accuracy in signature recognition, as a pattern classification technique used for the solution of multi-class problems. SVM separates two classes (genuine and forgery) by calculating the maximum margin boundary between classes. In Artificial

Neural Networks (ANN), the Multilayer Layer Perceptron (MLP) use supervised learning approach, they are mostly used for pattern recognition [11]. ANN is suitable to find solutions of linear and non-linear problems and for complex classification problems like signature recognition [12].

In this paper, k-NN approach is used due to its success and high accuracy found from the literature review. The other two techniques, *i.e.*, Decision Tree and Naïve Bayes Tree have shown success in Data Mining domain, but very few authors have used these techniques for the signature verification [13]. In this paper, these classifiers are used for the signature verification and identification. The proposed model is suitable for both online and offline signatures. The proposed approach cover both aspects; correctly identifies the owner of the signature, and correctly classifies whether the signature is genuine or a forged.

The rest of the paper is organized as, In Section 2, the Related Work is included, the model is proposed in Section 3, Section 4 covers Experimentation, and following that Section 5 contains Experimental Results. The results are validated in Section 6, In Section 7, Discussion is presented and finally in Section 8, Conclusion and Future work is provided.

2. Related Work

Odeh *et. al.*, in [14], used four features of signature images; skewness, eccentricity, orientation, and kurtosis. The training of the system was carried out with multilayer perceptron's MLPs neural network, as the activation function sigmoid function was used. The output of MLP network is confidence value to compare with a threshold of target signature to be verified. For experiments, the GPDS300 signature database was used.

Tomar *et. al.*, in [15] proposed a model based on directional and energy density features for offline signatures. Aspect ratio (height-to-length) was considered as a global feature. For classification purpose, feed-forward propagation, neural network with hyperbolic tangent sigmoid transfer function was used. The system does not show robustness against the rotation of signatures. With a limited number of training data sets, directional feature results were the best. The proposed system had some drawbacks one it required extra time for training and was found not suitable for large databases, second the random variance was noticed in the FRR.

Parodi *et. al.*, in [16] proposed a circular grid based feature extraction model. In this model, three geometric features were measured; pixel density distribution (xPD), gravity center angle (xAGC) and gravity center distance (xDGC). Fourier transform was used for mapping of features and in addition, to support vector machine SVM, the linear, polynomial and radial basis function (RBF) kernels were used to classify the signatures. The polynomial kernel has shown the best results, with GPDS300 Signature CORPUS. Their study has certain limitations as the girding schemes don't show robustness against the signature rotations, further the identity of the writer in case of forgery remains unknown. Also, the signature with the underlying thin lines which are too long as compared to the remaining body of the signature were not identified correctly because the resulting center of the signature results in poor performance.

Yilmaz *et. al.*, in [17] focused on local histogram features of signature image. For the division of images into zones, the Cartesian and polar coordinate system has been used. Two features histogram oriented gradient (HOG), and local binary patterns (LBP) were used with support vector machines (SVMs) for the classification of the signature image. For training and testing, author used GPDS-300 dataset. The disadvantage of the static grid was uniform scaling through the

elaboration of strokes at starting and end of signatures differ in location, size and orientation that change the global shape of the signature.

Shekar *et. al.,* in [18], proposed Eigen-signature based model. The features from the signature vector were extracted with the Principal Component Analysis (PCA). Euclidian distance between the two vectors was calculated to identify the class of the signature to which it belongs. Author have used a Kammada signature database called MUKOS database.

Ramachandra *et al.,* in [19], proposed model based on the horizontal histogram, horizontal and vertical center, and vertical histogram, edge points of the signature, the signature area, and aspect ratio. The signatures were classified on the basis of k-Nearest Neighbor classifier (Euclidian distance).

Daramola *et. al.,* in [20] focused on the combination of discrete cosine transform (DCT) and hidden Markov model (HMM) features. From the stochastic approaches, HMMs proved very effective in both dynamic and static signals.

Vargas *et. al.,* in [21] measured the robustness of gray level features, distorted by the background complexity, they have used GPDS and MCYT databases. They focused on the histogram of local binary, local derivative and local directional patterns of the measure of texture. The nearest neighbor and SVM along with the histogram kernel GHI and classical RBF kernels were used to evaluate and classify the signatures. The SVM proved more robustness against the distortion of gray levels in the signature images containing the complex background of checks. The LDerivP parameters gave best results.

Kruthi *et. al.,* in [22] used support vector machine (SVM) classifier based model with kernel perceptron algorithm and SMO algorithm and linear and polynomial kernel function to map the overlapping data. The edges of signatures were detected with edge function. They have considered seven different global features; aspect ratio, normalized area, horizontal and vertical profiles, vertical centroid, slant angle edge histogram and edge direction histogram.

Arunalatha *et. al.,* in [23] extracted pseudo dynamic features from the small grids of images and compared them with the principal component variance. The signatures were compared with the help of Eigen value calculated against every signature image.

Baltzakisa *et. al.,* in [24] used global features as signature height, Image area, pure width, pure height, baseline shift, vertical center of signature, horizontal center of signature, maximum vertical projection, maximum horizontal projection, vertical projection peak, horizontal projection peak, global slant angle, local slant angle, number of edge points and number of closed loops. For texture features, concurrence matrix, of the signature image used. The signature images were classified with Multilayer perceptron (MLP).

Various offline and online signature, verification and identification methods were proposed in the literature. The claimed accuracies of offline signature verification models are given in Table 1. Further Quan *et. al.,* in [25], proposed a hybrid approach for online signature verification based on the Hidden Markov Model (HMM) and Artificial Neural Network (ANN). Fierrez *et. al.,* in [26], proposed online signature verification model based on the fusion of both local and global information of signatures. Jain *et. al.,* in [27], proposed online signature verification model and achieved the false rejecting rate of 2.8% and a false accepting rate of 1.6%. Unfortunately, due to complexity and variance of offline signatures, none of the offline signature verification methods have produced satisfactory results in comparison with online signature verification methods in which up to 99.65% accuracies are achieved due to the dynamic features of online signatures [28]. Hence, there is a lot of gap to be filled in the field of offline signature, verification and identification to bring the accuracy rate of offline signatures to the achieved

accuracy rate of online signatures. In this paper an attempt to fill that gap is made, with the contribution of a model based for both online and offline signature verification and classification, while the earlier works have focused on one aspect either verification of the signatures or classification to identify the owner of the signature. For achieving the target, three statistical classifiers are used to achieve the better accuracies with a small number of training samples than the classifiers previously appeared in literature for the signature verification.

Table 1. Results Claimed By Authors

Author	Technique Used	FRR	FAR	EER	Accuracy
[14]	MLP	-	-	21.20%.	78.80%
[15]	Energy Density Method	44.00%	42.00%	43.00%	57.00%
	Directional Features	20.00%	40.00%	30.00%	70.00%
	Directional Feature with Energy Density Method	30.00%	28.00%	29.00%	71.00%
[16]	Graphometric Features, Circular Grid, SVM	7.82%	0.49%	4.21%	95.79%
[17]	(HOG), (LBP),SVM 5 Signatures	-	-	17.65%	82.35%
	12 Signatures	-	-	15.41%	84.59%
[18]	Eigen-Signature, Pixels Intensity Levels, 5 signatures	-	-	21.45%	78.55%
	10 Signatures	-	-	14.33%	85.67%
	15 Signatures	-	-	8.78%	91.22%
[19]	Euclidian Distance	5.40%	4.60%	-	-
[22]	Kernel Perceptron	6.15%	4.82%	27.72%	72.28%
	SMO algorithm	7.16%	6.57%.	27.72&	72.28%
[23]	Eigen Value, PCA	-	-	17.08%	82.92%
[24]	Global Grid and Texture Features, Neural Network	3.00%	9.81%	19.19%	80.81%

The Table 1, contain results claimed by references included in related work. False Rejection Rate (FER); when a genuine signature got rejected wrongly. False Acceptance Rate (FAR); when a forgery got accepted wrongly and Equal Error Rate (EER) defined as threshold values for its false acceptance rate and its false rejection rate, and when the rates are equal, the common value is referred to as the equal error rate (EER). The value indicates that the proportion of false acceptances is equal to the proportion of false rejections. The lower the equal error rate value, the higher the accuracy of the biometric system [29].

3. Proposed Model

The proposed model is based on the pixel's intensity levels, for the classification of offline signatures into genuine and forgery and identification of users. For the classification three statistical classifiers are used *i.e.*, Naïve Bayes Tree, K-Nearest Neighbor and Decision Tree. The selection of classifiers is based on their robustness

against complex problems and accuracy with limited training data [8]. The proposed model is divided into two portions *i.e.*, Offline Model Figure 1 (a), and Online Model Figure 1 (b). The offline model is for the offline signature verification and the online model for online signature verification. The offline model has few additional stages than the online model; the reason for adding additional stages is due to the complexity of offline signatures and database types used. Two databases were considered for the experiments *i.e.*, ATVS ATVS-SSig DB database contains dynamic features of online signatures. The signature database is divided into five vectors *i.e.*, x-coordinate, y-coordinate, timestamp, pen-ups and pressure function [30-31] and the MCYT offline signature database contains .bmp images of 75 user signatures against every user, 15 genuine and 15 signatures are present [32-33]. The signature databases have been acquired from the ATVS group website, they offer the databases for non-commercial research after filling the agreement. The offline model has pre-processing step; in preprocessing the unnecessary outer white regions of signature images has been cropped and the images have been resized to make them of uniform size. The original size of the signature image was approximately 850x360, after cropping the sizes of image became non uniform, we have converted them into uniform size of the 120x120 after that the signature images were converted into pixel intensity levels. In the feature matrix generation; signature image intensity levels were then converted into (120x120) matrix, the matrix is converted to Signature1.arff. Total 75% of the data set is used for training of the classifier. Both 10 fold cross validation and 75% training split has been used for training and testing. The signature images were scanned and converted into digital form and they contain some kind of noise. In post-processing Discrete Wavelet Transform (DWT) has been applied on offline signatures. DWT has sets of two functions; scaling functions and wavelet functions, which are associated with the low pass and high pass filters, respectively. The scanning of hard paper images into digital form adds some noise into images. The noise reduces the accuracy rate of verification that's why the noise should be reduced to minimum level for reducing noise different filters are mostly used. The use of low pass filter reduces the noise and images became clear for feature extraction which ultimately reduces the false acceptance rate and increases the accuracy rate of the signature verification system. The de-noising is a common technique to remove the noise from the image, mostly the low pass filter is used for the removal of that kind of noise which efficiently removes the noise but it blurs the images. For testing the classifier 25% of the data set is used and finally in the verification step the user of signature will be identified as well as the signatures will be classified into genuine and forgery. The stages of online signature, verification model are less than the offline because the online signature database was already in the form of vectors which later converted to comma separated Signature2.arff, using Excel in the preprocessing stage. For the training, three classifiers (Naïve Bayes, Decision Tree and k-nearest neighbor) have been used. The training was carried with 10 fold cross validation and 75% training split, for the testing remaining 25% testing data set has been used with already trained model, and in the verification stage the identity of the users of signatures were identified. Further, the detailed description of the model is provided in the subsequent sub-sections.

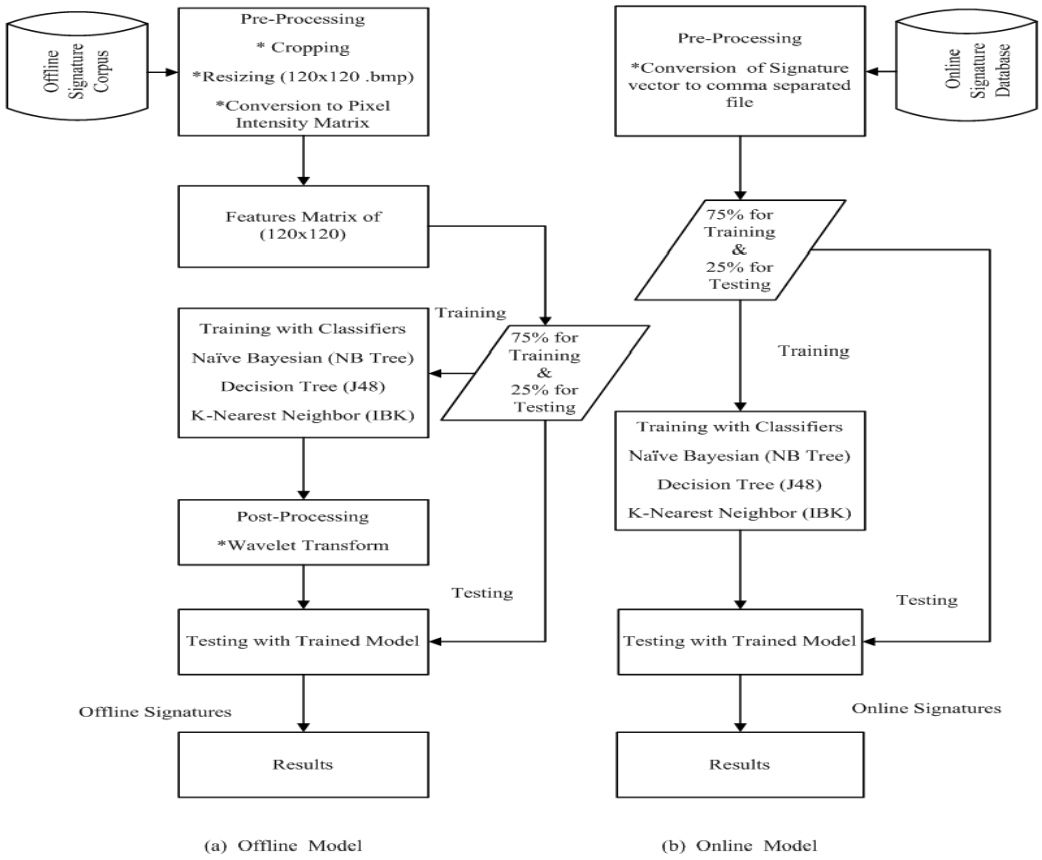


Figure 1. Offline & Online Signature Identification and Verification Model

3.1. Datasets Description

ATVS-SSig DB and MCYT 75-offline signature corpus has been used for experimentation. The detailed description of databases is given as under:

Online Signature Database: The online signature database, ATVS-SSig DB [30-31] is freely available at biometric recognition group-ATVS for non-commercial research, the database contains two datasets *i.e.*, Dataset 1: DS1_Modification_TimeFunctions and Dataset 2: DS2_Modification_LNparameters, in this paper the Dataset 2 is used for experimentation: It contains the online signatures of 350 users having 25 signature samples against each user. The signature database is divided into five vectors *i.e.*, x-coordinate, y-coordinate, timestamp, pen-ups and pressure function.

Offline Signature Database: The second data set used for experimentation is offline signature sub corpus of Ministerio De Ciencia Y Tecnologia (MCYT) large database containing signatures of 75 individuals. The sub corpus contains 15 genuine and 15 forgeries against each signer and total of 2250 signatures [32-33].

The database is freely available for non-commercial research at http://atvs.ii.uam.es/mcyt75so.html. The database has been acquired after completing the license agreement requirements. The database contains .bmp images having a size of 850x360. The forgeries of each user are denoted with ‘f’ in their image name *e.g.*, "0003_0002f00" and the naming of the genuine signature are named like, *e.g.*, "0002v00".

3.2. Preprocessing

The preprocessing is necessary because the images cannot be directly loaded into Weka tool for feature extraction and classification. The three preprocessing steps were carried out for offline signature images. The all three steps are not applied on online signature because the database of online signature is already in the form of vectors that can be simply converted to comma separated .raff file. The first step was cropping, second resizing and the third conversion to pixels intensity levels. Detailed description of these three steps is given bellow:

Cropping: The signature images in MCYT-75 offline signature corpus contain unwanted white regions that were cropped with Microsoft Paint.

Resizing: The signature images in MCYT-75 signature corpus have bigger size. The size of images was reduced to (120x120) so that they consume less amount of memory and the resultant matrix containing intensity values will be smaller as compared to the original image matrix. The size of the image was reduced with Matlab.

Conversion to Pixel Intensity Matrix: The signature images having the size of (120x120) are stored in a Matrix. The intensity values from matrix were then copied to Excel and have assigned every instance, a class label *i.e.,* user to which signature belongs (genuine) and forged.

3.3. Post Processing

The post processing step has been carried out by applying discrete wavelet transform DWT (Haar Wavelet) to remove the noise and for the enhancement of the signature images for better feature extraction. Wavelet transform domain has a de-nosing algorithm for balancing the de-noising; it reserves the edges of the image and other features. Multi-resolution analysis provides an advantage that the unnoticed features of signature images at one resolution might be detected with other resolution [34]. Ribeiro *et. al.,* in [35] have used DWT for the evaluation of horizontal, vertical and diagonal pixels variations. The gravity centers of the frequency amplitudes are calculated for every orientation, and those values represent signature regions where those frequency intervals are most present.

Discrete Wavelet Transform (DWT): The wavelets are mostly used in image processing to cover spatio-frequency domain by scaling and translating basic function [36]. A wavelet $\psi(t)$ is a function produced by shifting b and scaling by a a basic function $\psi(t)$ called mother wavelet and given by (1).

$$\psi_{ab}(t) \equiv \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right) \tag{1}$$

The Fourier Transform of this function can be given by (2).

$$\psi_{ab}(w) = \sqrt{a}\psi(aw)e^{-jbw} \tag{2}$$

The DWT allows to evaluate signals at different frequency bands and resolutions by decomposing the signal into rough approximations and detailed information for deep and precise evaluation. DWT has the group of scaling and wavelet function. They are related with a low pass and high pass filters, respectively. The signals decompose into various frequency bands by applying high pass and low pass filters in the time domain signal [34]. The wavelets can be given by (3).

$$wf(a,b) = \frac{1}{\sqrt{a}}\int_{-\infty}^{+\infty} f(x)\psi\left(\frac{x-b}{a}\right)dx \tag{3}$$

The choice of wavelet plays an important role in the image processing field. The Haar wavelet supports two filters high pass filter and low pass filter, the Haar wavelet has an advantage that it takes less computation and reduces the complexity. We have chosen Haar wavelet for image analysis in this paper. The Haar wavelet is the sub category of Discrete Wavelet Transform (DWT) [35]. The Stankovic *et. al.*, in [36] presented detailed calculations, the description and functioning of Haar Wavelets. The Harr wavelet can be given by (4) and (5) [34].

$$\psi_h(x) = \begin{cases} 1 & 0 < x \le 0.5 \\ -1 & 0.5 \le x < 1 \\ 0 & else \end{cases} \tag{4}$$

$$\phi_h(x) = \begin{cases} 1, & 0 \le x \le 1 \\ 0, & else \end{cases} \tag{5}$$

The above functions can be represented with a graph as in Figure 2, [34].

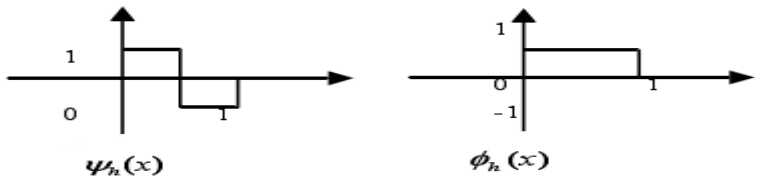


Figure 2. Haar Wavelet Function and Scale Function

3.4. Classifiers

Three statistical algorithms, Decision Tree (J 48), Naïve Bayes (NB tree) and k-Nearest Neighbor (IBk) have been used for verification and classification of both the online and offline signatures [8]. The detailed description and working of the classifiers is given as under:

Decision Tree (J48): Decision trees are generated with an algorithm through continuous recursive portioning. With some criteria, like mutual in formation [37] using (6).

$$MI(X,Y) = \iint p(x,y) \log \frac{p(x,y)}{p(x)\,p(y)} dxdy \tag{6}$$

The p(x, y) is the joint probability density of X and Y, and p(x) and p(y) are marginal probability densities of X and Y, respectively [38].

The other criteria can be Gain-ratio (7) or Gini index using (9), a single attribute is selected as a root of the tree.

$$GR(S,A) = \frac{Gain(S,A)}{IntrI(S,A)} \tag{7}$$

The Intrinsic information (Intr) of a split is required to calculate the amount of information required deciding which branch of tree and instance belongs to. The Intrinsic information can be given by (8).

$$IntI(S,A) = -\sum_i \frac{|S_i|}{|S|} \log \left(\frac{|Si|}{S} \right) \tag{8}$$

The Gini Index is an alternative of measuring information gain and can be given by (9), it measures impurity instead of entropy. The average Gini Index can be given by (10), it is used instead of an average entropy of information gain.

$$Gini\ (S) = 1 - \sum_i p_i^2 \tag{9}$$

$$Gini\ (S, A) = \sum_i \frac{S_i}{S}.Gini\ (S_i) \tag{10}$$

The division of the data is carried out on the basis of the test root, this recursive process repeats for every child until a full tree is generated [8]. Weka has java implemented C 4.5 algorithm which is the advanced version of ID3 [13] algorithm introduced by Ross Quinlan named J 48 statistical classifier after its implementation in java. The splitting criterion of J 48 is the normalized information gain; the tree will be generated on the basis of the attribute having the highest normalized information gain. The Entropy measures uncertainty in the data, a set having S samples with the C number of classes, the Entropy of that can be given by (11), Pc denotes the probability of sample/element of S belongs to class C.

$$H(S) = -\sum_{c \in C} Pc \log_2 Pc \tag{11}$$

The information gain (IG) of a feature F can be given by (12).

$$IG(S, F) = H(S) - \sum_{f \in F} \frac{|S_f|}{|S|} H(S_f) \tag{12}$$

The number of elements of S with feature F having value f is denoted by Sf while IG(S, F) measures the increase in certainty of S when the value of F is known.

Naive Bayes Classifier (Bayesian Tree): In NB Tree the leaf nodes do not predict the single class, but they are Naïve Bayes categorizers except, that the Naïve Bayes tree is similar to recursive portioning schemes. By using the standard entropy minimization technique, some threshold is chosen for continuous attributes similar like decision trees. Naïve Bayes Tree produces a decision tree at the leaves with Naive Bayes classifiers [8]. The Naive Bayes works with the supposition that every feature has its own independent value that has no concern with the values of other features in the class variable. The NB tree based model has the ability to be trained under supervised learning setting very efficiently with some type of probability. Naïve Bayes classifiers have a plus point that they can even predict the necessary parameter on less quantity of data for training and give better classification accuracy. The dataset was limited for experimentation and in limited data, the Naïve Bayes is the best choice to be used for the classification of the signature into genuine and forgery [8]. The conditional probability is given using (13).

$$P(B / A) = \frac{P(A \cap B)}{P(A)} \tag{13}$$

The Bayes formula is based on the expression P(B) = P(B|A)P(A) + P(B|Ac)P(Ac), which simply states that the probability of the event B is the sum of the conditional probabilities of event B given that event A has or has not occurred. For independent events A and B, this is equal to P(B)P(A) + P(B)P(Ac) = P(B)(P(A) + P(Ac)) = P(B)(1) = P(B), since the probability of an event and its complement must always sum to 1 [38]. The Bayes formula is given as (14).

$$P(A/B) = \frac{P(B/A)P(A)}{P(B/A)P(A) + P(B/A^{C})P(A^{C})} \quad (14)$$

k-NN: k-Nearest Neighbor classifier (IBk): Finds group of k object at the same time from training data and it does not exactly match with the test object (like rote classifier), but they do match closely with the test object, and on the basis of that predicts labels for objects and also predict the class in the neighborhood to which the chances are more that objects may belong.

This approach has three main fundamentals: labeled objects, *e.g.*, a set of stored instances of signatures, a distance metric on the basis of which the distances of objects will be calculated using, and the value of k.

If some new object arrives the labeled object distance to the new object's distance is calculated, the k-Nearest Neighbors are identified, and on the class labels of k-Nearest Neighbors the class of the new object will be decided [8].

The k-NN is also called as a non-parametric approach, it does not learn from the training data, but it utilizes the training data at the time of testing to make predictions. The k-NN commonly uses Euclidean distance to measure the distance between real-valued features ($xi \in R^{D}$) [12].

The distance d among two points (x1, y1) and (x2, y2), can be calculated with the Euclidian distance using (15) and (16).

$$d = \sqrt{(x2 - x1)^{2} + (y2 - y1)^{2}} \quad (15)$$

$$d(xi, xj) = \sqrt{\sum_{m=1}^{D} (x_{im} - x_{jm})^{2}} = \sqrt{\| xi \|^{2} + \| xj \|^{2} - 2x^{T}{}_{i} xj} \quad (16)$$

4. Experimental Setup

Experimentation was carried out using Matlab R2015a and Weka 3.6. The pre-processing of signature images was carried out with Matlab and after pre-processing the signature images were classified with Weka 3.6.

Online Signature Experimentation Setup: The criteria of experiments with online signature were same as offline except wavelet transform which did not performed here because the online database was already in the form of x, y matrix which later converted into .arff file. In online signature, signatures of 10 users were tested in the first step, then with 20 up to 50 users (5 signatures against every user and no synthetic signatures were included). Three classifiers were used for classification, *i.e.*, Decision Tree (J48), Naïve Bayes Tree and k-Nearest Neighbor. The settings of classifiers were default the reason for selecting default setting is that other setting of classifiers like by increasing the number of k in k-NN, changing the parameters like tree size, pruning and un-pruning with Decision Tree, has not provided the results as good as with default settings. The training time of the Naïve Bayes Tree was high, among the three classifiers. The value of k=1 is used during experiments, the increase of k decreased the accuracy rate. Experimentation was performed with 10 fold cross validation (except with 50 users Naïve Bayes Tree shown low memory heap and so tested with 5 folds) and 75% training and 25% testing criteria.

Offline Signature Experimentation Setup: We have randomly chosen 5 users (5 genuine signatures against every user no forgery signatures included) from the preprocessed dataset and supplied to the classifiers *i.e.*, Decision Tree (J48), Naïve Bayes Tree and k-Nearest Neighbor. Then increased the number of users and

performed the same steps again with 10 users. In the next step included forgeries (3 forgeries against each user) and performed the same steps with 5 and 10 users. The experiments were performed by 10 fold cross validation (except with 10 users with forgery Naïve Bayes shown low memory heap and tested with 5 folds) and 75% split criteria. Dataset of offline signature has images containing noise due to scanning and further the signature images contains thin lines which can be visible with the wavelet transform. For removal of noise, discrete wavelet filter was applied which available in Weka 3.6 tool. During experiment default setting of classifiers was used.

5. Experimental Results

Online Model Results: The Table 2, contains experimental results achieved with the online model with ATVS-SSig database, under different evaluation criteria, like by changing the number of training samples, changing the classifiers, changing the parameters of classifiers, changing the training testing percentage, performing experiments with cross-validation. Among all the classifiers the J 48 has shown the fastest training time, as well as the accuracy rate of Decision Tree (J 48) with 10 fold cross validation was 99.90% and Error Rate of 0.10%. Similarly, J 48 accuracy rate with 75%, training, and 25% testing split was 99.87% and Error Rate of 0.13%. With the same criteria supplied the same dataset to the Naïve Bayes Tree and achieved accuracy rate of 99.82% and error rate of 0.18% with 10 fold cross validation and with 75% training and 25% testing split the accuracy rate reduced to 99.81% and Error Rate increased to 0.19%. The results of k-Nearest Neighbor (IBk) were low among the three classifiers, with 98.11% accuracy rate and error rate of 1.89% with 10 fold cross validation and with 75% training and 25% testing split 98.03% accuracy rate and 1.97% error rate. The increase in number of users has decreased the performance of k-Nearest Neighbor in online signatures, the decrease in performance is noticed due to the similar characteristics of every signature, but surprisingly with offline signatures the accuracy rate of k-Nearest Neighbor is higher among the three classifiers.

Table 2. Results with Online Database

No of Users	**Decision Tree J 48**		**Naïve Bayes Tree**		**k-Nearest Neighbor**	
10 Fold Cross Validation						
	Accuracy	Error Rate	Accuracy	Error Rate	Accuracy	Error Rate
10	99.94%	0.06%	99.82%	0.18%	99.01%	0.99%
20	99.88%	0.12%	99.82%	0.18%	98.56%	1.44%
30	99.90%	0.10%	99.85%	0.15%	97.96%	2.04%
40	99.89%	0.11%	99.85%	0.15%	97.67%	2.33%
50	99.90%	0.10%	99.78%	0.22%	97.36%	2.64%
Average	**99.90%**	**0.10%**	**99.82%**	**0.18%**	**98.11%**	**1.89%**
75% Training and 25% Testing						
10	99.95%	0.05%	99.89%	0.11%	99.38%	0.62%
20	99.80%	0.20%	99.79%	0.21%	98.32%	1.68%
30	99.83%	0.17%	99.80%	0.20%	97.80%	2.20%
40	99.89%	0.11%	99.83%	0.17%	97.29%	2.71%
50	99.87%	0.13%	99.76%	0.24%	97.38%	2.62%
Average	**99.87%**	**0.13%**	**99.81%**	**0.19%**	**98.03%**	**1.97%**

Offline Model Results: The Table 3 contains the results achieved with the offline model with the MCYT-75 sub corpus database. The experimented are carried in two phases with the database, *i.e.*, in the first step, tested 5 and 10 users respectively, without including forgery signatures of the users. In the second step added 3 forgery signatures against each user and repeated the same procedure with 5 users and then 10 users. Without including forgery signatures, Decision Tree (J 48) has shown 64.97% accuracy rate and 35.03% error rate with 10 fold cross validation and 63.30% accuracy rate and 36.70% error rate with 75% split.

The Naïve Bayes Tree has shown 76.16% accuracy rate and 23.84% error rate with 10 fold cross validation. The results reduced to 73.54% accuracy rate and 26.46% error rate with 75% split. The accuracy rate of k-Nearest Neighbor was 91.91% and error rate of 8.10% with 10 fold cross validation and with 75% split the results reduced to 90.93% accuracy rate and 9.03% error rate.

By including 3 forgery signatures (against each user) achieved accuracy rate of 55.63% and error rate of 44.37%, with Decision Tree (J 48) by 10 fold cross validation and with 75% split the results reduced to accuracy rate of 53.27% and error rate increased to 46.73%. The Naïve Bayes has shown 67.02% accuracy rate and 32.98% error rate with 10 fold cross validation. The results reduced to 65.54% accuracy rate and 34.46% error rate with 75% split. The accuracy rate of k-Nearest Neighbor was 88.12% and error rate of 11.88% with 10 fold cross validation and with 75% split the results reduced to 84.98% accuracy rate and 15.02% error rate. Among three classifiers, the k-Nearest Neighbor results were higher with offline signatures.

Table 3. Results with Offline Database

No of Users	Decision Tree J 48		Naïve Bayes Tree		k-Nearest Neighbor	
10 Fold Cross Validation (without forgery)						
	Accuracy	Error Rate	Accuracy	Error Rate	Accuracy	Error Rate
5	68.4%	31.6%	80.07%	19.93%	92.23%	7.77%
10	61.53%	38.47%	72.25%	27.75%	91.58%	8.42%
Average	**64.97%**	**35.03%**	**76.16%**	**3.84%**	**91.91%**	**8.10%**
10 Fold Cross Validation (with forgery)						
5	58.21%	41.79%	70.79%	29.21%	88.04%	11.96%
10	53.05%	46.95%	63.25%	36.75%	88.20%	11.80%
Average	**55.63%**	**44.37%**	**67.02%**	**32.98%**	**88.12%**	**11.88%**
75% Training and 25% Testing (without forgery)						
5	67.87%	32.13%	77.07 %	22.93%	92.53%	7.47%
10	58.73%	41.27%	70.00%	30.00%	89.33%	10.67%
Average	**63.30%**	**36.70%**	**73.54%**	**26.46%**	**90.93%**	**9.07%**
75% Training and 25% Testing (with forgery)						
5	55.75%	44.25%	68.08%	31.92%	84.58%	15.42%
10	50.79%	49.21%	63.00%	37.00%	85.38%	14.63%
Average	**53.27%**	**46.73%**	**65.54%**	**34.46%**	**84.98%**	**15.02%**

6. Results Validation

The online model has shown better accuracy than the proposed model in [28] which claimed 99.65% with 50 user's online signatures. Our model has shown 99.90% accuracy with online signatures of 50 users with decision tree (J 48), and

99.82% with NB tree. The results of the third classifier, k-Nearest Neighbor was 98.11% accuracy rate not promising, as compared to the decision tree (J48) and the NB tree and the results claimed by [28].

The results achieved with the proposed offline model with k-Nearest Neighbor are better than in [39, 40 and 41]. The error rate claimed by [39] with 5 and 10 users are 23.78% and 22.13% respectively.

The error rate claimed by [40] with 5 users is 13.50%, further error rates claimed by [41] with LBP features with 5 and 10 users were 12.82% and 10.68% respectively, and with GLCM features claimed error rates with 5 and 10 users were 16.27% and 12.65%, respectively.

The proposed model has reduced the error rate with 5 and 10 users to 11.96% and 11.80% with an accuracy rate of 88.04% and 88.20% respectively with k-Nearest Neighbor approach. The results of Decision Tree (J48) and Naïve Bayes Tree were not satisfactory that's why not compared in this section. The comparison of the accuracy rate of the proposed model with and offline models is presented in Table. 4.

Table 4. Results Comparison with Already Proposed Models

Author/ Reference	No of Users	Dataset Used	Technique used	Error Rate	Accuracy
Online Signature Verification Models					
[28]	50	-	Dynamic Time Wrapping	0.35%	99.65%
Proposed Model	50	ATVS-SSig DB	Decision Tree (J48)	0.10%	99.90%
			Probability (NB tree)	0.22%	99.78%
			K-Nearest Neighbor	1.89%	98.11%
Offline Signature Verification Models					
[39]	5	MCYT	-	23.78%	-
	10		-	22.13%	-
[40]	5	MCYT-75	Gray Level Features	13.50%	-
[41]	5	MCYT-75	LBP	12.82%	-
	10			10.68%	-
	5		GLCM Features	16.27%	-
	10			12.65%	-
Proposed Model	5	MCYT-75	Pixel Intensity levels k-Nearest Neighbor	11.96%	88.04%
	10			11.80%	88.20%

7. Discussion

For testing purpose we, have also tried some other classifiers like MLP (many of the authors in literature have used MLP) but the training time of MLP was too high. Further, the size of neural network is another issue that restricts the neural network to few problems and ignore few most critical problems which has large data. The larger size of the neural network causes training difficult and time consuming as we tried here, but the Neural Network took almost hours to train even with a small number of signature data where the other classifies has solved the same classification within few seconds. With the increase in data the training of the Neural Network even gets worst. The second issue with neural networks is the geometry that totally depends upon the size of the training data [24].

Further, we also tested 10 signature samples against every user and the results were almost same that shows this model can perform better with larger datasets. The pruning has decreased the results but improved the computation time. The synthetic

(forgery) online signatures were not included in the system which may reduce the accuracy rate of the model.

The Naïve Bayes Tree faced a low memory heap problem with large dataset to avoid this problem 5 fold cross validation performed with larger datasets (50 users in online and 10 users with forgery in offline). It has performed well and remained second after J48, the k-Nearest Neighbor has the lowest accuracy rate among the three classifiers with online signatures. The k-Nearest Neighbor has shown better accuracy with offline signatures among the three classifiers.

Still there are a lot of gaps to be filled in the area of the offline signature, and this gap can be reduced by modifying the preprocessing and feature extraction steps. The performance of Naïve Bayes Tree and J 48 was not good because of the feature selection; by selecting best features the performance can be further enhanced. Because the same two approaches have shown better results with the online model but didn't perform well with the offline model.

8. Conclusion and Future Work

In this study, the online and offline signature verification model based on pixel intensity levels has been proposed. The main focus of paper is on offline signatures, but for the purpose of comparison of accuracy rates of online signatures with offline signatures, an online signatures database is also used. The experiments have been conducted with the signature of 50 users from an online signature database (5 signatures against each user). The proposed online model is given in Figure 1 (b), have shown more than 99.90% accuracy rate in the case of online signatures with Decision Tree (J48), the accuracy of 99.82% is obtained using probability based Naïve Bayes (NB tree) and 98.11% accuracy rate is achieved with distance based k-Nearest Neighbor classifiers. The detailed results are presented in the Table 2 of the experimental results section. The same experiments were carried out with offline model, in Figure 1 (a), with the MCYT-75 offline signature sub corpus with the same three classifiers used, with online signatures. Among the three classifiers k-Nearest Neighbor has shown more than 91.91% accuracy rate (without forgeries) and 88.12% accuracy rate (with forgeries) with offline signatures. The detailed results are presented in the Table 3, of the experimental results section. The experiments were carried out with the Weka 3.6 tool. The main reason for robustness of this model is relying on individual pixel intensity levels. The statistical classifiers have also played a major role in the higher accuracy rate of the model. In the future work, we intend to extend research by changing the preprocessing approaches and image segmentation for Decision Tree J48 and the Naïve Bayes Tree, in order to improve the accuracy rate to the level of k-Nearest Neighbor in offline signatures. Research will also focus on the signatures used in security systems [42].

References

[1] A. Pansare and S. Bhatia, "Handwritten Signature Verification Using Neural Network", International Journal of Applied Information Systems, vol. 1, no. 2, **(2012)**, pp. 44-49.
[2] A. S. Shah, M. N. A. Khan and A. Shah, "An Appraisal of Off-Line Signature Verification Techniques", International Journal of Modern Education and Computer Sciences, vol. 7, no. 4, **(2015)**, pp. 67-75.
[3] M. I. Malik, M. Liwicki and A. Dengel, "Evaluation of Local and Global Features for Offline Signature Verification", Proceedings of the 1st International Workshop on Automated Forensic Handwriting Analysis (AFHA), **(2011)**, pp. 26-30.
[4] D. Impedovo, G. Pirlo and R. Plamondon, "Handwritten Signature Verification: new Advancements and Open Issues", Int. Conf. Frontiers in Handwriting Recognition, **(2012)**, pp. 365–370.
[5] M. N. Ayaz, I. Javed and W. Mahmood, "Handwritten Character Using Multiclass SVM Classification with Hybrid Feature Extraction", Pak. J. Engg. Appl. Sci, vol. 10, **(2012)**, pp. 57–67.

[6] G. Agam and S. Suneel, "Warping-Based Offline Signature Recognition", IEEE Transactions on Information Forensics and Security, vol. 2, no. 3, **(2007)**, pp. 430-437.
[7] J. A. Mahar, M. K. Khan and M. H. Mahar, "Off-line Signature Verification of Bank Cheque Having Different Background Colors", IEEE/ACS International Conference on Computer Systems and Applications, **(2007)**, pp. 738-745.
[8] X. Wu, V. Kumar, J. R. Quinlan, J. Gosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, P. S. Yu, Z. H. Zhou, M. Steinbach, D. J. Hand and D. Steinberg, "Top 10 Algorithms in Data Mining", Knowledge and Information Systems, vol. 14, no 1, **(2008)**, pp. 1-37.
[9] F. Wahid, M. Fayaz and A. S. Shah, "An Evaluation of Automated Tumour Detection Techniques of Brain Magnetic Resonance Imaging (MRI)", International Journal of Bio-Science and Bio-Technology (IJBSBT), vol. 8, no. 2, **(2016)**, pp. 265-278.
[10] M. Mohri, A. Rostamizadeh and A. Talwalkar, "Foundations of Machine Learning", The MIT Press, **(2012)**.
[11] R. Sathya and A. Abraham, "Comparison of Supervised and Unsupervised Learning Algorithms for Pattern Classification", International Journal of Advanced Research in Artificial Intelligence (IJARAI), vol. 2, no. 2, **(2013)**, pp. 34-38.
[12] T. Keit, R. Palaniappan, P. Raveendran and F. Takeda, "Signature Verification System Using Pen Pressure for Internet and E-Commerce Application", Proceedings of ISSRE **(2001)**, Organized by Chillarge Inc, USA.
[13] J. R. Quinlan, "Induction of Decision Trees", Machine Learning, vol. 1, no. 1, **(1986)**, pp. 81-106.
[14] S. Odeh and M. Khalil, "Apply Multi-Layer Perceptron Neural Network for Off-Line Signature Verification and Recognition", International Journal of Computer Science Issues, vol. 8, no. 6, **(2011)**, pp. 261-266.
[15] M. Tomar and P. Singh, "A Directional Feature with Energy Based Offline Signature Verification Network", International Journal on Soft Computing, vol. 2, no. 1, **(2011)**, pp. 48-57.
[16] M. Parodi, J. C. Gomez and A. Belaid, "A Circular Grid-Based Rotation Invariant Feature Extraction Approach for Off-Line Signature Verification", 2011 International Conference on Document Analysis and Recognition (ICDAR), **(2011)**, pp. 1289-1293.
[17] M. B. Yilmaz, B. Yanikoglu, C. Tirkaz and A. Kholmatov, "Offline Signature Verification Using Classifier Combination of HOG and LBP Features", 2011 International Joint Conference on Biometrics (IJCB), **(2011)**, pp. 1-7.
[18] B. H. Shekar and R. K. Bharathi, "Eigen-Signature: a Robust and an Efficient Offline Signature Verification Algorithm", 2011 International Conference on Recent Trends in Information Technology (ICRTIT), **(2011)**, pp. 134-138.
[19] A. C. Ramachandra, J. S. Rao, K. B. Raja, K. R. Venugopla and L. M. Patnaik, "Robust Offline Signature Verification Based on Global Features", IEEE International Advance Computing Conference, **(2009)**, pp. 1173-1178.
[20] S. A. Daramola and T. S. Ibiyemi, "Offline Signature Recognition Using Hidden Markov Model (HMM)", International Journal of Computer Applications, vol. 10, no. 2, **(2010)**, pp. 17-22.
[21] M. A. Vargas, J. F. Vargas, A. Morales and A. Ordonez, "Robustness of Offline Signature Verification Based on Gray Level Features", IEEE Transactions on Information Forensics and Security, vol. 7, no. 3, **(2012)**, pp. 966-977.
[22] C. Kruthi and D. C. Shet, "Offline Signature Verification Using Support Vector Machine", 2014 Fifth International Conference on Signal and Image Processing (ICSIP), **(2014)**, pp. 3-8.
[23] J. S. Arunalatha, C. R. Prashanth, V. Tejaswi, K. Shaila, K. B. Raja, A. Dinesh, K. R. Venugopal, S. S. Iyengar and L. M. Patnaik, "PCVOS: Principal Component Variances Based Off-Line Signature Verification", In proc. of 2nd International Conference on Recent Trends in Information Systems (ReTIS), **(2015)**, pp. 195-199.
[24] H. Baltzakisa and N. Papamarkos, "A New Signature Verification Technique Based on a Two-Stage Neural Network Classifier", Engineering Applications of Artificial Intelligence, vol. 14, **(2001)**, pp. 95-103.
[25] Z. H. Quan and K. H. Liu, "Online Signature Verification Based on the Hybrid HMM/ANN Model", International Journal of Computer Science and Network Security (IJCSNS), vol. 7 no. 3, **(2007)**, pp. 313-322.
[26] J. F. Aguilar, L. Nanni, J. L. Pentildealba, J. O. Garcia and D. Maltoni, "An On-Line Signature Verification System Based on Fusion of Local and Global Information", Proc. 5th IAPR Intl. Conf. AVBPA, vol. 3546, **(2005)**, pp. 523 -532.
[27] A. K. Jain, F. D. Griess and S. D. Connell, "On-Line Signature Verification", Pattern Recognition, vol. 35, no. 12, **(2002)**, pp. 2963-2972.
[28] D. Putra, Y. Pratama, O. Sudana and A. Purnawan, "An Improved Dominant Point Feature for Online Signature Verification", International Journal of Security & Its Applications, vol. 8, no.1, **(2014)**, pp. 57-70.
[29] V. Abirami, S. Divya, S. Urmela, N. Poonguzhali and M. Ezhilarasan, "A Secured Rank Based Multi Biometrics System Using Enhance Blind Encryption Technique", International Journal of Research in Computer and Communication Technology, vol. 2, no. 7, **(2013)**, pp. 411-419.

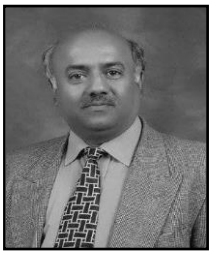
[30] J. Galbally, R. Palmondon, J. Fierrez and J. O. Garcia, "Synthetic On-Line Signature Generation, Part-I: Methodology and Algorithms", Pattern Recognition, vol. 45, no. 7, **(2012)**, pp. 2610-2621.
[31] J. Galbally, J. Fierrez, J. O. Garcia and R. Palmondon, "Synthetic On-Line Signature Generation, Part-II: Experimental Validation", Pattern Recognition, vol. 45, no. 7, **(2012)**, pp. 2622-2636.
[32] J. F. Aguilar, N. A. Hermira, G. M. Marquez and J. O. Garcia, "An off-Line Signature Verification System Based on Fusion of Local and Global Information", In Biometric Authentication, **(2004)**, pp. 295-306.
[33] J. O. Garcia, J. F. Aguilar, D. Simon, J. Gonzalez, M. F. Zanuy, V. Espinosa and A. Satue, "MCYT Baseline Corpus: a Bimodal Biometric Database", IEE Proceddings-Vision, Image and Signal Processing, vol. 150, no. 6, **(2003)**, pp. 395-401.
[34] Z. Xizhi "The Application of Wavelet Transform in Digital Image Processing", 2008 International Conference on Multi Media and Information Technology, **(2008)**, pp. 326-329.
[35] B. Ribeiro, I. Goncalves, S. Santos and A. Kovacec, "Deep Learning Networks for Off-Line Handwritten Signature Recognition", In Proceedings of the 16th Ibero american Congress conference on Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications, CIARP'11, **(2011)**, pp. 523–532.
[36] R. S. Stankovic and B. J. Falkowski, "The Haar Wavelet Transform: its Status and Achievements", Computer and Electrical Engineering, vol. 29, no. 1, **(2003)**, pp. 25-44.
[37] C. E. Shannon, "A Mathematical Theory of Communication", Reprinted with corrections from The Bell System Technical Journal, vol. 27, **(1948)**, pp. 379–423 and 623–656.
[38] M. Petrou and P. G. Sevilla, "Chapter 4. Non-Stationary Grey Texture Images", Image processing: Dealing with Texture Published Online, **(2006)** May.
[39] F. A. Fernandez, M. C. Fairhurst, J. Fierrez and J. O. Garcia, "Automatic Measures for Predicting Performance in Off-Line Signature", IEEE International Conference on, Image Processing, vol. 1, **(2007)**, pp. 369-372.
[40] J. F. Vargas, C. M. Travieso, J. B. Alonso and M. A. Ferrer, "Off-Line Signature Verification Based on Gray Level Information Using Wavelet Transform and Texture Features", In Proc. Int. Conf. Frontiers in Handwriting Recognition (ICFHR), **(2010)**, pp. 587-592.
[41] J. F. Vargas, M. A. Ferrer, C. M. Travieso and J. B. Alonso, "Off-Line Signature Verification Based on Grey Level Information Using Texture Features", Pattern Recognition, vol. 44, no. 2, **(2011)**, pp. 375-385.
[42] M. Shah and A. S. Shah, "Appraisal of the Most Prominent Attacks due to vulnerabilities in cloud computing," International Journal of Grid and Distributed Computing (IJGDC), vol. 9, no. 7, **(2016)**, pp. 13-22.

Authors

Abdul Salam Shah, he is currently doing specialization in Management Information System (MIS) from Virtual University of Pakistan. He has completed MS degree in Computer Science from SZABIST, Islamabad, Pakistan in 2016. He did his BS degree in Computer Science from Isra University Hyderabad, Sindh Pakistan in 2012. In addition to his degree, he has completed short courses and diploma certificates in Databases, Machine Learning, Artificial Intelligence, Cybercrime, Cybersecurity, Networking, and Software Engineering. He has published articles in various journals of high repute. He is a young professional and he started his career in the Ministry of Planning, Development and Reforms, Islamabad Pakistan. His research area includes Machine Learning, Artificial Intelligence, Digital Image Processing and Data Mining. Mr. Shah has contributed in a book titled "Research Methodologies; an Islamic perspectives," International Islamic University Malaysia, November, 2015.

Muhammad Fayaz, he is currently perusing Ph.D. in Computer Science from, JEJU National University, South Korea. Before joining the JEJU National University, he has also completed the course work of Ph.D from University of Malakand, Chakdara, KPK, Pakistan. He received MS in Computer Science from SZABIST, Islamabad,

Pakistan in 2014. He did MSC from the University of Malakand, KPK, Pakistan in 2011.



Asadullah Shah, he is working as Professor and Head of department of Information Systems (HOD) at the Kulliyyah of ICT, International Islamic University Malaysia (IIUM) before joining IIUM, he worked as Head of Telecommunication Engineering & Management department, IoBM Karachi Sindh, Dean Faculty of Computer and Management Sciences, Isra University Hyderabad Sindh and Head of Telecommunication Engineering and IT, Sukkur IBA, Sindh-Pakistan.

He did his Ph.D. from the university of Surrey UK, in 1998, with the specialization in Multimedia Communication. He started his academic carrier from University of Sindh Jamshoro, Pakistan in 1986 as a lecturer.

He has published 200 research articles in highly reputable international and national journal in the field of computers, communication and IT. Also, he has published 12 books in his 30 years of the academic carrier. Currently he is supervising great number of postgraduate students, working in multiple disciplines, specially, animation, social media and image processing in the Department of Information Systems, Kulliyyah of Information and Communication Technology, International Islamic University Malaysia.

Muhammad Naeem Ahmed Khan obtained D.Phil. degree in Computer System Engineering from the University of Sussex, England. His research interests are in the fields of software engineering, cloud computing, cyber administration, digital forensic analysis and machine learning techniques.

Fazli Subhan obtained his Ph.D. degree in Information Technology from University Teknologi PETRONAS, Malaysia in 2012. His research interest includes wireless communication, Bluetooth networks, pattern matching and machine learning.

