PERGAMON

Available at
www.ElsevierComputerScience.com
POWERED BY SCIENCE DIRECT®

Pattern Recognition 38 (2005) 341 – 356

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Off-line signature verification and forgery detection using fuzzy modeling

Madasu Hanmandlu[a,*], Mohd. Hafizuddin Mohd. Yusof[b], Vamsi Krishna Madasu[c]

[a]Department of Electrical Engineering, Indian Institute of Technology Delhi, Hauz Khas, New Delhi 110016, India
[b]Faculty of Information Technology, Multimedia University, Jalan Multimedia, 63100 Cyberjaya, Selangor Darul Ehsan, Malaysia
[c]School of Information Technology and Electrical Engineering, The University of Queensland, Brisbane QLD 4072, Australia

Received 21 March 2003; received in revised form 10 May 2004; accepted 10 May 2004

Abstract

Automatic signature verification is a well-established and an active area of research with numerous applications such as bank check verification, ATM access, etc. This paper proposes a novel approach to the problem of automatic off-line signature verification and forgery detection. The proposed approach is based on fuzzy modeling that employs the Takagi–Sugeno (TS) model. Signature verification and forgery detection are carried out using angle features extracted from box approach. Each feature corresponds to a fuzzy set. The features are fuzzified by an exponential membership function involved in the TS model, which is modified to include structural parameters. The structural parameters are devised to take account of possible variations due to handwriting styles and to reflect moods. The membership functions constitute weights in the TS model. The optimization of the output of the TS model with respect to the structural parameters yields the solution for the parameters. We have also derived two TS models by considering a rule for each input feature in the first formulation (Multiple rules) and by considering a single rule for all input features in the second formulation. In this work, we have found that TS model with multiple rules is better than TS model with single rule for detecting three types of forgeries; random, skilled and unskilled from a large database of sample signatures in addition to verifying genuine signatures. We have also devised three approaches, viz., an innovative approach and two intuitive approaches using the TS model with multiple rules for improved performance.

Keywords: Off-line signature verification; Forgery detection; Structural parameters; Fuzzy logic; TS model; Bank check recognition

1. Introduction

For centuries, handwritten signatures have been an integral part of consummating business transactions, contracts and agreements. The distinctiveness of a handwritten signature helps to prove the identity of the signer, while the act of signing a document represents the signer's acceptance of its terms and also codifies the document's contents as being official and complete at the time it was signed.

The four legal properties of a handwritten signature are briefly stated below:

- *Authentication*—a handwritten signature allows positive verification of the signer's identity.
- *Acceptance*—the signature conveys willful intent and acceptance of the terms stated in the document.

* Corresponding author. Tel.: +91-11-2659-1117; fax: +91-11-2658-1606.

E-mail addresses: mhmandlu@ee.iitd.ac.in (M. Hanmandlu), hafizuddin.yusof@mmu.edu.my (Mohd. Hafizuddin Mohd. Yusof), s4045138@uq.edu.au (V.K. Madasu).

0031-3203/$30.00
doi:10.1016/j.patcog.2004.05.015

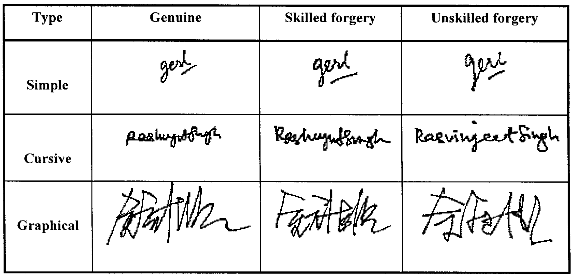

Fig. 1. Example of different types of signatures.

- *Integrity*—the signature establishes the integrity of the signed document, indicating that it has not been altered in any way.
- *Non-repudiation*—the accumulated effect of the above three factors promises such a high degree of purpose that the signer cannot deny he or she has signed.

Handwritten signatures are of different shapes and sizes and the variations in them are so immense that it is difficult for a human being to distinguish a genuine signature from a forged one by having a glance at the signature. Broadly speaking, signatures can be classified as simple, cursive or graphical based on their shape as shown in Fig. 1. Simple signatures are the ones where the person just writes his or her name. Cursive signatures are the ones that are written in a cursive way. Lastly, the signatures can be classified as graphical when cursive signatures depict geometric patterns.

Automated recognition of handwritten signatures became imperative when it was difficult to distinguish genuine signatures from simulated forgeries on the basis of visual assessment. This led to computer recognition of handwritten signatures, which though a bit slow, is more reliable and efficient.

In this paper, an automatic off-line signature verification and forgery detection system based on fuzzy modeling is proposed. This system uses the Takagi–Sugeno (TS) model incorporated with structural parameters to take account of local variations in the characteristics of the signature. The system has been tested on a large database of both genuine and forged signatures.

The main contributions of this paper are: Modification of TS model [1] with structural parameters, derivation of two formulations of this model, use of box features of Ref. [2] for signature verification forgery detection by devising an innovative approach and two intuitive approaches.

The organization of the paper is as follows. We present an overview of both signature verification and forgery detection tasks in Sections 2 and 3, respectively. The proposed system involving these tasks is described in Section 4. Section 5 tabulates the experimental results on the database of signature images. Finally, the conclusions are presented in Section 5.

2. Overview of signature verification systems

Automated handwritten signature verification can be divided into two classes, namely, on- and off-line. In the on-line signature verification systems [3] the signature is captured during the writing process, thus making the dynamic information available whereas in the off-line signature verification systems, the signature is captured once the writing process is over and thus only a static image is available. In this context mention may be made of Ammar et al. [4], who were one of the earliest researchers to extract "dynamic" information from a static image for signature verification.

As compared to on-line signature verification systems, off-line systems are difficult to design as many desirable characteristics such as the order of strokes, the velocity and other dynamic information are not available in the off-line case. The verification process has to wholly rely on the features that can be extracted from the trace of the static signature image only. Although difficult to design, off-line signature verification is crucial for determining the writer identification as most of the financial transactions in present times are still carried out on paper. Therefore, it becomes all the more essential to verify a signature for its authenticity. The design of any signature verification system generally requires the solution of five sub-problems: data acquisition, pre-processing, feature extraction, comparison process and performance evaluation [5].

Surveys of the state of the art off-line signature verification systems designed up to 1993 appear in Refs. [5–7]. Another survey article [8] has summarized the approaches used for off-line signature verification from 1993 to 2000. We present here a review of a few papers in this field, which have not been covered in the survey articles. The emphasis of these papers is mainly on fuzzy-based techniques for off-line signature verification.

An off-line signature system consisting of signature recognition and verification is proposed in Ref. [9]. In this, the recognition phase is based on the multi-stage classifier and a combination of global and local features whereas the verification is done using fuzzy concepts. HMM-based approach in Ref. [10] derives dynamically and automatically the author dependent parameters to set up an optimal decision rule for off-line verification process. Here the cross validation principle is used to derive not only the best HMM models, but also an optimal acceptation/rejection threshold for each author. This threshold leads to a high discrimination between the authors and impostors in the context of random forgeries.

Signature verification is also attempted using the Pseudo-Bacterial Genetic Algorithm (PBGA) [11], which introduces a new operation called bacterial operation. Its basic idea is to try to improve parts of chromosomes. In the cases where there are weak inter-relations within the parameters encoded in one chromosome, it should be possible to perform optimization in parts. As a test problem, the PBGA was applied for the discovery of fuzzy rules. The rules are units themselves and they are constituted by several parameters to be optimized, however, the performance of a fuzzy system is obtained synergistically as a sum of the outputs of several rules. The PBGA was then applied for the extraction of personal features for signature verification.

A pseudo-outer product-based fuzzy neural network drives the signature verification system in Ref. [12]. This system is primarily used for verifying skilled forgeries. Signature verification using TS model is reported in Ref. [1] and features for this model are drawn from the box approach of Ref. [2]. In the present work, we follow the same features as in Ref. [2] but the TS model is modified to enhance its capability for the detection of forgeries.

3. Overview of forgery detection systems

Automatic examination of questioned signatures did not come into being until the advent of computers in the 1960s. As computer systems became more powerful and more affordable, designing an automatic forgery detection system became an active research subject. Most of the work in off-line forgery detection, however, has been on random or simple forgeries and less on skilled or simulated forgeries [21–25]. Before looking into the landmark contributions in the area of forgery detection, we first enumerate the types of forgeries.

3.1. Types of forgeries

The forgeries involved in handwritten signatures have been categorized based on their characteristic features [6]. We have also attempted to classify the various kinds of forgeries into the following types:

1. Random forgery—The signer uses the name of the victim in his own style to create a forgery known as the simple forgery or random forgery. This forgery accounts for the majority of the forgery cases although they are very easy to detect even by the naked eye [13].
2. Unskilled forgery—The signer imitates the signature in his own style without any knowledge of the spelling and does not have any prior experience. The imitation is preceded by observing the signature closely for a while.
3. Skilled forgery—Undoubtedly the most difficult of all forgeries is created by professional impostors or persons who have experience in copying the signature. For achieving this one could either trace or imitate the signature by hard way.

In the 1980s, Ammar et al. [4] have worked on the detection of skilled forgeries. They have calculated the statistics of dark pixels and used them to identify changes in the global flow of the writing. The later work of Ammar [25] is based on reference patterns, namely the horizontal and vertical positions of the signature image. The projections of the questioned signature and the reference are compared using Euclidean distance. Guo et al. [14] have presented an algorithm for the detection of skilled forgeries based on a local correspondence between a questioned signature and a model obtained a priori. Writer-dependent properties are measured at the sub-stroke level and a cost function is trained for each writer.

Forged samples of a genuine signature are not readily available as it is difficult to imitate the various styles of signatures by amateurs for producing the unskilled forgeries and by professional impostors for producing the skilled forgeries. Keeping this point in mind and considering the real-world scenario, we have trained our system with only genuine signatures, i.e., none of the forgeries were used for training the system. Most of the signature verification systems trained with both genuine and forged signatures was subject to errors. For example, the automatic off-line signature verification of Pender [15] has a false acceptance rate (FAR) of 100% when trained with only genuine signatures. This means that it could not distinguish even a single forgery from genuine signatures when the system is not trained with the samples of forged signatures.

4. The proposed system

The proposed system includes both signature verification and forgery detection parts. The difference between the two parts is that verification is based on inherent characteristics of a signer whereas the detection is based on specification of a limit, which exceeds the inherent variation in the genuine signatures of a signer. Different categories of forgery arise depending on what limit of variation we allow over the inherent variation. The various phases of the verification and detection are briefly discussed in the following.

4.1. Data acquisition

The signatures were handwritten on a white sheet of paper, using a black pen. The signature images were then scanned at a resolution of 200 dpi and resampled/resized by 50% using a B-Spline filter in IrfanView. IrfanView is a very fast, small, compact and innovative FREEWARE graphic viewer for Windows 9x/ME/NT/2000/XP. Some typical signatures along with their forgeries are given in Fig. 2. Some signatures were also extracted from bank checks as shown in Fig. 3. The extraction of a signature from a bank check is in itself a very difficult task [16] as the check backgrounds are complex in nature. For this reason, we have considered bank checks with a uniform background and done a simple histogram analysis to determine the global threshold that separates the signature from its background [17,18].

The original scanned signatures are pre-processed involving size normalization, binarization and thinning before features are extracted from each of them. These features constitute the knowledge base, which is then used for

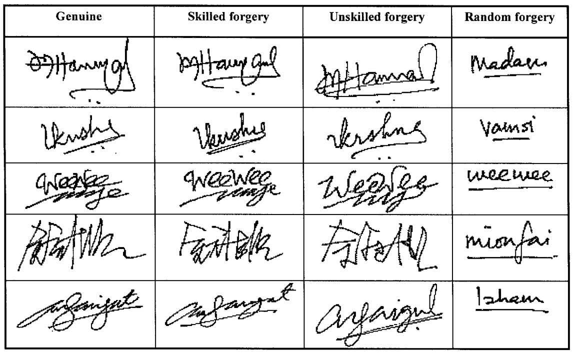

Fig. 2. Examples of different types of forgeries of some typical signatures.

verifying the genuine signatures and detecting forgeries. We now briefly explain the various stages in the signature verification system.

4.2. Pre-processing

Pre-processing of scanned signatures/signature images is necessary before feature extraction. In this system, all the signature images are first resized to a fixed window of size (120×60 pixels), then binarized and thinned using the modified SPTA algorithm [2]. Features are then extracted from this pre-processed signature image.

4.3. Feature extraction

The pre-processed image is then partitioned into eight portions using the equal horizontal density method. In this method, the binarized image is scanned horizontally from left to right and then from right to left and the total number of dark pixels is obtained over the entire image. The pixels are clustered into eight regions such that approximately equal number of dark pixels falls in each region. This process is illustrated in Fig. 4a and explained in the following.

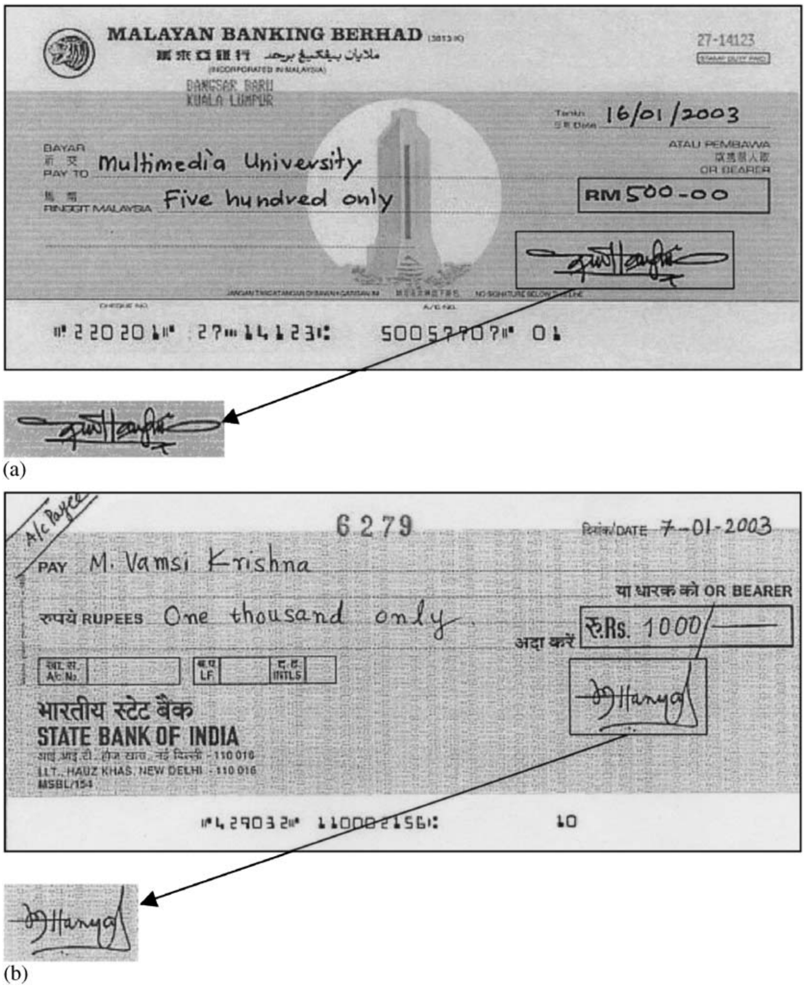

Fig. 3. (a and b) Extraction of signatures from two bank checks.

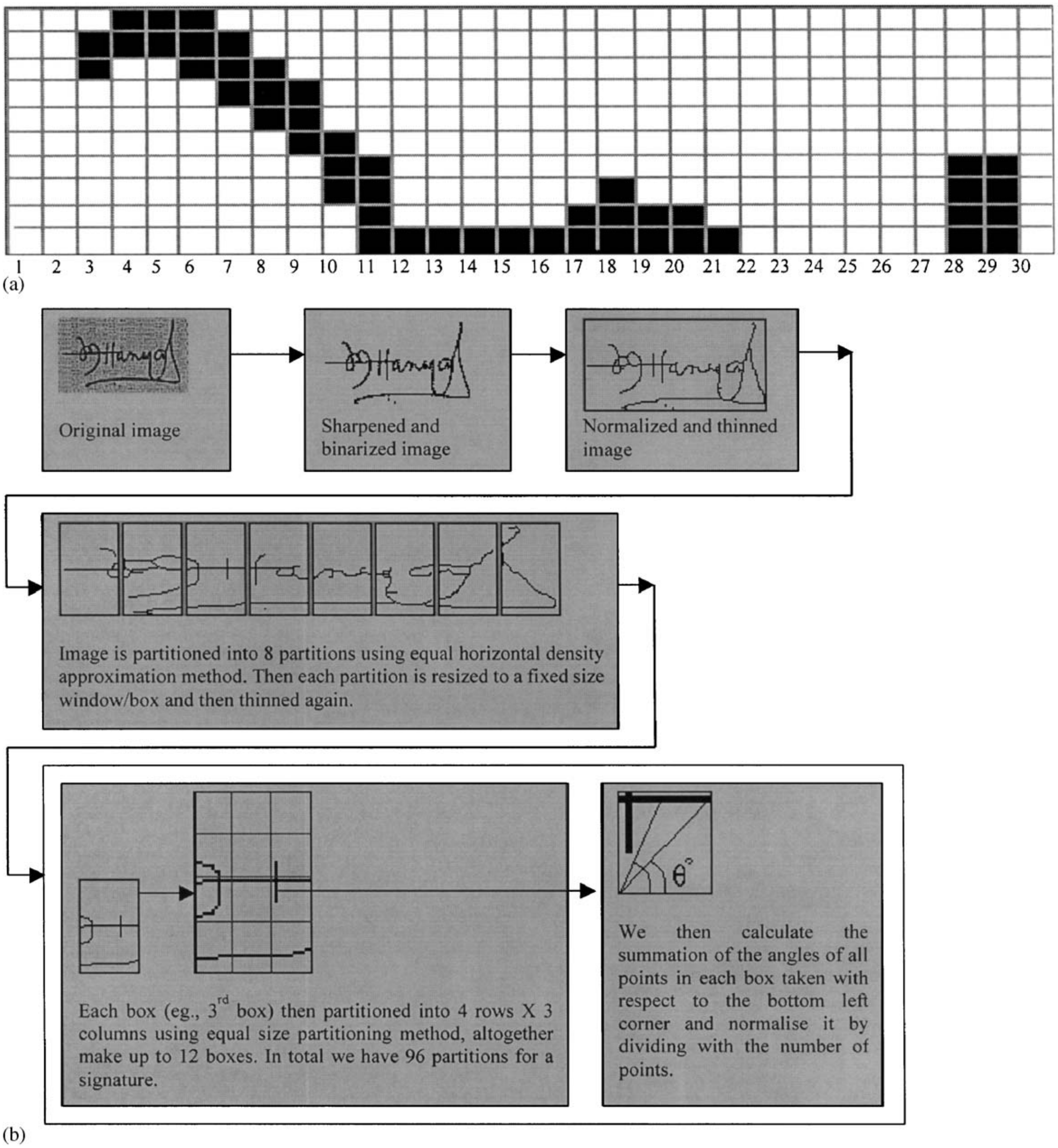

Fig. 4. (a) Partition using horizontal density. (b) Preprocessing and feature extraction.

Here, the total number of points (dark pixels) is 48. If we partition these pixels into 4, we get $48/4 = 12$ pixels per partition. Since the partition is done column wise, getting exactly 12 points in each partition is quite impossible. Therefore, we have taken approximately 12 points in each partition using two-way scanning approach as described below. We scan the image from left to right till we reach the column where the number of points in a particular partition is 12 or more. We then repeat the same procedure while scanning the image from right to left direction. For left to right, we partition the image at column numbers 1–7 (12 points), 8–11 (13 points), 12–19 (12 points) and 20–30 (11 points). For right to left we partition the image at column numbers 30–19 (13 points), 18–11 (14 points), 10–7 (12 points) and 6–1 (9 points). We then take the average of two column numbers in each partition as given in Table 4.

Each partition is now resized to a fixed window (box of 38×60 pixels) size and is thinned again. Each box is again partitioned into 4 rows $\times$ 3 columns, constituting 12 boxes. In total we have 96 partitions for a single signature. Since

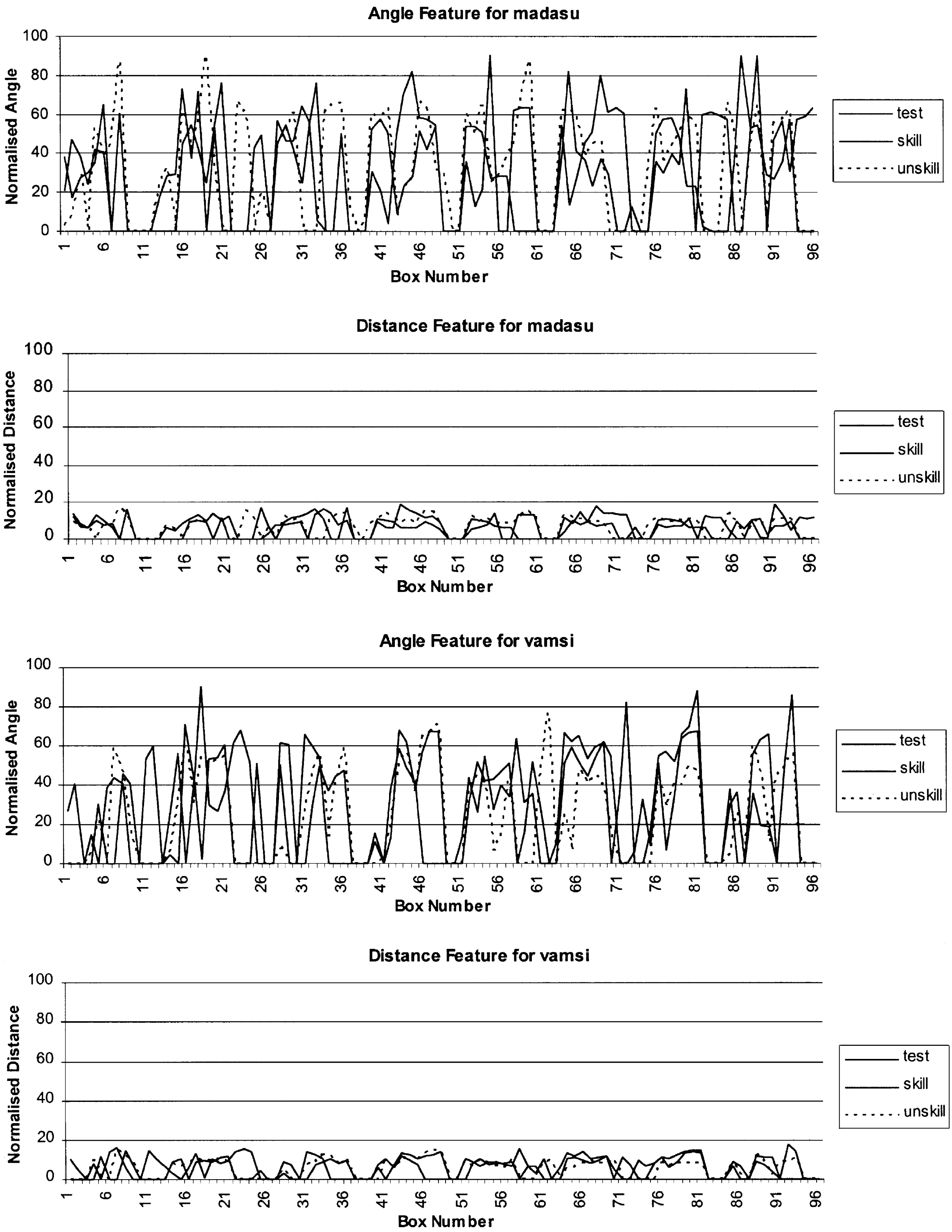

Fig. 5. Distance and angle distributions.

the angle distribution shows more variation than the distance distribution [1] as depicted in Fig. 5 we have considered only the angles in the present work. We calculate the summation of the angles of all points in each box with respect to the bottom left corner of the box. The summation of angles is normalized with the number of pixels in the box. These angles constitute the feature database for a given signature. The feature extraction process is illustrated in Fig. 4b.

4.4. Verification using TS model with structural parameters

Since the main thrust here is to establish the genuineness of the signature thereby detecting the forgeries, we go in for fuzzy modeling of angle features. For the purpose of signature verification and detection of forgeries, we have employed the TS model. In this, we are following the same concept as outlined in Ref. [19] for considering each feature as forming a fuzzy set over large samples. This is because the same feature exhibits variation in different samples giving rise to a fuzzy set. So, our attempt is to model the uncertainty through a fuzzy model such as the TS model.

The First Formulation: Let x_k be the kth feature in a fuzzy set A_k, so the kth IF THEN fuzzy rule in TS model has the following form:

$$\begin{aligned} &\textit{Rule } k: \text{IF } x_k \text{ is } A_k \\ &\qquad \text{THEN } y_k = c_{k0} + c_{k1} x_k. \end{aligned} \tag{1}$$

Each signature will have a rule so we have as many rules as the number of features. The fuzzy set A_k is represented by the following exponential membership function (MF) that includes two structural parameters s_k and t_k:

$$\mu_k(x_k) = \exp - \left[\frac{(1-s_k) + s_k^2 |x_k - \overline{x_k}|}{(1+t_k) + t_k^2 \sigma_k^2} \right], \tag{2}$$

where $\overline{x_k}$ is the mean σ_k^2 is the variance of kth fuzzy set. Note that the inclusion of these parameters will help track the variations in the handwriting of signatures. When $s_k = 1$ and $t_k = -1$, the MF is devoid of structural parameters and hence it is solely governed by the mean and variance. The justification for the modified MF is two-fold: (i) Easy to track variations over mean and variance, and (ii) no need of sophisticated learning technique. The numerator and denominator of exponential function in Eq. (2) contain a constant term (i.e., 1) plus a function of parameter and the known variation (i.e., either change in mean or in variance). This choice is guided by the consideration of no role for parameters if the signatures of a person do not change. But this need not be the case for other applications.

The strength of the rule in Eq. (1) is obtained as

$$w_k = \mu_k(x_k). \tag{3}$$

The output is expressed as

$$Y = \sum_{k=1}^{L} w_k y_k, \tag{4}$$

where L is the number of rules.

We define the performance function as

$$J = (Y_r - Y)^2, \tag{5}$$

where, Y and Y_r denote the output of the fuzzy model and of the real system, respectively. If Y_r is not available, it can be assumed to be unity.

In order to learn the parameters involved in the membership function (i.e., s_k and t_k) and the consequent parameters c_{k0} and c_{k1}, Eq. (5) is partially differentiated with respect to each of these parameters. Accordingly, we have

$$\frac{\partial J}{\partial c_{k1}} = \frac{\partial J}{\partial Y} \cdot \frac{\partial Y}{\partial y_k} \cdot \frac{\partial y_k}{\partial c_{k1}} = 2(Y - Y_r) w_k x_k, \tag{6}$$

$$\frac{\partial J}{\partial c_{k0}} = \frac{\partial J}{\partial Y} \cdot \frac{\partial Y}{\partial y_k} \cdot \frac{\partial y_k}{\partial c_{k0}} = 2[Y - Y_r] w_k = 2\delta w_k, \tag{7}$$

$$\begin{aligned} \frac{\partial J}{\partial s_k} &= \frac{\partial J}{\partial Y} \cdot \frac{\partial Y}{\partial w_k} \cdot \frac{\partial w_k}{\partial t_k} \\ &= 2(Y - Y_r) \cdot y_k \cdot \frac{\mu_k \{1 - 2s_k |x_k - \overline{x_k}|\}}{\{(1+t_k) + t_k^2 \sigma_k^2\}} \\ &= 2\delta y_k \mu_k [\{1 - 2s_k |x_k - \overline{x_k}|\}/T], \end{aligned} \tag{8}$$

$$\begin{aligned} \frac{\partial J}{\partial t_k} &= \frac{\partial J}{\partial Y} \cdot \frac{\partial Y}{\partial w_k} \cdot \frac{\partial w_k}{\partial t_k} \\ &= 2(Y - Y_r) y_k \mu_k \frac{\{(1-s_k) + s_k^2 |x_k - \overline{x_k}|\}\{1 + 2t_k \sigma_k^2\}}{\{(1+t_k) + t_k^2 \sigma_k^2\}^2} \\ &= 2\delta y_k \mu_k \{(1-s_k) + s_k^2 |x_k - \overline{x_k}|\}\{1 + 2t_k \sigma_k^2\}/T^2, \end{aligned} \tag{9}$$

where $\delta = Y - Y_r$, $T = (1+t_k) + t_k^2 \sigma_k^2$ and $k = 1, \ldots, L$ denotes the rule number.

We use the gradient descent learning for the parameters as follows:

$$c_{ki}^{\text{new}} = c_{ki}^{\text{old}} - \epsilon_1 \frac{\partial J}{\partial c_{ki}} \quad i = 0, 1, \tag{10}$$

$$s_k^{\text{new}} = s_k^{\text{old}} - \epsilon_2 \frac{\partial J}{\partial s_k}, \tag{11}$$

$$t_k^{\text{new}} = t_k^{\text{old}} - \epsilon_3 \frac{\partial J}{\partial t_k}, \tag{12}$$

where $\epsilon_1, \epsilon_2, \epsilon_3$ are the learning coefficients such that ϵ_1, ϵ_2 and $\epsilon_3 > 0$.

4.5. Global gradient descent learning of parameters

We can go for global learning when we have large sets of data, say, M. This is known as the batch learning scheme, in which change in any parameter is governed by the equation [20]:

$$\Delta w(q) = \sum_{j=1}^{M} \varDelta_j w(q) + \alpha_m \Delta w(q-1) - \gamma w(q) \tag{13}$$

and the parametric update equation is;

$$w(q+1) = w(q) + \Delta w(q) \tag{14}$$

where w in Eq. (13) may stand for any of the parameters c_{ki}, s_k, t_k and q is the qth epoch, α_m is a momentum coefficient in the limits $0 \leqslant \alpha_m < 1$ (typically $\alpha_m = 0.9$), γ is a decay factor (typically in the range of 10^{-3}–10^{-6}). We can obtain initial $\Delta w(q)$ from Eqs. (10) to (12) by computing the partial derivatives of J.

We will now show that the recognition approach of Ref. [2] is a special case of TS model. For this, assume $c_{k0} = 1/L$ and $c_{k1} = 0$ so that $y_k = 1/L$ in Eq. (1). Substituting this in Eq. (4) yields

$$Y = \frac{1}{L} \sum_{i=1}^{L} \mu_i . \tag{15}$$

In Eq. (15) Y is given by the average of the membership functions (MFs) and this average is used in [2] for identifying the unknown character. It is now proved that the average MF is a special case of TS model. The recursive Eqs. (10)–(12) have to be iterated until the summation of δ for all feature values is small enough. The initial values of the structural parameters are obtained from:

$$\frac{\partial J}{\partial s_k} = 0 \Rightarrow 1 - 2s_k|x_k - \overline{x_k}| = 0$$
$$\Rightarrow S_k = \frac{1}{2|x_k - \bar{x}_k|}, \tag{16}$$

$$\frac{\partial J}{\partial t_k} = 0 \Rightarrow 1 + 2t_k\sigma_k^2 = 0 \quad \Rightarrow \quad t_k = -\frac{1}{2\sigma_k^2}. \tag{17}$$

Note that the above initial values do not yield satisfactory results. We have to tune these values to come up with an efficient set of values.

The Second Formulation: Alternatively, it is possible to use only a single rule for all input features. The corresponding TS model will have the fuzzy rule of the form

$$\textit{Rule}: \text{IF } x_1 \text{ is } A_1, x_2 \text{ is } A_2, \ldots, x_n \text{ is } A_n$$
$$\text{THEN } y = c_0 + \sum_{i=1}^{n} c_i x_i . \tag{18}$$

The performance function now becomes

$$J = \{Y_r - wy\}^2 \text{ with } w = \prod_{j=1}^{n} \mu_j . \tag{19}$$

The derivatives of J with respect to c_0, c_i, s_i, t_i are given by the following equations:

$$\frac{\partial J}{\partial c_0} = -2\{Y_r - wy\} \prod_{j=1}^{n} \mu_j , \tag{20}$$

$$\frac{\partial J}{\partial c_i} = -2\{Y_r - wy\} x_i \prod_{j=1}^{n} \mu_j , \tag{21}$$

$$\frac{\partial J}{\partial s_i} = -2\{Y_r - wy\} y\mu_i \frac{\{1 - 2s_i(x_i - \overline{x_i})\}}{\{(1+t_i) + t_i^2\sigma_i^2\}}, \tag{22}$$

$$\frac{\partial J}{\partial t_i} = -2\{Y_r - wy\}$$
$$\times y\mu_i \frac{\{(1-s_i) + s_i^2(x_i - \overline{x_i})\}\{1 + 2t_i\sigma_i^2\}}{\{(1+t_i) + t_i^2\sigma_i^2\}^2}. \tag{23}$$

The parameters can be found by the gradient descent technique. We will now derive the simplified version of the performance function. For this, assume $c_0 = 1$ and $\forall c_i = 0$ in Eq. (18). This leads to

$$y = wy = w = \prod_{j=1}^{n} \mu_j . \tag{24}$$

From Eqs. (15) and (24), we observe that if we have a rule for each input feature, the simplified performance function is given by the average MF. On the other hand, if all input features are linked by a single rule, the simplified performance function corresponds to the multiplication of all MFs. We find that Eq. (24) is more stringent than Eq. (15) as it requires that all membership values must be nonzero. The recognition using Eq. (15) is bound to be better in view of a large number of rules and parameters involved. So, our implementation follows this recognition strategy but by making subtle changes to suit the real world problems.

5. Results of implementation

The proposed fuzzy modeling based on TS model discussed above has been applied on a signature database, developed in the Graphics Visualization and Games Development (GVGD) lab at the Multimedia University, Cyberjaya, Malaysia. We will consider two cases: In the first case, we use the simplified TS model in which the coefficients of the THEN part (Consequent) are fixed. In the second case we adapt the coefficients. The details of the signature database and the experimental results are discussed in the following:

5.1. Signature database

In the field of off-line signature verification, no standard international database is available due to the confidentiality

Table 1
Number of samples for each person

	DATABASE		
	Training set	Testing set	TOTAL
Genuine	40 × 10	40 × 5	600
Skilled forgery		40 × 5	200
Unskilled forgery		40 × 5	200
Random forgery		40 × 5	200

Table 3
Horizontal density approach for partition

Scan direction	Partition No			
	1 Column no.	2 Column no.	3 Column no.	4 Column no.
Left to right	7	11	19	30
Right to left	6	10	18	30
Average	6.5 = > 7	10.5 = > 11	18.5 = > 19	30

of this type of data. The proposed signature verification system was trained and tested on a database consisting of a total of 1200 handwritten signature images. Out of these, 600 were authentic signatures and 600 were forged ones. These signatures were obtained from 40 volunteers with each person contributing to 15 signatures as shown in Table 1. The signatures were collected over a period of a few weeks to account for variations in the signature with time. The forgeries of these signatures were collected over a similar time frame. The random forgeries were obtained by supplying only the names of the individuals to the casual forgers who did not have any access to the actual genuine signatures. The unskilled forgeries in turn, were obtained by providing sample genuine signatures to the forgers who were then allowed to practice for a while before imitating them to create their forgeries. Each volunteer had to provide five imitations of any one of the genuine signatures, apart from his or her own signatures. These samples constituted the set of unskilled forged signatures for the set of genuine signatures. We then requisitioned the services of six expert forgers who provided five forgeries of each genuine signature in the test set to create the skilled forged samples of all the persons.

5.2. Experimental results

It is a well known fact that any automatic signature verification system requires a very small training set of signatures. For this reason, we have set the number of training signatures for each individual at ten. The angle features for our study are shown in Table 2. The partition of features is shown in Table 3.

Case 1: TS model with consequent coefficients fixed.

In view of Eq. (15) and taking $Y_r = 1$, then Eq. (5) becomes

$$J = \left(1 - \frac{1}{L}\sum_{i=1}^{L} \mu_i\right)^2. \tag{25}$$

With the above performance index, we compute $\frac{\partial J}{\partial s_i}$ and $\frac{\partial J}{\partial t_i}$ in order to update the structural parameters s_i and t_i; $i = 1, \ldots, 96$. Using these values, we compute the membership functions for all the features. This process is repeated for all training samples of a person. Here, we have devised an innovative approach for the classification of all signatures (i.e., test signatures and random, skilled and unskilled forgeries) of a person.

Innovative approach using variation in MF: In order to know the extent of variation in the genuine signatures, we determine the maximum and minimum membership functions for each feature over all signatures in the training set. The difference between these two gives the inherent variation in the signatures of a person. We add some tolerance to the maximum and delete the same from the minimum so as to increase the range of variation in the different signatures. This tolerance is meant for possible increase in the inherent variation over a time.

We now use the inherent variation to judge the test signatures. We will also explain its utility in the testing phase. For a particular feature, if the membership value lies within the range of variation which is given by the difference of minimum and maximum thresholds, it is counted as ‘true’.

Table 2
Angle features of one of the signatures used for training

Cluster	Partition											
	1	2	3	4	5	6	7	8	9	10	11	12
1	21.1	46.8	38.1	23.9	41.6	40.1	0	0	0	0	0	0
2	0	0	0	44.8	54.8	41.7	24.7	54.5	76.1	0	0	0
3	0	0	40.1	46.9	39.8	0	72.4	60.2	21.0	46.1	56.4	0
4	0	0	30.5	20.5	3.7	49.4	70.2	81.9	58.0	57.5	54.2	0
5	0	0	35.6	12.7	21.3	90	0	0	61.6	63.3	63.3	0
6	0	0	54.7	13.6	26.9	45.6	50.6	79.9	60.9	63.3	60.5	0
7	0	0	36.1	30.1	39.4	33.9	73.4	0	59.6	61.3	59.6	57.4
8	0	0	52.1	90	0	47.1	56.6	30.7	57.4	59.1	63.3	0

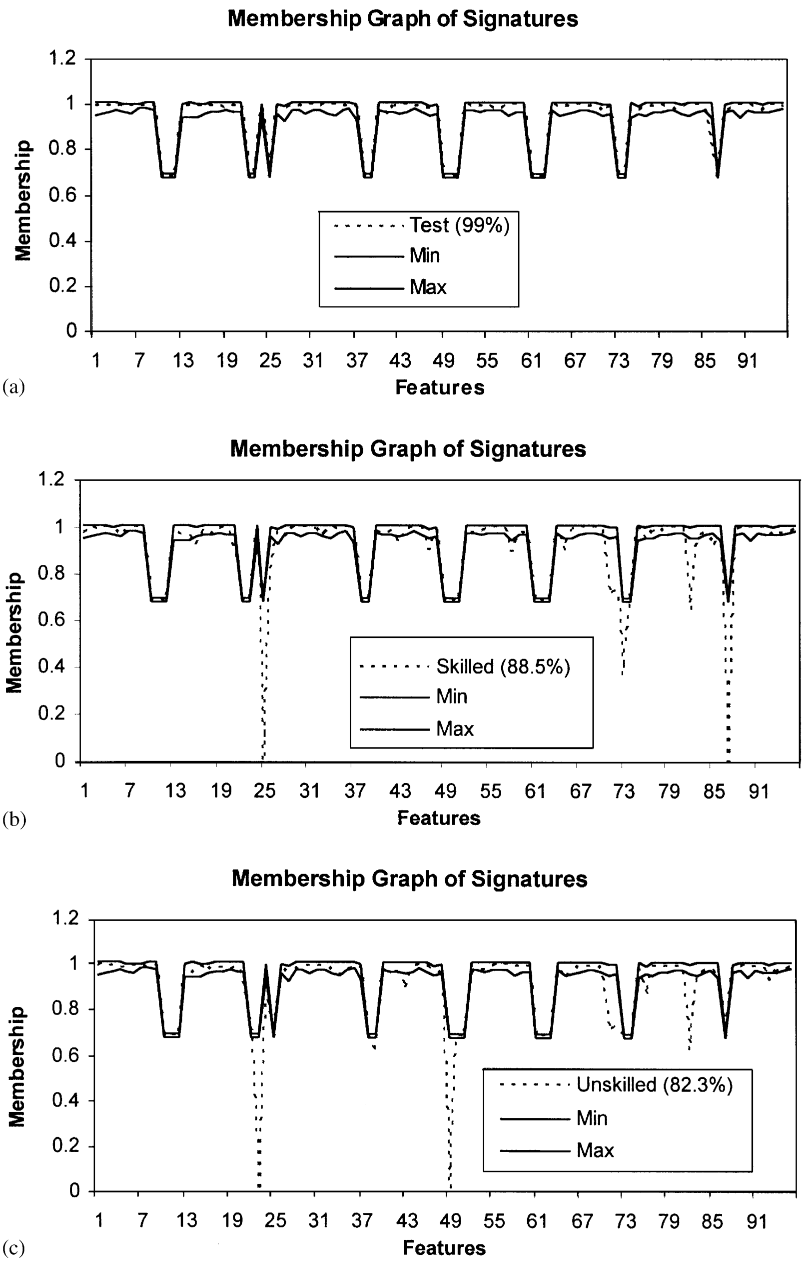

Fig. 6. Membership graph of signatures: (a) Classified as Genuine signature as it is above 91% (refer to Test signature no. 3 in Fig. 8a). (b) Classified as Skilled forgery as it is below 91% (refer to Skill-forged signature no. 5 in Fig. 8a). (c) Classified as Unskilled forgery as it is below 91% (refer to Unskill-forged signature no. 1 in Fig. 8a).

The total number of 'true' cases for a particular signature is divided by the total number of features (i.e., 96) to get the percentage. For example, in Fig. 6a, the test signature has 99% of its features lying well within the threshold as can be seen from the membership function (i.e., 95 out of 96 features are within the range of inherent variation). The skill- and unskill-forged signatures have corresponding figures of 88.5% (Fig. 6b) and 82.3% (Fig. 6c), respectively. We set the minimum limit or acceptable percentage for genuine signature at 91% referring to the output result of signature of a signer, one of the authors nicknamed *madasu*. Signatures that have percentage less than 91% are treated as forged signatures. Table 4 gives the initial values of learning and structure parameters. Table 5a summa-

Table 4
Initial values of the structural and learning parameters

Parameter	Simplified TS model Initial values	TS model Initial values
s	0.1	1
t	1.4	2
c_0	1/96	1/96
c_1	0	0
ε_1	—	0.00000001
ε_2	0.01	0.01
ε_3	0.01	0.01
Precision	0.01	0.01

Table 5
Results using TS model

	Total	Accepted	Rejected
(a) *Consequent coefficients fixed* (*Formulation* 1)			
Genuine	200	200 (100%)	0 (0%)
Skilled forgery	200	0 (0%)	200 (100%)
Unskilled forgery	200	0 (0%)	200 (100%)
Random forgery	200	0 (0%)	200 (100%)
(b) *Consequent coefficients fixed* & *Average J* (*Formulation* 1)			
Genuine	200	184 (92%)	16 (8%)
Skilled forgery	200	44 (22%)	156 (78%)
Unskilled forgery	200	8 (4%)	192 (96%)
Random forgery	200	0 (0%)	200 (100%)
(c) *Consequent coefficients fixed* & *Max. J* (*Formulation* 1)			
Genuine	200	200 (100%)	0 (0%)
Skilled forgery	200	42 (21%)	158 (79%)
Unskilled forgery	200	6 (3%)	194 (97%)
Random forgery	200	0 (0%)	200 (100%)

rizes the results of forgery detection using this innovative approach.

Intuitive approaches using the average and max J: Next, we have used the performance index (5) and its derivatives to adapt the structural parameters during the training phase. These are used to determine the extent of inherent variation in terms of J in the training phase. We have tried two intuitive approaches. In the first case we have taken average J and in the second case we have taken maximum J, both serving as thresholds. The samples in the testing phase are judged by comparing their J values against the thresholds. Tables 5(b) and (c) provide the results of forgery detection. Comparing these results with those of Table 5(a), we find that the innovative approach yields the best performance.

Case 2: TS model with adaptive consequent coefficients

Next, we have used the performance index (5) and its derivatives to adapt the both consequent coefficients and the structural parameters during the training phase. As mentioned above, we have used both the average and maximum

Table 6
Results using TS model

	Total	Accepted	Rejected
(a) *Coefficients adapted* & *Average J* (*Formulation* 1)			
Genuine	200	172 (86.0%)	28 (14%)
Skilled forgery	200	47 (23.5%)	153 (76.5%)
Unskilled forgery	200	8 (4%)	192 (96.0%)
Random forgery	200	0 (0%)	200 (100%)
(b) *Coefficients adapted* & *Max. J* (*Formulation* 1)			
Genuine	200	200 (100.0%)	0 (0%)
Skilled forgery	200	44 (22%)	156 (78%)
Unskilled forgery	200	6 (3%)	194 (97.0%)
Random forgery	200	0 (0%)	200 (100%)

values of J for the detection of forgeries. Tables 6(a) and (b) show results using these two thresholds.

The features, structural parameters for two typical signatures, i.e., *madasu* and *vamsi*, are given in Tables 7 and 8, respectively. Fig. 7 shows the sample output of the program.

For the second formulation involving a single TS rule, the results with consequents fixed are shown in Table 9 and results with coefficients adapted are shown in Table 10. As compared to the results of the first formulation, these results are not promising for the reasons cited above. Here, we have not made use of the average and maximum values of J.

All the experiments have been carried on a Pentium III, 1.1 GHz Celeron processor having 256 MB SDRAM with Windows XP operating system. With this configuration, the system takes about 19 s to train 10 signature images while it takes about 2 s to test one signature. Surprisingly, only a single iteration is required to achieve the convergence as the learning parameters and initial structure parameters have been selected optimally.

5.3. *Results of a typical case*

To test the efficacy of the proposed signature verification system, we subjected it to a typical test. The current signatures of a signer, i.e., *madasu*, who had changed his signature a few years ago, have been used to train the parameters and thresholds for testing the old signatures. As the old signatures have a slight change at their ends, the verification system declared the old signatures as forged. The sample outputs for the typical case are shown in Fig. 8. This test demonstrates the capability of the system in detecting even the minutest changes in the signature samples.

6. Conclusions

An off-line signature verification and forgery detection system is modeled by TS model, which involves structural parameters in its exponential membership function. The

Table 7

Cluster	Partition											
	1	2	3	4	5	6	7	8	9	10	11	12
(a) *Features of a trained signature* (*Madasu*)												
1	21.1	46.8	38.1	23.9	41.6	40.1	0	0	0	0	0	0
2	0	0	0	44.8	54.8	41.7	24.7	54.5	76.1	0	0	0
3	0	0	40.1	46.9	39.8	0	72.4	60.2	21.0	46.1	56.4	0
4	0	0	30.5	20.5	3.7	49.4	70.2	81.9	58.0	57.5	54.2	0
5	0	0	35.6	12.7	21.3	90	0	0	61.6	63.3	63.3	0
6	0	0	54.7	13.6	26.9	45.6	50.6	79.9	60.9	63.3	60.5	0
7	0	0	36.1	30.1	39.4	33.9	73.4	0	59.6	61.3	59.6	57.4
8	0	0	52.1	90	0	47.1	56.6	30.7	57.4	59.1	63.3	0
(b) *Parameter S* (*Madasu*)												
1	0.100523	0.100618	0.100621	0.100581	0.100226	0.100256	0.100698	0.10075	0.10076	0.09998	0.09998	0.09998
2	0.100412	0.10046	0.100436	0.10023	0.100368	0.100492	0.100734	0.100624	0.100479	0.09998	0.09998	0.100392
3	0.09998	0.100263	0.10047	0.100406	0.100626	0.100601	0.10074	0.100735	0.100683	0.100607	0.100696	0.100721
4	0.100365	0.09998	0.09998	0.100569	0.100606	0.100577	0.100444	0.100705	0.100749	0.100417	0.100089	0.100256
5	0.09998	0.09998	0.09998	0.100607	0.100532	0.100555	0.1007	0.100719	0.100721	0.100423	0.10069	0.100703
6	0.09998	0.09998	0.09998	0.10063	0.100585	0.100408	0.100603	0.100619	0.100588	0.10068	0.100156	0.100155
7	0.09998	0.09998	0.100517	0.100134	0.100374	0.100572	0.100638	0.100705	0.100681	0.100157	0.100641	0.100699
8	0.100473	0.100544	0.09998	0.100307	0.100693	0.100689	0.100683	0.100505	0.100149	0.100703	0.100735	0.100764
(c) *Parameter T* (*Madasu*)												
1	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.39953	1.39908	1.39911
2	1.38695	1.399945	1.391186	1.4	1.4	1.4	1.4	1.4	1.4	1.39934	1.39932	1.39989
3	1.38929	1.399922	1.399964	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4
4	1.39922	1.39908	1.39972	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4
5	1.39461	1.39921	1.39989	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4
6	1.39558	1.39567	1.39945	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4
7	1.39115	1.39229	1.39939	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4
8	1.4	1.399631	1.39906	1.4	1.4	1.399985	1.4	1.4	1.4	1.4	1.4	1.4

Table 8

Cluster	Partition											
	1	2	3	4	5	6	7	8	9	10	11	12
(a) *Features of trained signature* (*Vamsi*)												
1	0	0	0	14.965	0	38.722	43.532	40.98	0	0	53.395	59.758
2	0.003	4.217	0	70.753	41.751	90	29.9	26.763	36.933	61.228	67.941	51.955
3	0	0	0	50.431	0	0	0	30.674	49.371	37.092	44.078	47.229
4	0	0	0	11.412	0	13.514	68.505	62.43	38.13	57.084	67.561	67.379
5	0	0	13.676	43.929	26.237	55.107	27.633	40.245	34.333	63.712	30.749	35.793
6	0	0	0	51.225	59.57	51.501	44.163	53.358	61.855	55.174	0	0
7	6.173	32.505	8.906	51.428	6.552	35.435	66.243	69.402	87.94	0	0	0
8	29.745	35.961	0	55.425	62.666	65.889	0	0	0	0	0	0
(b) *Parameter S* (*Vamsi*)												
1	0.099999	0.100239	0.100479	0.100443	0.100428	0.100467	0.100485	0.100334	0.100501	0.09998	0.100482	0.100466
2	0.100376	0.100436	0.100472	0.100432	0.100478	0.100518	0.100377	0.100133	0.100312	0.100487	0.100507	0.10052
3	0.100444	0.09998	0.09998	0.100481	0.100468	0.100491	0.10042	0.100426	0.100298	0.100511	0.100413	0.100347
4	0.09998	0.09998	0.09998	0.100192	0.100478	0.10048	0.100185	0.10025	0.100413	0.10023	0.100155	0.100417
5	0.09998	0.100286	0.100383	0.100369	0.100325	0.100384	0.100395	0.100338	0.100394	0.100471	0.100348	0.100507
6	0.100506	0.100493	0.100455	0.100332	0.100252	0.100115	0.100387	0.100308	0.10024	0.100497	0.100522	0.100502
7	0.100292	0.100429	0.100285	0.100303	0.10042	0.100364	0.100221	0.100421	0.100448	0.100482	0.100524	0.10052
8	0.100458	0.100449	0.09999	0.100289	0.100334	0.100342	0.100526	0.100514	0.09998	0.09998	0.09998	0.09998
(c) *Parameter T* (*Vamsi*)												
1	1.39688	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.39929	1.4	1.4
2	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4
3	1.4	1.39929	1.39972	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4
4	1.39922	1.39951	1.39965	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4
5	1.39952	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4
6	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4
7	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4
8	1.4	1.4	1.39927	1.4	1.4	1.4	1.4	1.4	1.39999	1.39976	1.39338	1.39727

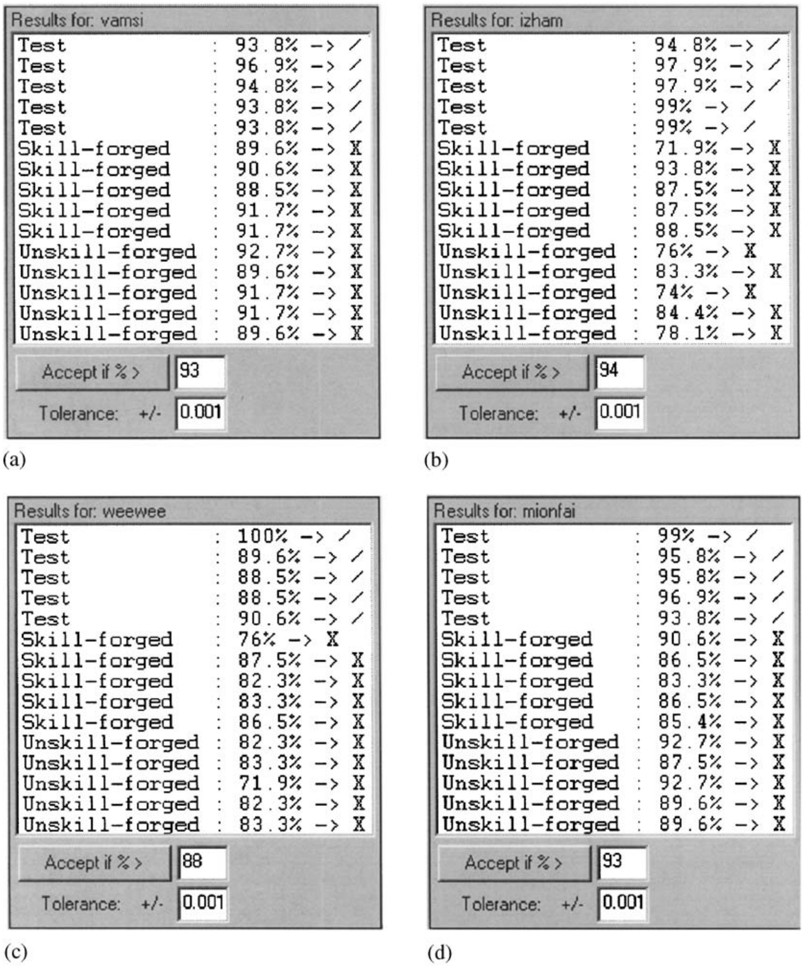

Fig. 7. Sample outputs from the program for the signer; (a) Vamsi. (b) Izham. (c) Weewee. (d) Mionfai.

Table 9
Results using TS model with consequent coefficients fixed (Formulation 2)

	Total	Accepted	Rejected
Genuine	200	125 (62.5%)	75 (37.5%)
Skilled forgery	200	68 (34%)	132 (66%)
Unskilled forgery	200	51 (25.5%)	149 (74.5%)
Random forgery	200	50 (25%)	150 (75%)

Table 10
Results using TS model with adapted consequent coefficients (Formulation 2)

	Total	Accepted	Rejected
Genuine	200	107 (53.5%)	93 (46.5%)
Skilled forgery	200	84 (42%)	116 (58%)
Unskilled forgery	200	68 (34%)	132 (66%)
Random forgery	200	45 (22.5%)	155 (77.5%)

features consisting of angles are extracted using box approach. Each feature yields a fuzzy set when its values are gathered from all samples because of the variations in handwritten signatures. Two cases are considered. In the first case, the coefficients of the consequent part of the rule are fixed so as to yield a simple form of TS model and in the second case the coefficients are adapted. In this first formulation (Multiple rules), each rule is constituted by a single feature. In the second formulation, we consider only a single rule encompassing all the features. Here again, we have considered two cases depending on whether coefficients of the consequent part are fixed or adapted.

The efficacy of this system has been tested on a large database of signatures using two formulations. The

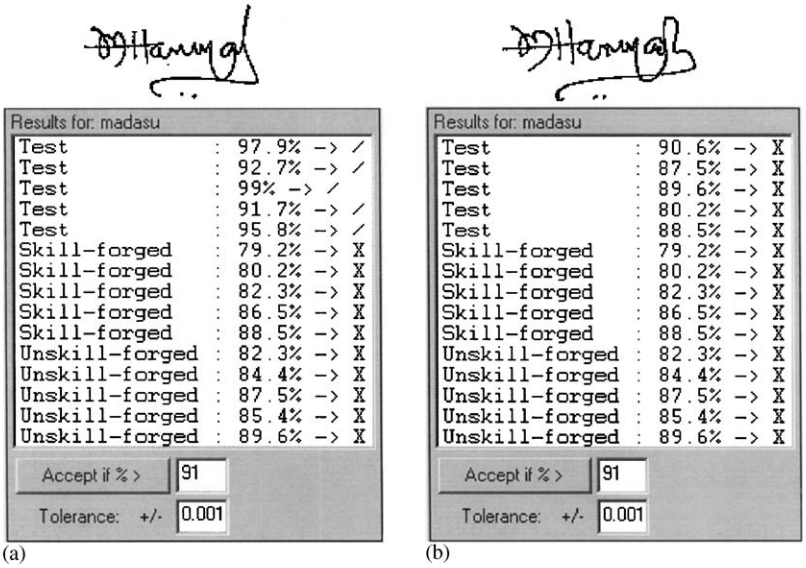

Fig. 8. Sample outputs from the program for the typical case showing that the old signatures are treated as: (a) Current signature of the signer and (b) Old signature of the same signer.

verification system using the first formulation is able to detect all types of forgeries: random, unskilled and skilled with utmost precision. In this formulation, the TS model with fixed consequent coefficients is found to be better than that with consequent coefficients adapted. We have also demonstrated the effectiveness of an innovative approach using variation in the membership function and of two intuitive approaches for signature verification by incorporating the average and maximum values of performance index J in the decision making. The results using the second formulation with both fixed and adapted consequent coefficients are not promising, so we did not try the average and maximum values of J.

The choice of initial parameters is important but not crucial. But, we need to make a proper choice only once and it is applicable to all types of signatures. We have not used global learning techniques for want of simplicity at the implementation stage.

References

[1] M. Hanmandlu, K.R. Murali Mohan, S. Chakraborty, G. Garg, Fuzzy modeling based signature verification system, in: Proceedings of the Sixth International Conference on Document Analysis and Recognition, USA, 2001, pp. 110–114.

[2] M. Hanmandlu, K.R. Murali Mohan, S. Chakraborty, S. Goel, D. Roy Choudhury, Unconstrained handwritten character recognition based on fuzzy logic, Pattern Recognition 36 (3) (2003) 603–623.

[3] R. Plamondon, The design of an on-line signature verification system: from theory to practice, Int. J. Pattern Recogn. Artif. Intell. 8 (1994) 795–811.

[4] M. Ammar, Y. Yoshida, T. Fukumura, A new effective approach for off-line verification of signatures by using pressure features, in: Proceedings of the International Conference on Pattern Recognition, 1986, pp. 566–569.

[5] R. Plamondon, F. Leclerc, Automatic signature verification: the state of the art 1989–1993, Int. J. Pattern Recogn. Artif. Intell. 8 (3) (1994) 643–660.

[6] R. Plamondon, G. Lorette, Automatic signature verification and writer identification: the state of the art, Pattern Recognition 22 (2) (1989) 107–131.

[7] R. Sabourin, R. Plamondon, G. Lorette, Off-line identification with handwritten signature images: survey and perspectives, Structured Image Analysis, Springer, New York, 1992, pp. 219–234.

[8] R. Plamondon, S.N. Srihari, On-line and off-line handwriting recognition: a comprehensive survey, IEEE Trans. Pattern Anal. Mach. Intell. 22 (1) (2000) 63–84.

[9] M.A. Ismail, Samia Gad, Off-line Arabic signature recognition and verification, Pattern Recognition 33 (10) (2000) 1727–1740.

[10] A. El-Yacoubi, E.J.R. Justino, R. Sabourin, F. Bortolozzi, Off-line signature verification using HMMS and cross-validation, in: Proceedings of the IEEE Workshop on Neural Networks for Signal Processing, USA, 2000, pp. 859–868.

[11] Y. Xuhua, T. Furuhashi, K. Obata, Y. Uchikawa, Study on signature verification using a new approach to genetic based machine learning, in: Proceedings of the IEEE International Conference on Systems, Man and Cybernetics USA, 1995, pp. 4383–4386.

[12] C. Quek, R.W. Zhou, Antiforgery: a novel pseudo-outer product based fuzzy neural network driven signature verification system, Pattern Recogn. Lett. 23 (2002) 1795–1816.

[13] A.S. Osborn, Questioned Documents, Patterson Smith Publication Corporation, 1973.

[14] Jinhong K. Guo, D. Doermann, A. Rosenfeld, Off-line skilled forgery detection using stroke and sub-stroke properties, in: Proceedings of the International Conference on Pattern Recognition, 2000, pp. 355–358.
[15] D.A. Pender, Neural Networks and handwritten signature verification, Ph.D. dissertation, Stanford University, 1991.
[16] S. Djeziri, F. Nouboud, R. Plamondon, Extraction of signatures from check background based on a filiformity criterion, IEEE Trans. Image Process. 7 (10) (1998) 1425–1438.
[17] R. Plamondon, G. Lorette, R. Sabourin, Automatic processing of signature images: Static techniques and methods, in: R. Plamondon, C.G. Leedham (Eds.), Computer Processing of Handwriting, 1990, pp. 49–63.
[18] R. Sabourin, R. Plamondon, L. Beaumier, Structural interpretation of handwritten signature images, Int. J. Pattern Recogn. Artif. Intell. 8 (1994) 709–748.
[19] M. Hanmandlu, K.R. Murali Mohan, Vivek Gupta, Fuzzy logic based character recognition, Proceedings of the International Conference on Image Processing, Santa Barbara, USA, pp. 714–717.
[20] M.F. Azeem, M. Hanmandlu, N. Ahmad, Parameter determination for a generalized fuzzy model, Soft Computing, Published online: 19 January 2004, pp. 1–11.
[21] R.W. Zhou, C. Quek, Automatic fuzzy neural network driven signature verification system, in: Proceedings of the IEEE International Conference on Neural Networks USA, 1996, pp. 1034–1039.
[22] R. Sabourin, G. Genest, F.J. Preteux, Off-line signature verification local granulometric size distributions, IEEE Trans. Pattern Anal. Mach. Intell. 19 (9) (1997) 976–988.
[23] B. Fang, C.H. Leung, Y.Y. Tang, P.C.K. Kwok, K.W. Tse, Y.K. Wong, Offline signature verification with generated training samples, IEE Proc. Vis. Image Signal Process. 149 (2) (2002) 85–90.
[24] Y. Mizukami, M. Yoshimura, H. Miike, I. Yoshimura, An off-line signature verification system using an extracted displacement function, Pattern Recogn. Lett. 23 (2002) 1569–1577.
[25] M. Ammar, Progress in verification of skillfully simulated handwritten signatures, Int. J. Pattern Recogn. and Artif. Intell. 5 (1991) 337–351.

About the Author—M. HANMANDLU received his B.E. degree in Electrical Engineering from Osmania University, Hyderabad, India, in 1973, the M.Tech. degree in power systems from R.E.C. Warangal, Jawaharlal Nehru Technological University, India, in 1976, and the Ph.D. degree in control systems from Indian Institute of Technology, Delhi, India, in 1981. From 1979 to 1981, he was a senior scientific officer in Applied Systems Research program of the Department of Electrical Engineering, IIT Delhi. He joined the department as a lecturer in 1981 and became a Professor in 1997. He was with Machine Vision Group, City University, London, in 1988, and Robotics Research Group, Oxford University, in 1993, as part of the Indo-UK research collaboration. He was a Visiting Professor with the Faculty of Engineering, Multimedia University, Malaysia from March 2001 to March, 2003. He worked in the areas of Power Systems, Control, Robotics and Computer Vision, before shifting to fuzzy theory. His current research interests mainly include fuzzy modeling of dynamic systems and applications of fuzzy logic to image processing, document processing, bio-medical imaging and intelligent control. He is a member of IEEE and is listed in *Reference Asia; Asia's who's who of Men and Women of achievement; 5000 Personalities of the World (1998), American Biographical Institute*.

About the Author—M.H.M. YUSOF received his B.Sc. degree in Computer Science (Software Engineering) from University of Essex, United Kingdom. He is currently a tutor in the Faculty of Information Technology, Multimedia University, Malaysia and is also pursuing his M.Sc. (IT). His research interests include image processing, computer graphics, fuzzy logic and object oriented programming.

About the Author—VAMSI K. MADASU received his B.Tech degree in Electronics and Communication Engineering from Jawaharlal Nehru Technological University, India. He is currently a tutorial fellow pursuing his Ph.D. in the School of Information Technology and Electrical Engineering at the University of Queensland, Brisbane, Australia. His research interests include computer vision, image processing, fuzzy logic and its applications to pattern recognition.