Design of Artificial Neural Network for Handwritten Signature Recognition

Debnath Bhattacharyya, and Tai-hoon Kim

***Abstract*—** Numerous approaches have been proposed for Handwritten Signature Identification systems. Various successful domain specific applications using ANN (artificial neural networks) can be found, but not flexible enough for generalization. Besides all, one approach that has shown great promise is the use of ANN in the Handwritten Signature Identification. To avoid forgery and ensure the confidentiality of Information in the field of Information Technology Security an inseparable part of it. In order to deal with security, Authentication plays an important role. This paper presents a view on the signature recognition technique and we also discussed about a technology of Signature Authentication by Back-propagation Algorithm with application. The purpose of this technique is to ensure that the rendered services are accessed only by a legitimate user, and not anyone else. By using this method it is possible to confirm or establish an individual's identity. In this paper we explore a new recognition technique; an ANN is trained to identify patterns among different supplied handwriting samples. Handwritten signature samples are considered input for our artificial neural network model and typically weights also supplied for recognition.

***Keywords*—** Recognition, neural network, signature, pattern recognition, authentication, and security.

I. Introduction

HUMANS recognize each other according to their various characteristics for ages. We recognize others by their face when we meet them and by their voice as we speak to them. Identity verification (authentication) in computer systems has been traditionally based on something that one has (key, magnetic or chip card) or one knows (PIN, password). Things like keys or cards, however, tend to get stolen or lost and passwords are often forgotten or disclosed.

To achieve more reliable verification or identification we should use something that really characterizes the given person. Biometrics offer automated methods of identity verification or identification on the principle of measurable physiological or behavioral characteristics such as a signature or a voice sample. These characteristics should not be duplicable, but it is unfortunately often possible to create a copy that is accepted by the biometric system as a true sample.

Signature authentication technology uses the dynamic analysis of a signature to authenticate a person. The technology is based on measuring speed, pressure and angle used by the person when a signature is produced. This technology uses the individual's handwritten signature as a basis for authentication of entities and data. An electronic drawing tablet and stylus are used to record the direction, speed and coordinates of a handwritten signature. There is no encryption or message confidentiality offered yet with signature dynamics, but more modern examples use one-way hash functions to encrypt the signature dynamics and data and append it to the document being signed. In one iteration, created by Topaz Systems, the signature actually disappears from view if the document is tampered with after signature.

Prof. Debnath Bhattacharyya, Director, Science and Engineering Research Support Center, India Part, Korea. (email: debnathb@gmail.com).

Dr. Tai-hoon Kim, Department of Multimedia Engineering, Hannam University, Daejeon, Korea. (email: taihoonn@empal.com).

II. Earlier Works

A number of biometric methods have been introduced, but few have gained wide acceptance. Before we discuss about our approach, we have concentrated on some techniques to gather knowledge.

In 2001 Adrian Perrig introduces the BiBa signature scheme. BiBa signature scheme is a new signature construction that uses one-way functions without trapdoors.The most important features of BiBa signature scheme is a low verification overhead and a relatively small signature size. In comparison to other one-way function based signature schemes, BiBa has smaller signatures and is at least twice as fast to verify (which probably makes it one of the fastest signature scheme to date for verification). BiBa stands for Bins and Balls signature a collision of balls under a hash function in bins forms the signature [3].

Alexander T. Ihler, John W. Fisher III, and Alan S. Willsky present modeling dynamical processes in 2002. They use this methodology to model handwriting stroke data, specifically signatures, as a dynamical system and show that it is possible to learn a model capturing their dynamics for use either in synthesizing realistic signatures and in discriminating between signatures and forgeries even though no forgeries have been used in constructing the model [4].

In the year of 2004 Sascha Schimke, Claus Vielhauer, Jana Dittmann in the Proceedings of the 17th International Conference on Pattern Recognition they discussed about the problem of authenticating for on-line signature. They are explained a method for on-line signature authentication, which is based on a event-string modeling of features derived from pen-position and pressure signals of digitizer tablets [1].

The intra- class variability is one of the key problems in biometrics. One possible approach to solve this inexact matching problem known from textual pattern recognition is the edit distance, which defines a way to measure the difference or distance between two strings by transforming one string into the other, applying a series of edit operations on individual characters. For every character in the first string, a sequence of the operations insert, delete and replace can be performed to transform that string into the other. The edit distance between two strings is then defined as is the minimum number of edit operations needed to transform the first string into the second. Since they used the scientist levenshtein distance algorithm introduced by v. I. Levenshtein's which is commonly known as levenshtein distance [2].

The edit distance D between two strings S1 and S2 can be calculated by a dynamic programming algorithm and can be formally described by the following recursion [2].

$$\left.\begin{array}{l} D(i, j) := \min[\, D(i-1, j) + wd, \\ D(i, j-1) + wi, \\ D(i-1, j-1) + wr \,] \\ \\ D(i, 0) := D(i-1, 0) + wd \\ D(0, j) := D(0, j-1) + wi \\ D(0, 0) := 0 \end{array}\right\} \forall i, j > 0$$

Use of Artificial neural networks in pattern recognition is a popular technique. Neural network based invariant character recognition system using double backpropagation model has proposed, the model consists of two parts. The first is a preprocessor, which is intended to produce a translation, rotation and scale invariant representation of the input pattern. The second is a neural net classifier. The outputs produced by the preprocessor at the first stage are classified by this neural net classifier trained by a learning algorithm called double back-propagation. The recognition system was tested with ten numeric digits (0-9). The test included rotated, scaled and translated version of exemplar patterns [6]. An artificial neural network is trained to identify similarities and patterns among different handwriting samples. Artificial neural network techniques have also proven helpful in the preprocessing that must take place before a handwriting sample can be considered suitable input for an artificial neural network [7]. Another major research is the integration of a feed-forward back-propagation neuronal network [10]. The inputs for this neuronal network are the individual tokens of an image, and a token normally consists of a cosinus and sinus angle, the amount of input layers for this network are the amount of tokens multiplied by two [8].

The used method in Neural Network is, various characters are taught to the network in a supervised manner. A character is presented to the system and is assigned a particular label. Several variant patterns of the same character are taught to the network under the same label. Hence the network learns various possible variations of a single pattern and becomes adaptive in nature [9].

Peter Pivonka, P. Nepevny, in 2005, proposed a new controller by combination of Generalized Predictive Control (GPC) algorithm and Neural Network model, with many advantages. Neural model has the ability to observe system changes and adapted itself, therefore regulator based on this model also will be adaptive. Algorithm was implemented in MATLAB-Simulink with aspect of future implementation to Programmable Logic Controller (PLC). It was tested on mathematical and physical models in soft-real-time realization. Predictive controller in comparison with classical PSD controller and the advantages and disadvantages were shown [11].

Peter Pivonka, et al., in 2006, presented an algorithm using Bayesian information criterion (BIC) to systematically choose the optimal multilayer network structure, via the number of hidden layers and hidden nodes of each layer, of backpropagation (BP) networks. They showed the Simulation results with daily data on stock prices in the Thai market that the algorithm performed satisfactorily. Moreover, the proposed algorithm also compared to Daqi-Shouyi method [12].

Stergios Papadimitrioy Konstantinos Terzidis, in 2007, proposed a fuzzy system that approximated the accurate set of rules keeping only the more important aspects of the data. The approximation algorithms either received an a priori description of a set of fuzzy sets or, especially for the case when interpretable fuzzy sets could not be prespecified by the experts, the algorithm presented for building them automatically. After the construction of the interpretable fuzzy partitions, the developed algorithms extract from the SVFI rules a small and concise set of interpretable rules. Finally, the Pseudo Outer Product (POP) fuzzy rule selection ordered the interpretable rules by using a Hebbian like evaluation in order to present the designer with the most capable rules [13].

Mehmet S. Unluturk, et. al., 2009, conducted an extensive study on the use of Neural Network in Biometric researches. They developed the emotion recognition neural network (ERNN) to classify the voice signals for emotion recognition. The ERNN had 128 input nodes, 20 hidden neurons, and three summing output nodes. A set of 97932 training sets was used to train the ERNN. A new set of 24483 testing sets was utilized to test the ERNN performance. The samples tested for voice recognition were acquired from the movies "Anger Management" and "Pick of Destiny". ERNN achieved an average recognition performance of 100%. This high level of recognition suggested that the ERNN was a promising method for emotion recognition in computer applications [14].

H. Erdinc Kocer, and Novruz Allahverdi, in 2008, proposed [15] a neural network based iris recognition approach by analyzing iris patterns. They used Average absolute deviation (AAD) algorithm for feature extraction. In matching process, Multi-Layered Perceptron (MLP) and Modular Neural Networks (MNN) applied to the iris feature vector for classifying the iris images. They focused on measuring the performance of MNN in iris recognition system compared with Multi-Layered Perceptron (MLP) neural network in their proposed work. The gray-level iris images, experimented in their work, were obtained from Institute of Automation Chinese Academy of Science (CASIA) iris images database and Departments of Informatics University of Beira Interior (UBIRIS) iris images database.

Erida Dule, et. al., in 2010, worked on outdoor vehicle images to determine the color of the vehicle and color classes. In this work, the Performances of different feature sets obtained by various color spaces and different classification methods were taken into account in order to improve the outdoor vehicle color recognition. Also, different Region of Interest (ROI) and feature vector construction methods were developed for gain better performance. They also examined two ROI (smooth hood peace and semi front vehicle), three classification methods (K-Nearest Neighbors, Artificial Neural Networks, and Support Vector Machines), and all possible combinations of sixteen color space components as different feature sets. They claimed 83.50% success in their experiments. As a result, the best performer combination of the classifier, the choice of the ROI, and the feature vector were demonstrated.

In March, 2007, [17] proposed a handwritten signature recognition technique using ANN (Artificial Neural Network). Authors explained the system as follow:

Neural Network Learning Process (NNLP): In this Process, network will be trained about the image data. An image data is presented to the system and is assigned a particular label. Several variant patterns of the same image are taught to the network under the same label. Hence the network learns different possible variations of a single pattern and becomes adaptive in nature. During the training process, the input to the neural network is the input matrix M, we defined as follows:

```
If Image[i, j] =1 Then
InputMatrix[i, j] =1
Else
InputMatrix[i, j] = -1
```

Considering Monochrome Image (with White as 0 and Black as 1).

The generated InputMatrix M will be now used as input to the neural network. Typical, weights have to be adjusted during Network Training. In this method of learning, each candidate Image data taught to the network possesses a corresponding weight matrix. For the k^{th} image data to be taught to the network, the weight matrix is denoted by W_k. As learning of the image data progresses, it is this weight matrix that is updated. At the commencement of teaching (supervised training), this matrix is initialized to zero. Whenever an image data is to be taught to the network, an input pattern representing that image is submitted to the network. The network is then instructed to identify this pattern as, say, the k^{th} character in a knowledge base of images. That means that the pattern is assigned a label k. In accordance with this, the weight matrix W_k is updated in the following technique (pseudo code):

```
Step 1: for i =1 to x
Step 2: loop
Step 3:     for j =1 to y
Step 4:     loop
Step 5:       Wk [i, j] = Wk [i, j] + M[i, j]
Step 6:     End loop
Step 7: End loop
```

Here x and y are the dimensions of the matrix W_k (and M).

The three input patterns representing D that are presented to the system for it to learn.

Remarkably, Fig 1(b) is slightly different from (a) and (c). This is the nature of Handwritten Signature. Signature may vary from person to person as well as time to time.

Fig. 1, gives the weight matrix, say, W_D corresponding to the image we compare. The matrix is has been updated thrice to learn the image we compare, here it is alphabet D. It should be noted that this matrix is specific to the image we compare; here it is alphabet D as an image alone.

Characteristics of the above W_D matrix:

The matrix-elements with higher (positive) values (3) are the ones that stand for the most commonly occurring image-pixels.

The elements with lesser or negative values (-3) stand for pixels which appear less frequently in the images.

Neural networks learn through such updating of their weights. Each time, the weights are adjusted in such a manner as to give an output closer to the desired output than before. The weights may represent the priority of a parameter, which in the instant case is the occurrence of a particular pixel in an image pattern. It can be seen that the weights of the most frequent pixels are higher and usually positive and those of the uncommon ones are lower or often negative. The matrix therefore assigns importance to pixels on the basis of their frequency of occurrence in the pattern. In other words, highly probable pixels are assigned higher priority while the less-frequent ones are seized. However, all labeled patterns are treated without bias and include impartial adaptation in the system.

Neural Network Design Structure (NNDS): Consider the following structure, used in our research work:–

In the structure, the candidate pattern I is the input. The block 'M' provides the input matrix M to the weight blocks W_k for each k. There are totally n weight blocks for the totally n images to be taught (or already taught) to the system.

Main part of this work is the integration of a feed-forward propagation neuronal network. As described earlier the inputs for this neuronal network are the individual tokens of an image, and the amount of input layers for this network is the amount of tokens.

Neural Network Statistical Scores (NNSS): This is a product of corresponding elements of the weight matrix W_k of the k^{th} learnt pattern and an input pattern I as its candidate [8]. Mathematical Expression is as follow:

$$\theta(k) = \sum_{i=1}^{x} \sum_{j=1}^{y} W_k(i, j) * I(i, j)$$

In the training process where M was the processed input matrix, in the recognition process, the binary image matrix I is directly fed to the system for recognition.

Statistical Analysis of Ideal Weight Score (μ): This gives the sum total of all the positive elements of the weight matrix of a learnt pattern. It can be computed as follow (with m (k) initialized to 0 each time):

Step 1: for i=1 to x
Step 2: Loop
Step 3: for j=1 to y
Step 4: Loop
Step 5: if ($W_k(i, j) > 0$) then
Step 6: $\mu(k) = \mu(k) + W_k(i, j)$
Step 7: End loop
Step 8: End loop

Result by Recognition Quotient (Q) Analysis: This is another statistical value which gives a measure of how well the recognition system identifies an input pattern as a matching candidate for one of its many learnt patterns. It is simply given by:

$$Q(k) = \theta(k) / \boldsymbol{\mu}(k)$$

The greater the value of Quotient (Q), the more confidence does the system on the input pattern as being similar to a pattern already known to it. The classification of input patterns now follows the following trivial procedure:-

For an input candidate pattern I, calculate the recognition quotient Q(k) for each learnt pattern k.

Determine the value of k for which Q(k) has the maximum value.

Too low maximum value of Q(k) (say less than 0.5) indicates poor recognition. In such a case:

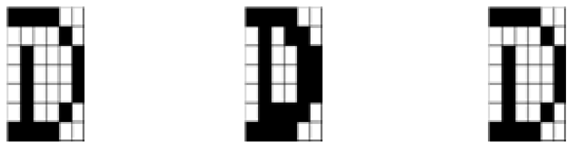
a. Conclude that the candidate pattern does not exist within the knowledge base, OR
b. Teach the candidate pattern to the network till a satisfactory value of Q(k) is obtained.

Conditionally, identify the input candidate pattern to the k_{th} learnt pattern OR proceed with the training for better performance.

The selector gives an output k by making the best selection as of the aforementioned algorithm. The adaptive performance of the network can easily be tested by an example: we submit two hand-drawn patterns representing D and b respectively to the system that has already learnt only the pattern D. The recognition quotient returned by the trained system is mentioned aside. Fig. 2, shows the weight matrix generated from 3 patterns stated in Fig. 1 (a), (b) and (c).

The system simply discards the pattern which will return very low value for Q, i.e., 0.20. It can be observed by regular teaching, that the system develops on its ability to identify a matching pattern and reject non-matching patterns. Thus, regular supervised teaching enhanced performance of the system. The neural system has some direct advantages like,

a. Due to adaptive method, system can tolerate minor changes in pattern identification.
b. Knowledge based system; system allows continuous teaching and different pattern of same object.
c. The system is generalized.



(a) (b) (c)

Fig. 1. Three different pattern representations of a Character.

$$W_D = \begin{array}{|c|c|c|c|c|c|} \hline 3 & 3 & 3 & 3 & -3 & -3 \\ \hline -3 & -1 & -3 & -1 & 3 & -3 \\ \hline -3 & 3 & -3 & -3 & -1 & 3 \\ \hline -3 & 3 & -3 & -3 & -1 & 3 \\ \hline -3 & 3 & -3 & -3 & -1 & 3 \\ \hline -3 & 3 & -1 & -1 & 3 & -3 \\ \hline 3 & 3 & 3 & 3 & -3 & -3 \\ \hline \end{array}$$

Fig. 2. Weight Matrix.

III. Our Work

First break the pixels of the into its RGB values. Then following steps have to be performed:

a. Calculate the corresponding Gray scale value of each pixel.
b. Create the R, G, B, Gray value table.
c. Create Neural Network with three input nodes and one output node where R, G, B values will be input and gray scale value will be output.
d. Train the weights with the above training set and store the weight matrix.
e. When you have the hypothesis, break the second image.
f. Input the pixel values of the new image into your network.

After performing the above steps, now we should check the followings as Output:

a. The predicted image corresponding to the second image.

b. Deviation of the predicted values from the target values of the second image.

Here in this paper we discuss six algorithms which are required for our proposed work.

A. Gray value calculation of bitmap image(s)

GRAYVALUECALCULATION(colorR,colorG,colorB)
INPUT: Each bitmap pixel (i.e. R, G, B).
OUTPUT: GRAY VALUE of each input pixel.
colorR, colorG, colorB can be represented by Red, Green and Blue respectively.

Repeat

$$gradient = \frac{(colorR + colorG + colorB)}{3} \quad \ldots\ldots (1)$$

Each R, G, B of bitmap pixel to be typecasted from unsigned char to unsigned int before the above equation is computed.

When gradient is being computed, it must be typecasted from float to unsigned char (say initially gradient was float type).

until
the termination condition is met i.e. image_file is exhausted.

B. Random numbers generation

Random numbers:
for i 0 to 20 ***do***
Here 0 to 20 represented that, our algorithm gives 21
Random numbers within specific range from -0.5 to 0.5

a ←rand() ………………………. (2)
b ←a mod 1000………………… (3)
f ← -0.5 + …………………… (4)
end;

C. Transpose of Matrix

INPUT: A matrix size m x n.
OUTPUT: Transposed B matrix.
for row 0 to (m - 1) ***do***
for col 0 to (n - 1) ***do***
B[row][col] ← A[col][row] …… (5)
end;
end;

D. Multiplication of two matrices

INPUT: Matrix A, size m x n
and matrix B, size k x q .
OUTPUT: Matrix C, size m x q.
for row 0 to (m - 1) ***do***
for col 0 to (q - 1) ***do***
c[row][col] ← 0 …….. (6)
for ip 0 to (n - 1) ***do***
C[row][col] ← C[row][col]+A[row][ip] * B[ip][col] ……... (7)
end;
end;
end;

E. Sigmoid output of a matrix

SIGMOID(A[m][0])
INPUT: Matrix A, size m x 0.
OUTPUT: Matrix B, size m x 0.
for row ← 0 to (m - 1) ***do***
deno ← exp(A[row][0] * (-1)) ….. (8)

$$B[row][0] \leftarrow \frac{1.0}{1.0 + deno} \quad \ldots\ldots\ldots (9)$$

end;

F. Artificial Neural Network Backpropagation Algorithm

Here we use the stochastic gradient descent version of the Back-propagation algorithm for feed-forward networks containing two layers of sigmoid units for signature recognition.

BACKPROPAGATION(training_example,η, $n_{in}, n_{out}, n_{hidden}$)

Each training example is a pair of the form $(\vec{x}, \vec{t})$, where $\vec{x}$ is the vector of network input values and $\vec{t}$ is the vector of target network output values. η is the learning rate(e.g. 0:05). n_{in} is the number of network inputs. n_{hidden} the number of units in the hidden layer and n_{out}t the number of output units.

Create a feed-forward network with n_{in} inputs, n_{hidden} the number of units in the hidden layer, and n_{out} output units.

Initialize all network weights to small random numbers (e.g. between -0.5 and 0.5).

For each $(\vec{x}, \vec{t})$ in training _examples , Do
Repeat
Propagate the input forward through the network:

1. Input the instance $\vec{x}$ to the network and compute the output ou of every unit u in the network. Propagate the errors backward through the network.
2. For each network output unit k,calculate its error term:
$$\delta_k \leftarrow O_k (1 - O_k)(t_k - O_k) \quad \ldots.. (10)$$
3. For each hidden unit h,calculate its error term:
$$\delta_k \leftarrow O_k(1 - O_k \textstyle\sum_{k\in outputs} w_{kh}\ \delta_k \quad \ldots\ldots. (11)$$
4. Update each network weight W_{ji}
$$w_{ij} \leftarrow w_j + \Delta w_{ji} \text{ ; where } \Delta w_{ji} = \eta\delta j^x ij \quad \ldots\ldots. (12)$$

until
the termination condition is met, i.e. , training_example is exhausted and E $\leq \epsilon$, where ϵ very small;

IV. RESULT AND ANALYSIS

Fig. 3, is the original image, on which experimental result is shown.

Here Table 1 and Table 2 show the analysis of results. From Table 1 we get Number of iterations each pixel for training weight matrix: 7204 times

Input file for hypothesis: sig1.bmp

Input file for prediction: sig1.bmp

After 7204 times iteration of each pixel of sig1.bmp, we get corresponding two weight matrix, between input and hidden layer (left) and hidden and output layer (right), shown in Fig. 4.

Fig 3. Original Signature Image (sig1.bmp).

$$\begin{pmatrix} -17.0294 & -3.7524 & 18.1117 \\ -1.8606 & -2:0283 & 6:3205 \\ -26.1097 & -1.8606 & 23:8842 \end{pmatrix} \qquad \begin{pmatrix} -3.7849 \\ 0.1231 \\ 3.7972 \end{pmatrix}$$

Fig.4. Original Signature (left), Predicted Signature (right).

Table 2 is used for the prediction of sig2.bmp. Now, compute:

$$e = \frac{\sum_{i=0}^{n} error\ i}{n} \quad \ldots\ldots\ldots\ldots (13)$$

we get e = 0:0130 as a output of Table 1, where n = 6285 represents number of pixels.

From Table 2, we get Number of iterations each pixel for training weight matrix: 39000 times

Input file for hypothesis: sig.1.bmp

Input file for prediction: sig.2.bmp

After 39000 times iteration of each pixel of sig1.bmp, we get corresponding two weight matrices, between input and hidden layer (left) and hidden and output layer (right):

$$\begin{pmatrix} -11.9081 & 2.1354 & 2.0328 \\ -0.8034 & 2.4683 & -1.2099 \\ -47.0260 & 7.3918 & 8.4300 \end{pmatrix} \begin{pmatrix} -1.8968 \\ 0.1149 \\ 0.0329 \end{pmatrix}$$

We get e = 0:0124 as a output of Table 2, where n = 4358 represents number of pixels (applying Equation (13)).

V. CONCLUSION

To avoid forgery and ensure the confidentiality of Information in the field of Information Technology we need Security. In this paper we discuss about a technology of signature authentication by “Back-propagation algorithm” with application. It is a new approach in the field of signature authentication.

In this paper, we implement the basic algorithm of artificial neural network through Back-propagation algorithm. Here we use three (Input, output and hidden) layer, six node (three in input layer, two in hidden layer and one in output layer; because number of nodes in first hidden layer is always less than or equal to number of input layer nodes and number of nodes in last hidden layer is always greater than or equal to number of nodes in output layer), number of iteration of each pixel is 39000 times. In this paper we solve this problem through artificial neural network. In further we will solve this through genetic algorithm [5].

ACKNOWLEDGMENT

This work was supported by the Security Engineering Research Center, granted by the Korea Ministry of Knowledge Economy.

REFERENCES

[1] Sascha Schimke, Claus Vielhauer, Jana Dittmann, “Using Adapted Levenshtein Distance For On- Line Signature Authentication”, 17th International Conference On Pattern Recognition (ICPR’04), Vol. 2, 2004, pp.931-934.

[2] V. I. Levenshtein., “Binary Codes Capable Of Correcting Deletions, Insertions, And Reversals”, Soviet Physics-Doklady, Vol. 10, 1996, pp. 707–710.

[3] [Adrian Perrig, “The Biba One-Time Signature And Broadcast Authentication Protocol”, Technical Report, University Of California, Berkeley, November, 2001.

[4] A.T. Ihler, J.W. Fisher, and A.S. Willsky, "Nonparametric estimators for online signature authentication", International Conference on Acoustics, Speech, and Signal Processing, May 07-11, 2001, Salt Lake City, UT , USA, Vol. 6, pp. 3473-3476.

[5] S.Rajasekaran and G.A. Vijayalakshmi Pai, “Neural Networks, Fuzzy Logic, and Genetic Algorithms”, Prentice-Hall India Pvt., Sixth printing, May 2006.R. J. Vidmar. (1992, August). On the use of atmospheric plasmas as electromagnetic reflectors. *IEEE Trans. Plasma Sci.* [Online]. *21(3)*. pp. 876—880.

Available: http://www.halcyon.com/pub/journals/21ps03-vidmar
[6] Joarder Kamruzzaman and S. M. Aziz, "A Neural Network Based Character Recognition System Using Double Backpropagation", Malaysian Journal of Computer Science, Vol. 11 No. 1, June 1998.
[7] Andrew T. Wilson, "Off-line Handwriting Recognition Using Artificial Neural Networks", University of Minnesota, Morris.
[8] Jens Langner, "Neuronal Network based recognition system of leaf images", http://www.jens-langner.de/, Last accessed on December 02, Last accessed on December 02, 2010.
[9] Berend-Jan van der Zwaag, "Handwritten Digit Recognition: A Neural Network Demo", International Conference on Computational Intelligence Theory and Applications, Dortmund, Germany, October 1-3, 2001, LNCS, Vol. 2206/2001, pp. 762-771.
[10] Joao Ricardo Bittencourt, Fernando Santos Osorio, "Adaptive Filters for Image Processing based on Artificial Neural Networks", XIII Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI.00), Gramado , Brazil, October 17-20, 2000, on page: 336.
[11] Peter Pivonka, and P. Nepevny, "Generalized Predictive Control with Adaptive Model Based on Neural Networks", Proceedings of the 6th WSEAS International Conference on NEURAL NETWORKS, Lisbon, Portugal, June 16-18, 2005, pp. 1-4.
[12] S. Sureerattanan, Huynh Ngoc Phien, N. Sureerattanan, and Nikos E. Mastorakis, "The Optimal Multi-layer Structure of Backpropagation Networks", Proceedings of the 7th WSEAS International Conference on Neural Networks, Cavtat, Croatia, June 12-14, 2006, pp.108-113.
[13] Stergios Papadimitrioy Konstantinos Terzidis, "Classification Process Analysis of Bioinformatics Data with a Support Vector Fuzzy Inference System", Proceedings of the 8th WSEAS International Conference on Neural Networks, Vancouver, British Columbia, Canada, June 19-21, 2007, pp. 90-95.
[14] Mehmet S. Unluturk, Kaya Oguz, and Coskun Atay, "Emotion Recognition Using Neural Networks", Proceedings of the 10th WSEAS International Conference on NEURAL NETWORKS, Prague, Czech Republic, March 23-25, 2009, pp. 82-85.
[15] H. Erdinc Kocer, and Novruz Allahverdi, "An Efficient Iris Recognition System Based on Modular Neural Networks", 9th WSEAS International Conference on NEURAL NETWORKS (NN'08), Sofia, Bulgaria, May 2-4, 2008, pp. 42-47.
[16] Erida Dule, Muhittin Gokmen, M. Sabur Beratoglu, "A Convenient Feature Vector Construction for Vehicle Color Recognition", Proceedings of the 11th WSEAS International Conference on NEURAL NETWORKS (NN '10), G. Enescu University, Iasi, Romania, June 13-15, 2010, pp. 250-255.
[17] Samir Kumar Bandyopadhyay, Debnath Bhattacharyya and Poulami Das, "A Flexible ANN System for Handwritten Signature Identification", Proceedings of the International MultiConference of Engineers and Computer Scientists 2007 Volume II, IMECS '07, March 21 - 23, 2007, Hong Kong, LNCS, pp. 1883-1887.

Table 1. Some important output corresponding pixel number.

Pixel number	Target output	Predicted output	Error deviation
22	0.2275	0.3639	0.0093
28	0.3294	0.4266	0.0047
6283	0.4784	0.5167	0.0007
6284	0.5490	0.5355	0.0001

Table 2. Some important output corresponding pixel number.

Pixel number	Target output	Predicted output	Error deviation
22	0.0588	0.3377	0.0389
28	0.0824	0.3555	0.0373
4356	0.4863	0.546	0.0002
4357	0.4902	0.5061	0.0001

Prof. Debnath Bhattacharyya, M.Tech in Computer Science and Engineering from West Bengal University of Technology, Kolkata, India.

He is associated with Science and Engineering Research Support Center, Daejeon, Korea, as the Director of India Part. He was the Education Officer in Computer Society of India, Kolkata for 10 years. He is the former Foreign Professor of Hannam University, Korea. He has 16 Years of experience in the line of Teaching and Projects. He has published 110 Research Papers in International Journals and Conferences and Four Text Books for Computer Science. His research interests include Biometric Authentication, Medical Imaging and Image Processing.

Prof. Bhattacharyya is the Life Member of Computer Society of India, India, Honorary Member of Association of Scientists, Developers and Faculties (ASDF), Pondicherry, India, and Member of IACSIT, Singapore.

Dr. Tai-hoon Kim, M.S., Ph.D (Electricity, Electronics and Computer Engineering), currently, Professor of Hannam University, Korea.

His research interests include Multimedia security, security for IT related Products, systems, development processes, operational environments, etc. He has 14 Years of experience in Teaching & Research. He has already got distinctive Academic Records in international levels. He has published more than 200 Research papers in International & National Journals and Conferences. He has published numerous text books in the line of Computer Science as well as Security related.

Dr. Kim is the Sr. Member of IEEE, Member of ACM, Member of Springer, SERSC, Korea, etc. Dr. Kim is the corresponding Author of this research paper.