Online Handwritten Signature Recognition by Length Normalization using Up-Sampling and Down-Sampling

Fahad Layth Malallah
Computer Science,
Cihan University, Kurdistan/ Iraq.
Fahad.layth.86@gmail.com

Sharifah Mumtazah Syed Ahmad
Faculty of Engineering,
Universiti of Putra (UPM),
Malaysia.
s_mumtazah@upm.edu.my

Wan Azizun Wan Adnan
Faculty of Engineering,
Universiti of Putra (UPM),
Malaysia.
wawa@upm.edu.my

Olasimbo Ayodeji Arigbabu
Faculty of Engineering,
Universiti of Putra (UPM),
Malaysia.
oa.arigbabu@gmail.com

Vahab Iranmanesh
Faculty of Engineering,
Universiti of Putra (UPM),
Malaysia.
vahab.iranmanesh@gmail.com

Salman Yussof,
College of Information technology,
UniversitiTenagaNasional (UNITEN),
Malaysia.
Salman@uniten.edu.my

ABSTRACT

With the rapid advancement of capture devices like tablet or smart phone, there is a huge potential for online signature applications that are expected to occupy a large field of researches in forthcoming years. Online handwritten signature encounters difficulty in the verification process because an individual rarely produce exactly the same signature whenever he signs. This difference in the produced signature is referred to as intra-user variability. Verification difficulty occurs especially in the case where the feature extraction and classification algorithms are designed to classify a stable length vector of input features. In this paper, we introduce an efficient algorithm for online signature length normalization by using Up-Sampling and Down-Sampling techniques. Furthermore, online signature verification system is also proposed by using both Principal Component Analysis (PCA) for feature extraction and Artificial Neural Network (ANN) for classification. The SIGMA database, which has more than 6,000 genuine and 2,000 forged signature samples taken from 200 individuals, is used to evaluate the effectiveness of the proposed technique. Based on the tests performed, the proposed technique managed to achieve False Accept Rate (FAR) of 5.5% and False Reject Rate (FRR) of 8.75%.

KEYWORDS

Artificial neural network, authentication, biometrics, principal components analysis, length normalization and signature verification.

1 INTRODUCTION

Biometric system is a pattern-recognition system, which recognizes an individual based on a feature vector extracted from physiological or behavioral characteristic that belongs to the person [1,2]. Biometric is one of the emerging techniques used for authentication because it has been shown to be both more fool-proof and reliable [3]. There are two main modes of a biometric system [4]. The first mode is the Identification Mode, which means comparing the target biometric data with all the data available in the system, or simply one that can be translated into this question: “Who are you?”. In other words it performs a one-to-many *(1: N)* match. Usually, this mode consumes much time because it needs to do many comparison operations. The main purpose of user identification is to search the closest matching identity. This type of biometric authentication is normally used in surveillance and forensic applications [3, 5]. The second mode of biometric system is the Verification Mode, which is based on this question: “Are you who you claim to be?”. In this mode, the target biometric data is compared with the specific reference stored in the system to authenticate its identity. In other words, it performs a one-to-one *(1:1)* match. Usually, this mode needs less time than the identification mode [5, 6-8].

Handwritten signature normally consists of the first and last name of a person and it is also common that the signature does not contain the full name but only one part of it. This type of signature is referred to as a *paraph* [9]. Signature is a behavioral type of biometrics, which has a high legal value for document authentication, as well as being dependent on by both commercial transactions and government institutions [10, 11]. Furthermore, it acts as a non-invasive and non-intrusive authentication process for the majority of the users [7, 12]. It is one of the most accepted biometrics, since most individuals have their own signatures that could be used as their own token [13]. The property of the high intra-user variability undermines the absolute signature accuracy. This property takes place because individuals cannot produce a signature that is exactly the same as one of the previous versions. Another limitation is that handwritten signature can be forged without using specialized hardware [14]. For this reason skilled forged signatures should be considered in the testing. Signature authentication is done either based on static or dynamic data. The former is referred to as offline signature verification that performs user verification using scanned signature images, which are written on paper-based document. The latter is referred to as online signature verification system where signature samples are captured digitally usually using digitized pen and graphical tablets. Here, a richer amount of information is captured, which often includes signals as a time series of $x[t]$ and $y[t]$ coordinates, as well as pen pressure $p[t]$.However, it is difficult to achieve high correct matching accuracy in signature verification due to the high intra-user variability, which will increase the False Reject Rate (FRR).

The rest of the paper is organized as follows. Section 2 is dedicated for literature review related to signature verification. In section 3, the framework design is described. Then, the proposed signature length normalization is presented in section 4. In section 5, the signature verification system is proposed. The experiment and implementation is described in section 6 and finally, in section 7, the conclusion is presented.

2 SIGNATURE VERIFICATION

Most of the works of online handwritten signature verification involve the following four phases: data acquisition, pre-processing, feature extraction and classification [15]. Online signature samples are acquired by using standard graphical tablets or Personal Digital Assistant (PDA) to capture the signature data [16-18]. In preprocessing phase, some techniques that are adapted from signal processing algorithm are used [16]. The benefit of this phase is to enhance the input data in order to achieve a better performance. Some of the commonly used preprocessing techniques are filtering, noise reduction, smoothing, signature re-sampling (which could be achieved by using interpolation) [19], signature normalization in terms of position and scale [15], Fourier transform to standardize signatures in the domain of position, size, orientation, and time duration [20]. In the feature extraction phase, there are two types of features that can be used, which are function features and parameter features. In function features the signature is characterized as a time series signals, for example, horizontal signal *x[t]* and vertical signal *y[t]* for positions, velocity signal, acceleration signal, pen pressure signal, and pen inclination signal. In parameter features, the signature is characterized as a vector of elements that consists of a statistical and mathematical computation based on the acquired signature data. Examples would be the total signature time duration, pen down ration, number of pen up / pen down, AVE/ RMS/ MAX/ MIN of positions, speed, and acceleration [16]. In general, function features result in a better performance compared to parameter features, but they usually require time-consuming procedures for verification [21]. In the last phase, the classification process can be applied by using pattern recognition classification methods. Usually, signature verification could be implemented using statistical or template matching approaches [16]. In the case of template matching techniques, a queried sample is matched against templates of authentic / forgery signatures. In this case, the most common approach used is the Dynamic Time Warping (DTW) technique [7, 22, 23]. In the case of statistical approaches, distance-based classifiers can be used to perform signature verification. For example, Artificial Neural Networks (ANNs) are widely used for signature verification, due to their capabilities in learning and generalizing as shown in [24] and [25]. Other techniques that have been proposed by researchers to perform online signature verification are Hidden Markov Models (HMMs) [11], Support Vector Machine (SVM) [16], Bayesian decision method [26] and Fuzzy control [27].

3 FRAMEWORK DESIGN

An online signature sample consists of time series signals of horizontal $x[t]$ and vertical $y[t]$ coordinates, as well as pen pressure $p[t]$ sampled at time *t*. The samples were fed into our system for normalization and then verification. Figure 1 depicts the system stages as a flowchart. The first stage is normalization, which normalizes the length of the signature to a fixed or desired length. The second stage is classification which main task is to decide on the “Accept” / “Reject” status. Here, classification is performed using Principal Component Analysis (PCA) features and Artificial Neural Network (ANN) classifier. The operation of transformation and verification of online signature is started by reading the $x[t], y[t]$ and $p[t]$ signals, which are the horizontal trajectories, vertical trajectories and pen pressure respectively, where $t = 1,2....\overline{N}$, and $\overline{N}$ is the desired signature length. In the enrollment process, the signature features $x[t], y[t]$ and $p[t]$ are read into the system and then passed to the normalization process to output the time series of the online signature that has the length of $\overline{N}$.

Then the output of the normalized signature is passed to the feature extraction operation, which is implemented using PCA and then the output features are stored in the database as a reference model to be used in the prospective matching with anyone who wants to verify her / his signature. In the authentication process, the queried identity signature will be read by the system. The same processes that have taken place during the enrollment operation should also be applied to the queried signature sample.

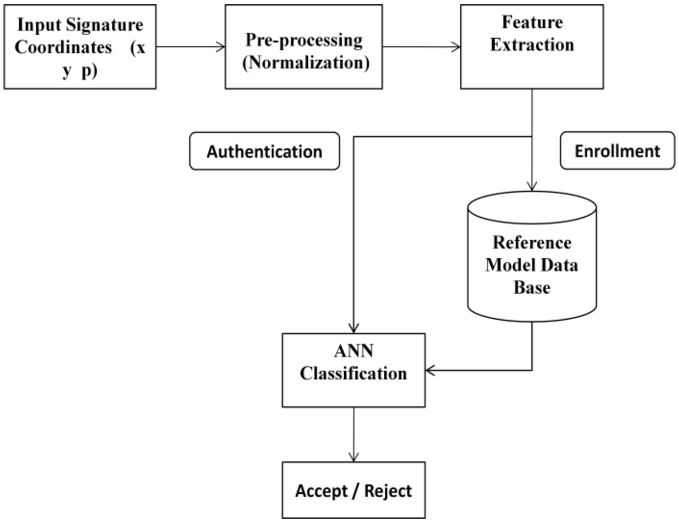
Input Signature Coordinates (x y p)
Pre-processing (Normalization)
Feature Extraction
Authentication
Enrollment
Reference Model Data Base
ANN Classification
Accept / Reject

Figure 1. Flowchart of the proposed biometric system.

In the classification process, ANN is used to compare the enrolled and authenticated signature features. Finally, decision maker based on threshold will decide whether the signature should be accepted or rejected.

4 SIGNATURE LENGTH NORMALIZATION

Intra-user variability of handwritten signature is inherited within genuine human signatures with regards to orientation size, and total signature duration (signal sampling parts). In this research work, normalization is performed in regard to time t such that all signature samples will have a fixed length of signing duration $\overline{N}$. In this work, the desired length of the normalization has been designed to be 256 signal sampling parts for all users in the database for a simulation as it is close to the average length (261 signal sampling part) of the SIGMA database for 200 users, as well as to assist the verification operation. Normalization is beneficial for verification operation, especially when the feature extraction and classifier algorithms are designed to take fix-length input signal.

The proposed normalization works by transforming an–unknown signature sample of variety length N into the desired signature length $\overline{N}$ without distorting the signature sample. The implementation of the normalization is based on Up-Sampling [28, 29] and Down-Sampling [29, 30]. Up-Sampling is defined as follows: the output $F[n]$ of a 2 factor up sampling is obtained by interlacing the input sequence $E[n]$ with zero value. To smooth the Up-Sampler operation, a specific value is inserted instead of zero interlacing, which is estimated by performing an average neighborhood interpolation as shown in (1) [31]:

$$value = \frac{E[n] + E[n+1])}{2} \qquad (1)$$

Down-Sampling is defined as follows: if $G[n]$ is the input of a 2 factor down sampling, and then the output is $H[n] = G[2n]$. Down-Sampling is required for those signatures that have signing duration N larger than the desired length $\overline{N}$, whereas Up-Sampling is required for those signatures that have signing duration N less than the desired length $\overline{N}$. Normally, Up-Sampler and Down-Sampler are applied to linear discrete-time systems. Figure 2 depicts the two re-sampling operations. The lower arrow shows an Up-Sampling operation where the number of sine waves is increased from 2 to 6 periods (up-sampling with a factor of 3). The upper arrow of the same figure shows a Down-Sampling operation where the number of sine waves is decreased from 6 to 2 periods (down-sampling with a factor of 3). In the current research, there is no stable factor due to the intra-user variability property of handwritten signature.

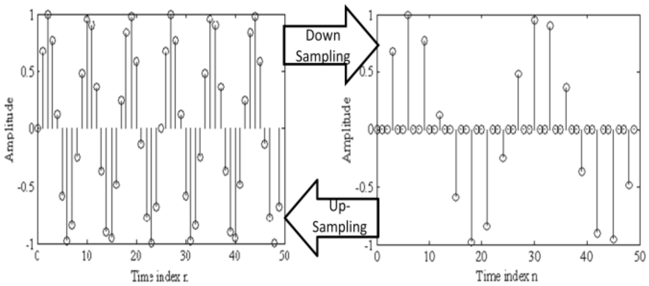

Figure 2.Up-Sampling and Down-Sampling illustration.

The online signature normalization process is depicted in Figure 3. The algorithm is started by comparing the target $\overline{N}$ (desired length) with the input signing duration N. If the input is less than $\overline{N}$, then Up-Sampling operation will be invoked to increase the total number of sampling parts in order to make the length equal to $\overline{N}$. Otherwise, Down-Sampling operation will be invoked to decrease the total sampling parts to make the length equal to $\overline{N}$.

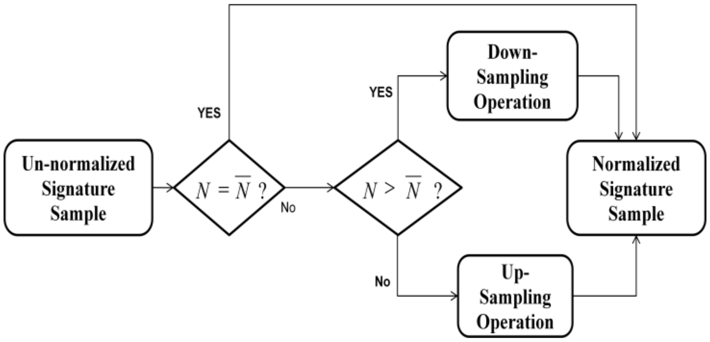

Figure 3.Online signature normalization process.

Up-Sampling process is implemented by using bi-linear interpolation [31] as shown in the following algorithm:

```
$\boldsymbol{Step1}: \boldsymbol{Target_{length} = \overline{N}}$;
$\boldsymbol{Step2}: \boldsymbol{x_{arry} = input\ original\ x\ signal(N)}$;
        $\boldsymbol{Step3}: \boldsymbol{while(x_{arry} < target_{length})}$
        $\boldsymbol{do}$
        $\boldsymbol{i = i + 1}$;
        $\boldsymbol{Interpolated_{value} = \frac{x_{arry}(i) + x_{arry}(i+1)}{2}}$;
    $\boldsymbol{insert\ (interpolated_{value})\ in(x_{arry})\ after\ (i)\ position}$;
                    $\boldsymbol{end}$
$\boldsymbol{Step4}: \boldsymbol{If x_{arry} reaches\ to\ end, repeat\ step3}$.
```

Then, **Down-Sampling process** is implemented by using decimation as shown in the following algorithm:

```
$\boldsymbol{Step1}: \boldsymbol{Target_{length} = \overline{N}}$;
        $\boldsymbol{Step2}: \boldsymbol{x_{arry} = input\ original\ x\ signal(N)}$;
            $\boldsymbol{Step3}: \boldsymbol{while(x_{arry} > target_{length})}$
                        $\boldsymbol{do}$
                        $\boldsymbol{i = i + 2}$;
        $\boldsymbol{delete\ \ value\ of\ (i)position\ \ from\ x_{arry}}$;
                        $\boldsymbol{end}$
$\boldsymbol{Step4}: \boldsymbol{If x_{arry} reaches\ to\ end, then\ repeat\ step3}$.
```

The same algorithm (Up-Sampling and Down-Sampling) is applied to $y[t]$ and $p[t]$ time series signals.

Results of the normalization stage are visualized using two display styles. For the first one, the signals x , y and p are visualized with respect to time *(t)*, and for the second one, the x signal versus y signal is shown in order to visualize whether there is a difference between un-normalized and normalized signatures. The reason for visualizing using the first style is to clarify what has been modified in the normalized signal compared to the original one. Furthermore, two types of normalized signature are shown. The first shows the result of a Down-Sampling operation of a signature that has more signal sampling parts (trajectories) than the required length of 256. The second shows the result of an Up-Sampling operation on a signature that has fewer signal sampling parts (trajectories) that the required length of 256.

In order to test the normalization process, a signature with the maximum signature length of 914 trajectories is chosen from the SIGMA database for the Down-Sampling test and a signature with the minimum signature length of 53 trajectories is chosen for the Up-Sampling test. The results of the Down-Sampling are depicted in Figure 4 and Figure 5. In Figure 4, signature signals x , y and p versus time t are depicted. The figure shows the signal length of 914 signal sampling parts (trajectories) before normalization, and after normalization, it is shown that the target length $\overline{N}$ is achieved while still maintaining the shape and position of the signature.

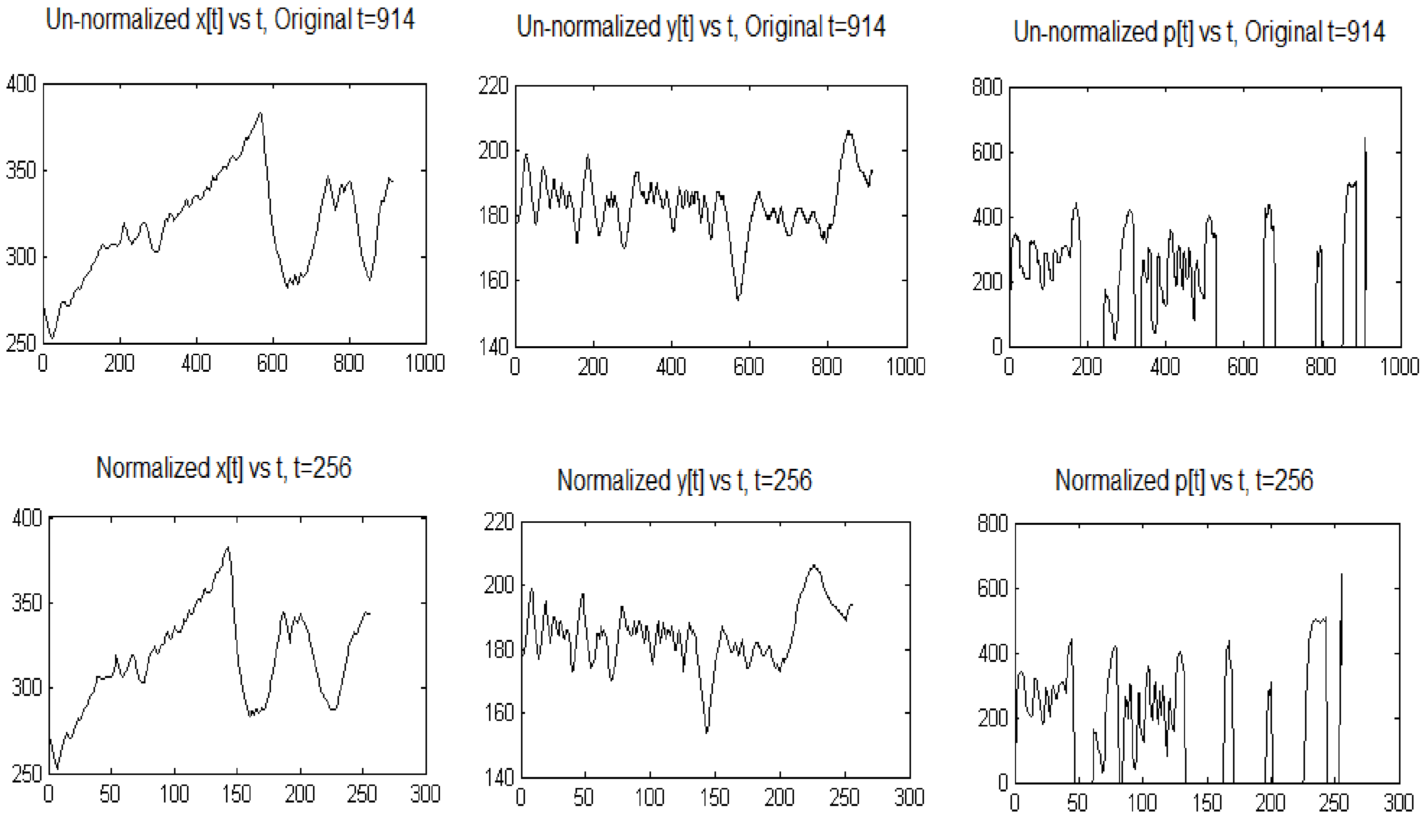

Figure 4. Down-Sampling normalized signals x , y and p diagram. The first row shows Un-normalized signals and the second row shows normalized signals.

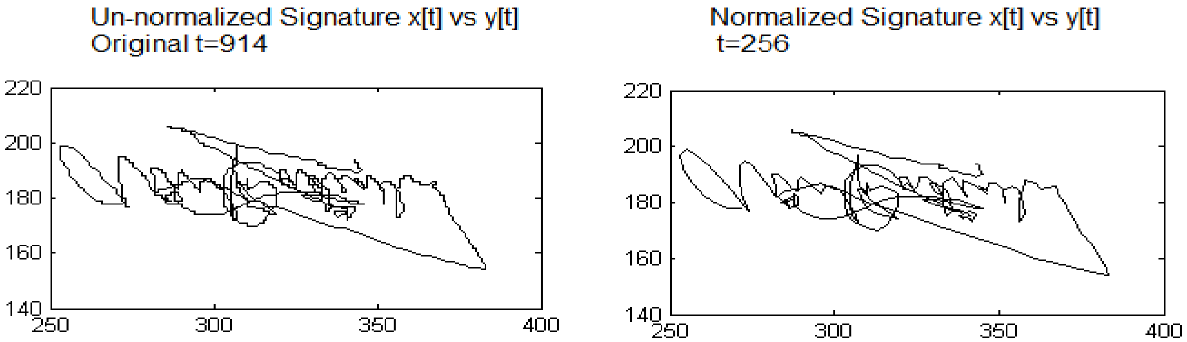

Figure 5. Signature before and after Down-Sampling normalization operation.

Figure 5 visualizes the signature before (left side) and after normalization operation (right side) as x signal versus y signal. It could be observed that, there is no significant difference between them. The results of Up-Sampling are shown in Figure 6 and Figure 7.

Specifically, in Figure 6 signature signals x, y and p versus time t are depicted. It is clear from the figure that each signal length before normalization was 53 trajectories, while after normalization the target length of 256 trajectories is achieved.

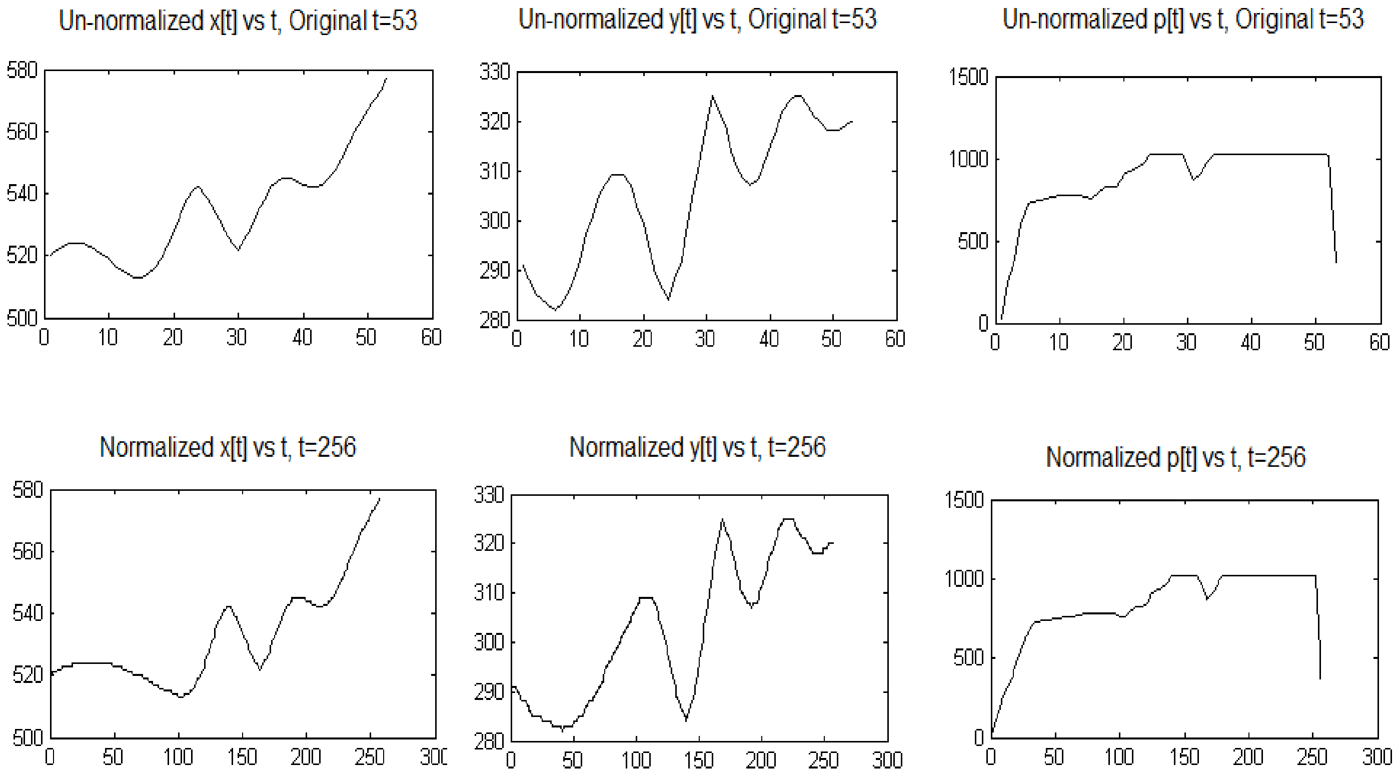

Figure 6. Up-Sampling normalized signals x, y and p diagram, the first row shows Un-normalized signals and the second row shows normalized signals.

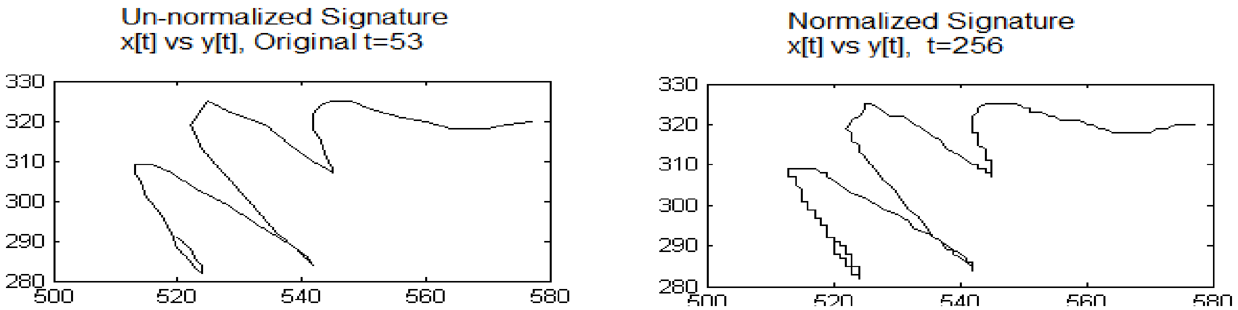

Figure 7. Signature before and after Up-Sampling normalization operation.

Figure 7 visualizes the signature before (left side) and after (right side) Up-Sampling normalization operation as x signal versus y signal from 53 to 256 signal sampling parts (trajectories).

From the tests performed, it is shown that the proposed normalization process has managed to achieve the intended objective to modify the length of a signature so that it has the desired length $\overline{N}$. The proposed normalization algorithm might be applicable to control the length of any biometric modal, where it is able to Up-Sample or Down-Sample a time series signal without any side effect for further processing operations.

5 SIGNATURE VERIFICATION SYSTEM

The signature verification system consists of two separate processes which are feature extraction and classification. Feature extraction is implemented using Principal Component Analysis (PCA), while classification is implemented using Artificial Neural Network (ANN).

5.1 Principal Component Analysis (PCA)

To increase the recognition rate of handwritten signature, it is preferred to use a feature extraction that transforms the signature signals from its original domain to another domain to maximize the variance and decrease the correlation between genuine and forged signature samples. Principal Component Analysis (PCA) is used in this research work to improve the recognition rate [32]. The reason for using PCA is due to its ability to transform a data set (signatures in our case) from correlated domain to another domain that is highly uncorrelated among the original data set [32,33]. Specifically, PCA can do variance maximizing between genuine and forged signature samples [32]. PCA is a method that uses orthogonal transformation to switch a set of points of possibly correlated variables into a set of points of uncorrelated variables namely principal components. More details relevant to PCA implementation can be found in [34]. In our work, PCA is implemented by transforming the normalized time series signals of $x[t]$, $y[t]$ and $p[t]$ into uncorrelated domain. After PCA implementation, three columns of outputs are produced. The first column corresponds to the first Eigen vector component that belongs to the highest Eigen value. The second column is the second highest Eigen value and the third column is the lowest Eigen value. The length of each column is 256 features, which is the length of the signature after normalization. Figure 8 shows the effect of PCA on our dataset of signatures. It is clear that before undergoing the PCA operation, it is difficult for the ANN to separate genuine from forged signatures and assign a group for each one because before PCA both genuine and formed signature samples are consisting of correlated features. Conversely, classification can be done more easily by ANN after the PCA operation. The three component vectors of PCA output are combined in a single vector to represent a signature sample. In order to avoid a long signature represented vector, which is unpractical for the classification, feature

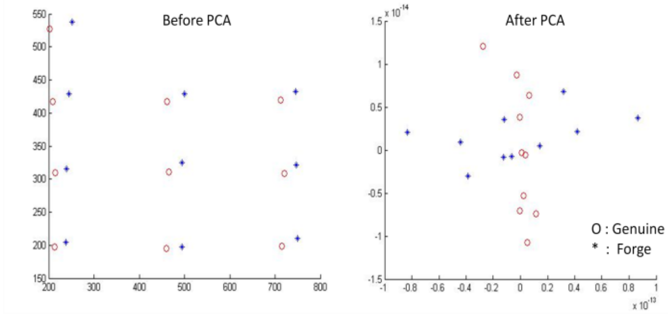

Figure 8. PCA is effective on generating variance maximization between genuine and forged signatures.

selection based on equal segment is employed to reduce the vector representation length. In this study feature selection is done empirically. The selection is implemented on each component vector by dividing the vector (256) into 8 segments (seg_xx); each segment size is 32 features. The selection is done by taking the 1st (seg_11), 4th (seg_14) and 8th (seg_18) segments among the 8 segments from the first component vector. The same goes for the second and third columns. The reason for this selecting is that the three segments are equivalent to the first, middle and last partitions of the signature, taking advantage of the proposed normalization, which maintains the shape of the original signature. Finally the length computation of each signature represented vector is performed as in equation (2):

$$32(\text{Seg. size}) \times 3\ (\text{No. of Seg.}) \times 3\ (\text{compnents}) + 9\ \text{Eigen vectors} + 3\ \text{Eigen values} = 300\ \text{features} \qquad (2)$$

Table 1 explains the detailed procedure of the represented vector construction of each signature sample. In this table, each bolded cell is considered to be included in the final signature represented vector. Eventually, the length of the final vector is 300 floating point numbers. The order of the features of the represented vector for each signature sample consists of the bolded values that are in the table. The order is: c11, c12, c13, c21, c22, c23, c31, c32, c33, v1, v2, v3, seg_11, seg_14, seg_18, seg_21, seg_24,seg_28, seg_31, seg_34, seg_38.

Where each of cxx and vx is a floating point number, while seg_xx is 32 floating point numbers.

Table 1
Represented vector of each signature sample consisting of the underlined features.

Eigen-Vector	Eigen-Value	Three Component Vectors (seg_(row, column))							
c11 c12 c13	**Highest-v1**	**seg_11**	seg_12	seg_13	**seg_14**	seg_15	seg_16	seg_17	**seg_18**
c21 c22 c23	**Middle-v2**	**seg_21**	seg_22	seg_23	**seg_24**	seg_25	seg_26	seg_27	**seg_28**
c31 c32 c33	**Lowest-v3**	**seg_31**	seg_32	seg_33	**seg_34**	seg_35	seg_36	seg_37	**seg_38**

(i.e.) c21: first value of the second Eigen vector, v1: first Eigen value, seg_38: 8th segment of the 3rd component vector.

5.2 Artificial Neural Network (ANN)

Multi-layer perceptron (MLP) artificial neural network is employed in our research. MLP is a feed-forward artificial neural network model that maps sets of input data into a set of target outputs [35, 36]. MLP consists of multiple layers (input layer, hidden layer(s) and output layer). Each layer contains several nodes. Each node is stimulated according to an activation function. Every node is connected to subsequent nodes in the next layer (full connection), with no connection among nodes that are in the same layer. The training (learning) type of MLP is supervised learning technique called back-propagation of the training network [37]. The training term is achieved by modifying the weights by using certain number of iterations (Epochs) until the modified values satisfy the gradient error.

In the network construction, the number of nodes in the input layer is the same as the number of input features of the represented vector. In our case the represented signature features has 300 real number features (as discussed in PCA). Therefore, our input layer consists of 300 nodes. The number of output nodes is the number that identifies the general category of the state of the system [38]. The proposed system has one output node because it is able to identify whether the signature is a genuine or forged sample. The MLP-ANN parameters used in our algorithm are as follows: 300 input nodes in the input layer, 2 hidden layers, the first and the second hidden layers consist of 80 and 40 nodes respectively, and the final output layer is one node. The training algorithm used is the Scaled Conjugate Gradient (SCG) algorithm [39]. It has the ability to train 300 features in one vector efficiently. Activation function is tangent sigmoid (to activate -1 and +1 threshold output). A score result that is between 0 and -1 is classified as forged signature. While a score result between 0 and +1 is considered as genuine signature. The number of training iteration is 150, which is set experimentally. Learning rate is 0.3. An interleave division method [35] is used as a validation in order to improve generalization. This procedure cycles trained samples among the training set, validation set, and test set according to percentages. The percentage of division is 70% training set, 15% validation set and 15%—testing set. And the maximum error number of validation that stops the training is set to 6 (as default). All the aforementioned parameters of ANN have been chosen empirically after intensive experiments using MATLAB tool. Figure 9 illustrates the finalized ANN topology.

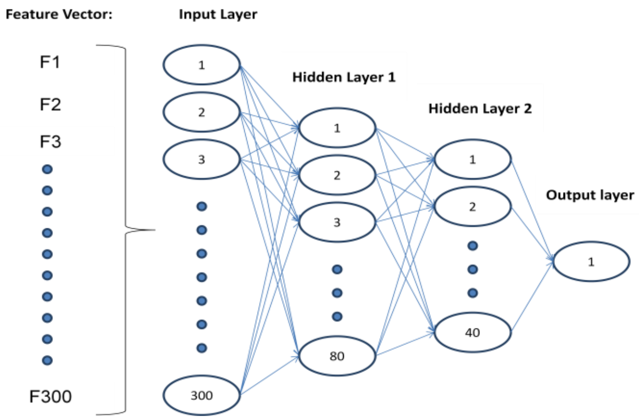

Figure 9. Artificial neural network structure for the proposed recognition system.

6 EXPERIMENT AND IMPLEMENTATION RESULT

The experiment is implemented on the online signature template after normalization preprocessing to measure the verification accuracy of the proposed normalization method. The experiment is done according to the following steps:

1- Using signatures from the SIGMA database [40], the training matrix is built. The training matrix consists of signatures from 200 individuals where for each individual, 10 genuine samples and 10 forged samples (five of them are random forged samples and the other five are skilled forged samples) are obtained. Each signature sample is represented by 300 features. Thus, the training matrix size is $[300 \times 20]$ (300 features for each sample with 20 samples for each individual). Training is run by ANN for the signatures of each individual separately.

2- The evaluation of the result produced by ANN is done by extracting the False Accept Rate (FAR) and the False Reject Rate (FRR) for each individual separately. The testing matrix is built similar to the way the training matrix was built.

3- In the training target (destination) of ANN, a sign $+1$ is assigned to the first 10 signature samples of the trained matrix, while -1 is assigned to the second 10 signature samples of the training matrix to mark and train the ANN that the first 10 are genuine samples and the second 10 are forged samples.

4- Compute the Receiver Operating Characteristics (ROC) curve of the verification. The threshold is varied from $+1$ till -1 with 0.1 intervals $(+1:0.1:-1)$.

5- FRR is computed by evaluating the resulting scores of the first 10 samples. If any sign of the first 10 samples is less than the threshold, False Rejection (FR) counter will be increased by one $(FR = FR + 1)$, since they are supposed to be as accepted (signs are larger than threshold) but they are wrongly rejected by the verifying system. On the other hand, if the results of the second 10 samples have signs more than the threshold, they are considered as False Accept (FA) and the counter will be incremented by one $(FA = FA + 1)$. The FAR and FRR are computed as in (3) and (4) respectively:

$$FAR = \frac{FA}{10} \times 100\% \qquad (3)$$

$$FRR = \frac{FR}{10} \times 100\% \qquad (4)$$

6- The accuracy of each user is computed by using (5):

$$User_{Accuracy}\% = 100 - \frac{FAR+FAR}{2} \qquad (5)$$

7- Then, to take into consideration all individuals in the SIGMA database, an average of the 200 individuals' accuracy is computed by using (6):

$$\mathrm{AVR}_{\mathrm{Accuracy}} = \frac{1}{200}\sum_{\mathrm{u}=1}^{200} \mathrm{User}_{\mathrm{Accuracy}}[\mathrm{u}] \qquad (6)$$

Concerning the result of the experiment, Table 2 lists FAR, FRR and their average. The error rates are extracted by using two experiments. The first one includes ANN validation and the second experiment excludes validation. The table lists the errors in terms of certain thresholds. For example, in the case of -0.2 threshold, the average error rate including validation is 10.525%, whereas without validation, the average error rate is 7.125%.

Table 2
Verification accuracies as an error rate (FAR and FRR) in terms of several thresholds.

Threshold	Validation		Average Error%	Without Validation		Average Error%
	FRR %	FAR %		FRR %	FAR %	
-0.5	10.1	11.9	11	8.2	6.85	7.525
-0.4	11.4	14.45	12.925	7.85	6.3	7.075
-0.3	10.05	11.6	10.825	9.15	6.2	7.675
-0.2	12.65	9.35	11	8.7	5.7	7.2
-0.1	**11.5**	**9.55**	**10.525**	**8.75**	**5.5**	**7.125**
0	13.85	10.5	12.175	9.1	6.4	7.75
0.1	14	8.55	11.275	10.85	4.75	7.8
0.2	14.6	6.8	10.7	10.55	4.7	7.625
0.3	16.25	7.85	12.05	11.55	4.85	8.2
0.4	16.35	5.85	11.1	11.65	4.45	8.05
0.5	21.2	5.3	13.25	11.75	4.3	8.025

Figure 10 depicts the ROC curves after the smoothing operation, where the threshold is between -1 and +1, and shows the error rate difference between verification with and without validation.

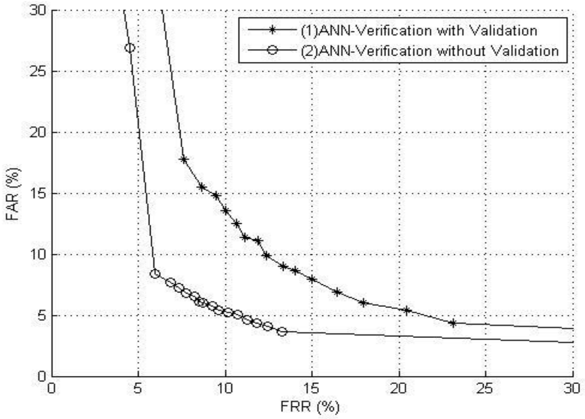

Fig. 10. ROC curves for verification accuracy with and without validation.

6 CONCLUSION

Online handwritten signature verification is implemented using function features, which are time series signals of the signature. These signals vary in the length according to the original signature samples given by the same signer. The proposed normalization, which comprises of Up and Down sampling, is implemented for generating a fixed or desired signal length for both horizontal (*x[t]*) and vertical (*y[t]*) signature position signals, as well as pen pressure signal (*p[t]*). In the result, the proposed normalization has eliminated the drawback of the intra-user variability by providing a fix length of signature samples without any side effects. Then, the normalized signature samples are passed to the proposed signature verification system, which consists of PCA and ANN. Finally, two types of experiments have been performed with and without validation. The average of the error rate accuracy achieved without validation is less than that with validation by 3.4%, because validation procedure tries to generalize the training data in ANN, which results in a more competitive verification rate.

7 ACKNOWLEDGE

This work was supported by Ministry of Higher Education of Malaysia which was made possible through the grant of Exploratory Research Grant Scheme (ERGS).

7 REFERENCES

1. L. O’Gorman, “Comparing passwords, tokens, and biometrics for user authentication,” in *Proc*. IEEE, Vol. 91, No. 12, Dec. 2003, pp. 2021–2040.
2. N. K. Ratha, J. H. Connell, and R. M. Bolle, “Enhancing security and privacy in biometrics-based authentication systems,” IBM Syst. J., vol. 40, no. 3, pp. 614–634, 2001.
3. S.M.S. Ahmad, B. M. Ali and W.A.W. Adnan ,“Technical issues and challenges of biometric applications as access control tools of information security,” international journal of innovative computing, information and control, vol. 8, vo. 11, pp.7983-7999 Nov. 2012.
4. S. G. Kanade, D. Petrovska-Delacr´ etaz, and B. Dorizzi, “Cancelable biometrics for better security and privacy in biometric systems,” Springer-Verlag Berlin Heidelberg, Part III, CCIS 192, pp. 20–34, 2011.
5. S. Prabhakar, S. Pankanti A.N. Jain, “Biometric recognition security and privacy concern,” IEEE security & privacy, 1540-7993/03/$17.00, pp 33-42. 2003.
6. A. K. Jain, K. Nandakumaand A. Nagar, “Biometric template security,” EURASIP J. Adv. Signal Process,” vol. 2008, no. 1, Article ID 579416, pp. 1-17, 2008.
7. E. Maiorana, P. Campisi, J. Fierrez, J. Ortega-Garcia and A.Neri, “Cancelable templates for sequence-based biometrics with application to on-line signature Recognition,” IEEE Transaction on system, man and cybernetics-part A: system and human, vol. 40, no.3, pp. 525–538, May. 2010.
8. K. R1. Radhika and S.V2. Sheela, “Fundamentals of biometrics—hand written signature and iris,” P. S. P. Wang (ed.), Pattern Recognition, Machine Intelligence and Biometrics, Higher Education Press, Beijing and Springer-Verlag Berlin Heidelberg, 2011.
9. B. Miroslav, K. Petra and F. Tomislav, ”Basic on-line handwritten signature features for personal biometric authentication,” in *Proc*. MIPRO 23-27, Opatija, Croatia, 2011, pp.1458–1463.
10. M. Faundez-Zanuy, “Signature recognition state-of-the-art,” IEEE Aerospace and Electronic Systems Magazine , vol. 20, no.7, pp. 28–32, Jul. 2005.
11. M. R. Freire , “Biometric template protection in dynamic signature verification,” M. S. thesis, Universidad Antonio de Nebrija, Madrid, Spain, Nov. 2008.
12. J. Fierrez-Aguilar, J. Ortega-Garcia, D. Ramos and J. Gonzalez-Ro driguez, “HMM-based on-line signature verification: Feature extraction and signature modeling,” Pattern Recognit. Lett., vol. 28, no. 16, pp. 2325–2334, Dec. 2007.
13. A. K. Jain and A. Kumar, “Biometrics of next generation: an overview,” to Appear in Second Generation Biometrics Springer, 2010.
14. S. Elliott, A. Hunt, “Dynamic signature forgery and signature strength perception assessment,” IEEE Aerospace and Electronic Systems Magazine, vol. 23, no.6, pp. 13 – 18, Jun. 2008.
15. Z. Zhang, K. Wang and Y. Wang ,“ A Survey of On-line Signature Verification,” CCBR, LNCS , vol. 7098, pp. 141–149, 2011.

16. D. Impedovo and G. Pirlo ," Automatic signature verification: The state of the art," IEEE transactions on systems, man , and cybernetics-part c: application and reviews, vol.38, no.5, pp.609-635, Sept.2008.
17. A. Kholmatov and B. Yanikoglu, "Biometric authentication using onlinesignatures," in Proc. ISCIS, vol. 3280, LNCS, C. Aykanat, T. Dayar, andKörpeolu, Eds., Berlin, Germany, 2004, pp. 373–380.
18. D. Muramatsu, M. Kondo, M. Sasaki, S. Tachibana, and T. Matsumoto,"A Markov chain Monte Carlo algorithm for Bayesian dynamic signatureverification," IEEE Trans. Inf. Forensics Security, vol. 1, no. 1, pp. 22–34, 2006.
19. A.K Jain, F.D.Griess, S.D.Connell, "On-line signature verification,"*PatternRecognition*, vol.35, no.12, 2963–2972,2002.
20. A. I. Al-Shoshan, "Handwritten signature verification using image invariantand dynamic features," *in Proc*. Int. Conf. Comput. Graphics,Imag. Vis. (CGIV 2006), 2008, pp. 173–176.
21. R. Plamondon and G. Lorette, "Automatic signature verification andwriter identification—The state of the art," Pattern Recognit., vol. 22,no. 2, pp. 107–131, Jan. 1989.
22. H. Bunke, J. Csirik, Z. Gingl and E. Griechisch ,"Online signature verification method based on the acceleration signals of handwriting samples," CIARP , LNCS 7042, pp. 499–506, 2011.
23. A. G. Reza, H. Lim and M. J. Alam," An efficient online signature verification scheme using dynamic programming of string matching," ICHIT 2011, LNCS 6935, pp. 590–597, 2011.
24. M.Khalid, H. Mokayed, R. Yusof and O. Ono ," Online signature verification with neural networks classifier and fuzzy inference ," Third Asia International Conference on Modelling& Simulation. 2009, pp.236-241.
25. M. J. Alhaddad, D. Mohamad, A. Ahsan ,"Online signature verification using probablistic modeling and neural network", *In Proc*. IEEE, May 2012,pp.1 – 5
26. A. Kholmatov, B. Yanikoglu, "Identity authentication using improved online signature verification method," Pattern Recognition Letters, pp. 2400–2408,2005.
27. J. Shin and T. Kikuchi, "On-line signature evaluation using fuzzy set theory," in *Proc.*Advanced Information Networking and Applications Workshops (WAINA), Barcelona, March 2013, pp. 273 – 277.
28. Y. Itoh and T. Ono, "Up-sampling of YCbCr4:2:0 Image exploiting inter-color correlation in RGB domain," IEEE Transactions on consumer electronics, vol. 55, no. 4, pp.2204-2210, Nov. 2009.
29. R. A. Gopinath and C. S. Burms ,"On upsampling, downsampling, and rational sampling rate filter banks," IEEE Trans Signal Process, vol. 42, no. 4, pp. 812-824, Apr. 1994.
30. L. Fang, O. C. Au. and A. K. Katsaggelos, " Adaptive joint demosaicing and subpixel-based down-sampling for bayer image," in *Proc.* IEEE,ICME, Barcelona, Spain, July, 2011, pp. 1 – 6.
31. B. Maguire, A. Miller , G. Gienko, "Elements of geographic information system," Training material (GII-01) ,Vilnius gediminas technical university, Lithuania , 2008, pp 143-144.
32. C. M. Bishop, " Pattern recognition and machine learning," in *Information Science and Statistics*, Springer, ISBN-10: 0-387-31073-8, 2006.pp.225-233.
33. J. Mohamad-Saleh and B. S. Hoyle ,"Improved neural network performance using principal component analysis on Matlab," International journal of the computer, the internet and Management vol.16, no.2, pp.1-8, May, 2008.
34. Jolliffe I.T. ,"Principal component analysis,"2nd ed. Series: Springer Series in Statistics, NY, 487 pp. 28 illus. ISBN 978-0-387-95442-4, 2002.
35. H. Demuth, M. Beale and M. Hagan ," Neural Network Toolbox™ 6 User's Guide," by The MathWorks, Inc., 2009.
36. H. Krishna, J. An and L. Zheng ,"A Neural network approach to classify inversion regions of high mobility ultralong channel single walled carbon nanotube field-effect transistors for sensing applications," in *Proc.* IEEE 5th International nanoelectronics conference (INEC), 2013, pp.85-88.
37. C. Pratola, F. D. Frate, G. Schiavon and D. Solimini ,"Toward fully automatic detection of changes in suburban areas from VHR SAR images by combining multiple neural-network models," IEEE Transaction on geosciences and remote sensing, vol.51, no. 4, pp.2055-2066, April, 2013.
38. M. R. G. Meireles, P. E. M. Almeida and M. G. Simõe ," A comprehensive review for industrial applicability of artificial neural networks," IEEE Trans Ind Electron, vol. 50, no. 3, pp.585-601, Jun. 2003.
39. M. F. Moller , "A scaled conjugate gradient algorithm for fast supervised learning," Neural Networks, vol.6, no.4, pp.525-533, 1993.
40. S.M. Syed Ahmad, A. Shakil, A.R. Ahmad, M.A. M. Balbed and R. Md. Anwar, " SIGMA – A Malaysian signature's database," in *Proc.* AICCSA 2008, IEEE/ACS International conference on computer systems and applications, Doha, Qatar, 31 March – 4 Mar.2008, pp.919-920.