# Signature Recognition and Verification using Hybrid Features and Clustered Artificial Neural Network(ANN)s

Manasjyoti Bhuyan,  Kandarpa Kumar Sarma and Hirendra Das,

*Abstract*—Signature represents an individual characteristic of a person which can be used for his / her validation. For such application proper modeling is essential. Here we propose an offline signature recognition and verification scheme which is based on extraction of several features including one hybrid set from the input signature and compare them with the already trained forms. Feature points are classified using statistical parameters like mean and variance. The scanned signature is normalized in slant using a very simple algorithm with an intention to make the system robust which is found to be very helpful. The slant correction is further aided by the use of an Artificial Neural Network (ANN). The suggested scheme discriminates between originals and forged signatures from simple and random forgeries. The primary objective is to reduce the two crucial parameters-False Acceptance Rate (FAR) and False Rejection Rate (FRR) with lesser training time with an intension to make the system dynamic using a cluster of ANNs forming a multiple classifier system.

*Keywords*—offline, algorithm, FAR, FRR, ANN.

## I. INTRODUCTION

Signature has been a distinguishing feature for person identification through ages. An increasing number of transactions, especially financial, are being authorized via signatures; hence methods of automatic signature recognition and verification is essential if authenticity is to be verified regularly [1]. Approaches to signature verification fall into two categories according to the acquisition of the data: On-line and Off-line. On-line data records the motion of the stylus while the signature is produced, and includes location, and possibly velocity, acceleration and pen pressure, as functions of time. Online systems use this information captured during acquisition [2]. These dynamic characteristics are specific to each individual and sufficiently stable as well as repetitive. Off-line data is a 2-D image of the signature. Signatures are composed of special characters and lines and therefore most of the time they can be unreadable. Also intrapersonal variations and interpersonal differences make it necessary to analyze them as complete images and not as letters and words put together. Offline processing is complex as there is an absence of stable dynamic characteristics. Difficulty also is related to the fact that it is hard to segment signature strokes due to highly stylish and unconventional writing styles. Other factors include non-repetitive nature of variation of the signatures, variation due to age, illness, geographic location and perhaps to some

Manasjyoti Bhuyan, Kandarpa Kumar Sarma and Hirendra Das are with the Department of Electronics and Communication Technology, Gauhati University, Guwahati - 781014, Assam, India. e-mail: (manasjyoti.b@gmail.com, kandarpaks@gmail.com and hirendra.das100@gmail.com).

extent the emotional state of the person. This complicates the problem further. All these factors together provide large intra-personal variations and make system design for signature verification to be a tedious task. The system should neither be too sensitive nor too coarse. It should have an acceptable trade-of between a low False Acceptance Rate (FAR) and a low False Rejection Rate (FRR) [5]. We approach the problem in two steps. Initially the scanned signature image is preprocessed to be suitable for extracting features. The slanting angle of the signature with horizontal line is calculated and rotation of the signature is adjusted. Then the preprocessed image is used to extract relevant geometric parameters that can distinguish signatures of different persons. Again we are using each preprocessed and rotation normalized signatures as a whole at a time after size normalization, as it provides useful information. Finally, results generated by the ANN classifiers are compared. Some of the relevant works are as in [1] to [8].

## II. EXPERIMENTAL DETAILS

Figure 1 illustrates a general offline signature verification system. Here an Artificial Neural Network (ANN) is trained with different feature sets extracted from the signature. The processed signature as a whole is also included in the training set as the signature can not be segmented for characters. Preprocessing of the raw scanned signature is the most important part of work for efficient recognition. We have done rotation normalization of the input signature such that signatures for each individual make same inclination because the orientation of the signature on the paper depends on the orientation of the paper at the time of signing (Fig-1).

### A. Signature Database:

We collected seven signatures from each individual on A4 size paper which is divided into eighteen blocks each of width 5.5cm and height 3.5cm. Then the paper is scanned using a HP flatbed scanner setting the resolution at 150 dpi. Both the signatures used in training and testing should be scanned at the same resolution. Samples from ten individuals were collected which gives us a database of seventy signatures. Five out of the seven were used for the training of the ANNs and the rest two were used for testing. After completion of training again few signatures were collected from individuals for testing which were not involved in the training set. The samples include signature of Assamese, English and Hindi. These three language specific signatures are taken to show that the system is language - independent.
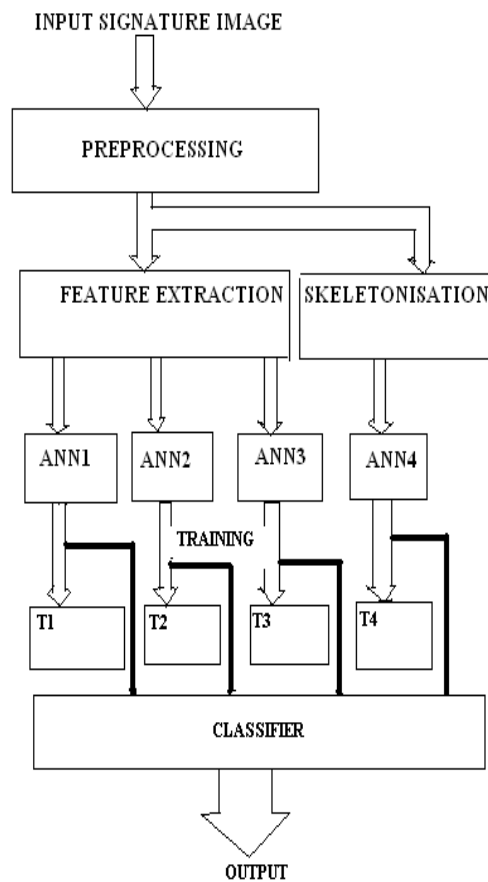
World Academy of Science, Engineering and Technology
International Journal of Electronics and Communication Engineering
Vol:4, No:8, 2010

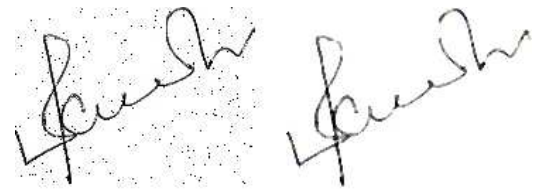Fig. 1. Verification process of signature

Fig. 2. Noisy image and the Filtered image

different types of thinning algorithm available including iterative algorithms, parallel algorithms, and sequential algorithms. Though they all take different approaches, their outcomes are the same- reduction of the character features to aid in feature extraction and classification. The goal of thinning is to eliminate the thickness differences of pen by making the image one pixel thick (Fig-3).

Fig. 3. Input image and the thinned image

## B. Preprocessing:

The preprocessing step is applied both in training and testing phases. Signatures are scanned in gray. The purpose in this phase is to make signatures standard and ready for feature extraction. The preprocessing stage includes seven steps:

*1) Noise reduction::* The scanned signature image contains salt and pepper noise due to roughness of the paper surface. Again we have added salt and paper noise to the image before preprocessing to make the training immune to noisy images. Median filtering is used to remove the noise(Fig-2).

*2) Binarisation::* The filtered image is converted to binary image with a threshold of 0.8 . The threshold is set to this value because after filtering the intensity of the signature pixels are also get reduced.

*3) Clutter Removal::* The converted binary image contains black dots of pixels which are not connected to the signature pixels. The signature is scanned with a 5 by 5 matrix to remove these unconnected pixels.

*4) Thinning::* This reduces the lines and arcs that represent the character down to a width of one pixel. There are several

*5) Rotation correction::* Rotation correction is an important step of our work. At the first step the geometrical center of the signature is calculated. It is done by scanning the signature row wise and column wise. Then the signature is divided into two parts by its geometrical center. Number of pixels of the signature in each half is calculated and each half is scanned row wise and column wise to get the geometrical center for each half. The slope of the line connecting the two co-centers with the X-axis of the image is calculated. Then the signature image is rotated with an angle such that the slope becomes zero (Fig-4). Bicubic interpolation is used to smooth the image after rotation. The algorithm is explained as below:
Suppose (x,y) and (a,b) be the coordinates of the two co centers of the signature. Then the slope of the signature can be calculated as

$$slope = arctan(x - a)/(y - b) \qquad (1)$$

The rotation detection and correction is also carried out using an ANN based approach (Fig-5). The signature rotation detection and correction system consists of the following blocks-

World Academy of Science, Engineering and Technology
International Journal of Electronics and Communication Engineering
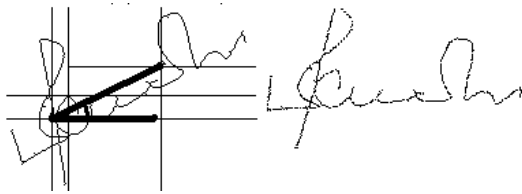Vol:4, No:8, 2010

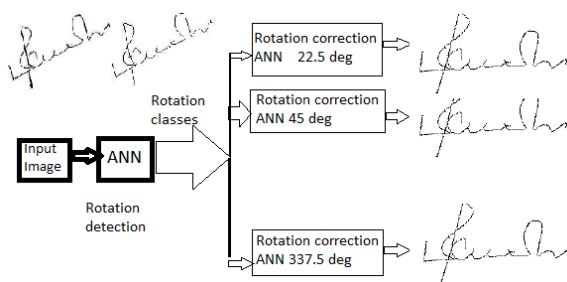Fig. 4.   image before and after rotation



Fig. 5.   Image rotation correction using ANN

- **Input:** Samples to the system is provided by the input block. A preprocessed signature image is given as the input to the system.
- **Rotation:** The input image is then rotated for use as test samples. The input images are rotated the entire cycle with gaps of $22.5^o$ which gives a total of 16 orientations.
- **Feature extraction:** The rotated images are next passed through the feature extraction system which is a Principal Component Analysis (PCA) feature block that provides the best set of relevant details.
- **Rotation Classifier:** This is a Generalized Feed Forward Artificial Neural Network (GFFANN) block designed to detect rotation angle of an input image. The PCA feature of each rotated image is applied to the GFFANN with an associated class code. The GFFANN is subjected to extensive training using (error) back propagation algorithm. At the end of the training the GFFANN is able to correctly recognize the angle of rotation of an image and generates the response by producing a class code appropriate for the specific rotation.
- **Correction:** This provides the correction algorithm applied to the rotated image to recover the original image. The correction block is constituted by a set of 16 GF-FANNs each trained to correct the rotated image. Figure 5 shows the correction setup where each ANN block handles the specific rotation correction depending upon the decision generated by the preceding rotation detection ANN. Image rotation is corrected by using the relation

given by eq. 2, called the three pass shear rotation. It is a separable image rotation technique which decomposes the rotation into operations which only involve rows or columns of the image [6]. A three step decomposition is used in this approach which is quiet tedious hence requires a method which is faster. A ANN based method can be a solution which is attempted in this work as summarized by the block diagram shown in Figure 5.

$$R(\theta) = \begin{pmatrix} cos(\theta) & -sin(\theta) \\ sin(\theta) & cos(\theta) \end{pmatrix} \qquad (2)$$

Each GFFANN is trained to implementation the image rotation correction using eq. 2 for specific rotation detected by the previous ANN block. During training each GFFANN is configured to handle the image rotation correction. For a decided angle, the GFFANN receives the rotation corrected image as its reference for training. The reference image is generated using the eq. 2 and the training carried out. At the end of the training, the rotation corrected image is generated.

The image rotation detection is performed by generating a set of outputs by the GFFANNs as below (Figure 5):

$$y_{1p} = f[\sum_k \sum_j f\{\sum \sum x[i,j] \times w[i,j] + b_{ij}\} \times w[j,k]] \quad (3)$$

where f(.) is the activation function associated with the ANNs, w[.] are the inter-layer connectionist weights, b[.] are the bias values and p is the number of rotation decision states. For each of the p rotation states image correction is carried out using eq. 2 which is implemented using 16 ANNs one each for the detected states. The output of these ANN blocks are governed by the following output expression:

$$y_{2p} = f[\sum_{jk} f\{\sum_i \sum_j x[i,j] \times w[i,j] + b[i,j] \times w[j,k]\}] \quad (4)$$

where b[.] are bias values required for the GFFANN training.

*6) Centering the Signature:* The signature image is scanned from each side to get the coordinates of the edge pixels. The rectangular area covering only the signature is cropped and a new image is obtained in which the sides of the image is touching the signature.

*7) Skeletonisation:* Skeletonisation removes pixels on the boundaries of objects but does not allow objects to break apart. The pixels remaining make up the image skeleton. Skeletonisation is performed on every image after the image is resized into a fixed size(Fig-6).

## III. FEATURE EXTRACTION

Feature extraction is a process used to capture essential details of the input image sample.

### A. Euclidian distances from vertical sectioning of the signature:

The signature image after fitting at the middle of a frame is scanned first column wise and then row wise to obtain the geometrical center of the signature. The signature is divided into two parts by the column of the center. Each half of the

World Academy of Science, Engineering and Technology
International Journal of Electronics and Communication Engineering
Vol:4, No:8, 2010

IMAGE SKELETON



Fig. 6. Image Skeleton



Fig. 8. Horizontal sectioning of the Signature

signature is then treated as two different samples and the process of finding the center is repeated for each half. This sectioning is done to obtain 15 geometrical centers(Fig-7). Now the distance of every other point from the geometrical center of the whole signature is calculated. Thus we obtain 14 distance values.
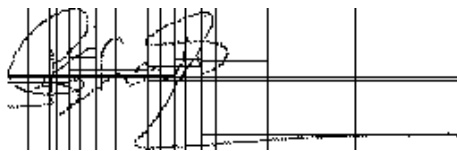


Fig. 7. Vertical Sectioning Of the Signature

*B. Euclidian distances from horizontal sectioning of the signature:*

The signature image after fitting at the middle of a frame is scanned first row wise and then column wise to obtain the geometrical center of the signature. The signature is divided into two parts by the row of the center. Each half of the signature is then treated as two different samples and the process of finding the center is repeated for each half. This sectioning is also repeated to obtain 15 geometrical centers(Fig-8. The distance of every other point from the geometrical center of the whole signature obtained from horizontal sectioning of the signature is calculated. Fourteen distance values are obtained this way.

*1) Sum of pixel values row wise, column wise and diagonals::* Sum of pixel values are calculated row wise and column wise. Sum of the diagonal elements are also calculated. Standard deviation of the row wise sum and column wise sum are calculated and used as feature vectors. Sum of the diagonal elements are used as they are.

*2) Projection of the signature image::* Projection data from the signature is obtained by using Radon transform. The signature is projected from ten different directions 15 degree apart. That gives important information about the distribution of pixel mass in the signature image field. Standard deviation, mean and median are calculated for each projection of the signature and used as features.

## IV. LEARNING AND CLASSIFICATION MODULE IN ANNs

ANNs are successful in pattern recognition application [9] as they have the ability to learn. Here ANNs are trained to perform signature verification. In the classification module features from test signatures were applied as the input to the ANNs after preprocessing stage. This output is then compared with the pattern in the database and a result is displayed.
Three ANNs were trained with features obtained in our work with a mind to reduce FAR. The first one was trained with Euclidian distances obtained from vertical sectioning of the signature. The second one was trained with Euclidian distances from horizontal sectioning of the signature images. The third network was trained with the other two feature sets together. A fourth ANN is trained with image skeleton obtained as described in Section II-B7.

*A. Training the network:*

Five signatures from each individual is used in the training. After taking each input signature, it is preprocessed and is fed to the feature extraction module. Input matrices for the training of the first three ANNs are prepared After getting all the input signature features.
Average of the corresponding Euclidian distances obtained from vertical sectioning of the five signatures of an individual is taken. Thus from five signature of an individual a template feature set is obtained. The features in this feature set are arranged as a column matrix. After getting the signatures from all the persons, the input matrix is formed by appending each feature column to the earlier one. This is the input matrix for the first network. The target is generated after formation of the input matrix, so that each individual is well classified.
The above mentioned steps are repeated to obtain the input and the target matrix for the second and the third network respectively. The image skeleton is converted into a single column by appending columns of the image matrix at the bottom of previous column. Then each single column is appended on the previous column obtained from previous signature in the

World Academy of Science, Engineering and Technology
International Journal of Electronics and Communication Engineering
Vol:4, No:8, 2010

same way. That will give the input matrix for the fourth ANN. Target matrix is formed such that classification is feasible. Let $\sum F_i[n]$ be the features extracted. The outputs extracted from the ANN classifiers can be expressed as:

$$\sum Y = \sum_p g_p[\sum_j g_j\{R_m(i,j) \times w_2(j,k)\} \times w_3(k,p)] \quad (5)$$

where

$$R_m(i,j) = \sum_i g_i\{F_m(i,j) \times w_1(i,j) \quad (6)$$

and g(.) are activation functions. Let the desired output be $d_m$. Then for each of the feature sets, the error maybe expressed as

$$e_m = \sum_m [y_m - d_m] \quad (7)$$

The error reduces as the training is continued. Batch training method is adopted as it accelerates the speed of training and the rate of convergence of the MSE to the desired value [9] [10]. The steps are as below:

- **Initialization**: Initialize weight matrix **W** with random values between [-1,1] if a tan-sigmoid function is used as an activation function and between [0, 1] if log-sigmoid function is used as activation function. **W** is a matrix of $CxP$ where $P$ is the length of the feature vector used for each of the $C$ classes.
- **Presentation of training samples**: Input is $p_m = [p_{m1}, p_{m2}.....p_{mL}]$. The desired output is $d_m=[d_{m1}, d_{m2}......d_{mL}]$.
  - Compute the values of the hidden nodes as:

$$net_{mj}^h = \sum_{i=1}^L w_{ji}^h p^{mi} + \emptyset_j^h \quad (8)$$

  - Calculate the output from the hidden layer as

$$o_{mj}^h = f_j^h(net_{mj}^h) \quad (9)$$

where f(x)=$\frac{1}{e^x}$
or f(x)=$\frac{e^x - e^{-x}}{e^x + e^{-x}}$
depending upon the choice of the activation function.
  - Calculate the values of the output node as:

$$o_{mk}^o = f_k^o(net_{mj}^o) \quad (10)$$

- **Forward Computation**: Compute the errors:

$$e_{jn} = d_{jn} - o_{jn} \quad (11)$$

Calculate the mean square error(MSE) as :

$$MSE = \frac{\sum_{j=1}^M \sum_{n=1}^L e_{jn}^2}{2M} \quad (12)$$

Error terms for the output layer is:

$$\delta_{mk}^o = o_{mk}^o(1 - o_{mk}^o)e_{mn} \quad (13)$$

Error terms for the hidden layer:

$$\delta_{mk}^h = o_{mk}^h(1 - o_{mk}^h)\sum_j \delta_{mj}^o w_{jk}^o \quad (14)$$

- **Weight Update**:

  - Between the output and hidden layers

$$w_{kj}^o(t+1) = w_{kj}^o(t) + \eta\delta_{mk}^o o_{mj} \quad (15)$$

where $\eta$ is the learning rate(0¡$\eta$¡1). For faster convergence a momentum term($\alpha$)maybe added as:

$$w_{kj}^o(t+1) = w_{kj}^o(t)+\eta\delta_{mk}^o o_{mj}+\alpha(w_{kj}^o(t+1)-w_{kj}) \quad (16)$$

  - Between the hidden layer and input layer:

$$w_{ji}^h(t+1) = w_{ji}^h(t) + \eta\delta_{mj}^h p_i \quad (17)$$

A momentum term maybe added as:

$$w_{ji}^h(t+1) = w_{ji}^h(t)+\eta\delta_{mj}^h p_i+\alpha(w_{ji}^o(t+1)-w_{ji}) \quad (18)$$

One cycle through the complete training set forms one epoch. The above is repeated till MSE meets the performance criteria. While repeating the above the number of epoch elapsed is counted. A few methods used for GFFANN training includes:

- Gradient Descent (GDBP)
- Gradient Descent with Momentum BP (GDMBP)
- Gradient Descent with Adaptive Learning Rate BP (GDALRBP) and
- Gradient Descent with Adaptive Learning Rate and Momentum BP (GDALMBP).

Training continues till the error between the actual output and the desired output, $\overline{e}$ approaches the desired goal. Several configurations of the ANN can be utilize for training. The ANN configurations used have one input layer, one hidden layer and one output layer. A single hidden layered ANN is found to be computationally efficient for the work as 2-hidden layered or a 3-hidden layered ANNs are found to be showing no significant performance improvement at the cost of slowing down training. The choice of the length of the hidden layers have been fixed by not following any definite reasoning but by using trial and error method. For this case several sizes of the hidden layer have been considered. Table I shows the performance obtained during training by varying the size of the hidden layer.

TABLE I
PERFORMANCE VARIATION AFTER 1000 EPOCHS DURING TRAINING OF AN ANN WITH VARIATION OF SIZE OF THE HIDDEN LAYER

| Case | Size of hidden layer (x input layer) | MSE Attained | Precision attained in % |
|---|---|---|---|
| 1 | 0.75 | $1.2 \times 10^{-3}$ | 87.1 |
| 2 | 1.0 | $0.56 \times 10^{-3}$ | 87.8 |
| 3 | 1.25 | $0.8 \times 10^{-4}$ | 87.1 |
| 4 | 1.5 | $0.3 \times 10^{-4}$ | 90.1 |
| 5 | 1.75 | $0.6 \times 10^{-4}$ | 89.2 |
| 6 | 2 | $0.7 \times 10^{-4}$ | 89.8 |

The case where the size of the hidden layer taken to be 1.5 times to that of the input layer is found to be computationally efficient. Its mean square error (MSE) convergence rate and learning ability is found to be superior to the rest of the cases. Hence, the size of the hidden layer of the ANNs considered is 1.5 times to that of the input layer. The size of the input layer depends upon the length of the input vector and the output layer represents the number of parameters. Noise free and noised data were used for the training.

World Academy of Science, Engineering and Technology
International Journal of Electronics and Communication Engineering
Vol:4, No:8, 2010

*B. Testing:*

The scanned signature to be verified is preprocessed where rotation correction is also done. The centered signature is fed to the feature extraction module and the features mentioned above are extracted. The features are then fed to the input formation module and inputs for the corresponding neural networks are formed in the same way as in the training stage. The networks are simulated with the corresponding inputs. Classification is done on the basis of outputs of simulation of all the networks at a time. Threshold condition is set to take care of the trade off between FAR and FRR.

## V. RESULTS

As mentioned earlier, we used four ANNs, in our experiments. Network parameters were experimentally chosen and the best configurations were used for comparison purposes. It is observed that gradient descent with momentum and adaptive learning rate back propagation had produced the best results in training. The numbers of neurons and their transfer functions in a layer varies from network to network.

The logic for classification of a test signature is also derived experimentally considering the simulated outputs of all the networks all together. For a typical test signature simulated output all the network may not agree with same result. Logic is developed prioritizing the networks for their outputs so that the trade off between FAR and FRR is taken in care.

We found that All the feature set can uniquely classify a signature. But the FAR and FRR obtained from different feature set are different. It is found that FAR is smaller than FRR for features obtained from vertical sectioning of the signature and FRR is smaller than FAR for features obtained from horizontal sectioning of the signature. FAR and FRR obtained was 15 percent and 25 percent respectively for features obtained from vertical sectioning of the signature and 30 percent and 13 percent for features obtained from horizontal sectioning of the signature.

The third ANN with the features like row wise, column wise and diagonal sum and the projections attained FRR of of 25 percent and FAR of 20 percent. The image skeleton also provided reasonable result. The FAR and FRR obtained was 15 percent each for the skeleton. The simulated output of each ANN for a test signature varies. We trialed with few signature samples and developed a logic to include all the four ANNs for classification of a single test signature. With that scheme the FRR and FAR obtained was 10 percent and 15 percent respectively. To evaluate FAR we used random forgeries.

## VI. CONCLUSION

The method described here is found to be successful in dealing with tilted and forged signature. The feature set formulated is found to be effective enough to capture finer variations in the signature. The work can be extended to include a wide class of signature and form an effective verification system.

## REFERENCES

[1] A. Prasad: *An Offline Signature Verification System.*, Surathkal -574157, amaresh191@yahoo.com..

[2] Y. Kato, D. Muramatsu and T. Matsumoto: *"A Sequential Monte Carlo Algorithm for Adaptation to Intersession Variability in On-line Signature Verification"*, Department of Electrical Engineering and Bioscience,Waseda University, 3-4-1 Okubo, Shinjuku, Tokyo 169-8555, Japan,

[3] D. Jena, B. Majhi and S. K. Jena: *"Improved Offline Signature Verification Scheme Using Feature Point Extraction Method"* , National Institute of Technology Rourkela, Orissa, India, Journal of Computer Science 4 (2): 111-116, 2008 ISSN 1549-3636 2008 Science Publications

[4] T. E. Emrezgndz and M. E. Karslgil: *"OffLine Signature Verification And Recognition By Support Vector Machine"*, Computer Engineering Department, Yldz Technical University Yldz , Istanbul, Turkey,

[5] A. T. Wilson: *"Offline Handwriting Recognition Using Articial Neural Networks"*, University of Minnesota, Morris Morris.,

[6] C. B. Owen and F. Makedon: High Quality Alias Free Image Rotation, Proceedings of $30^{th}$ Asilomar Conference on Signals, Systems, and Computers Pacific Grove, California, November 2-6, 1996.

[7] I. S. I. Abuhaiba: *"Offine Signature Verication Using Graph Matching"*, Department of Electrical and Computer Engineering, Islamic University of Gaza,

[8] Mathworks: *"Inc. Matlab Toolbox"*, http://www.mathworks.com,

[9] S. Haykin, *Neural Networks A Comprehensive Foundation*, Pearson Education, $2^{nd}$ edition, 2003.

[10] S. Kumar, *Neural Networks A Classroom Approach*, Tata McGraw Hill, $8^{th}$ Reprint, 2009.

**Manash Pratim Sarma** , completed MSc in Electronics from Gauhati University, Guwahati Assam, India in 2008. He also completed MTech from Tezpur University, Assam, India. Presently he is a research associate at Department of Electronics and Communication Technology, Gauhati University, Assam, India. His field of interest includes Wireless Communication and ANN applications.

**Kandarpa Kumar Sarma** , presently with the Department of Electronics and Communication Technology, Gauhati University, Assam, India, completed MSc in Electronics from Gauhati University in 1997 and MTech in Digital Signal Processing from IIT Guwahati, Guwahati, India in 2005 where he further continued his research work. His areas of interest include Applications of ANNs and Neuro-Computing, Document Image Analysis, 3-G Mobile Communication and Smart Antenna.

**Hirendra Das** , completed MSc in Electronics from Gauhati University, Guwahati Assam, India in 2009. Presently he is a project fellow at Department of Electronics and Communication Technology, Gauhati University, Assam, India. His field of interest includes ANN applications and Nanotechnology.