Title **An MLP using Hough transform based fuzzy feature extraction for Bengali script recognition**

Authors **Shamik Sural ***
CMC Limited, 28 Camac Street, Calcutta 700 016, India.

P.K.Das
Dept. of Computer Science and Engineering,
Jadavpur University, Calcutta 700 032, India.

Abstract

We define fuzzy sets on the Hough transform of character pattern pixels from which additional fuzzy sets are synthesized using t-norms. A multilayer perceptron trained with a number of linguistic set memberships derived from these t-norms can recognize characters of Bengali scripts by their similarities to different fuzzy pattern classes.

Keywords : Bengali Script Recognition, Hough Transform, Fuzzy Sets, Multilayer Perceptron

Name and Address of the Author for Correspondence

Shamik Sural
151 Kanungo Park,
Garia
Calcutta 700 084
India

Telephone (91)-(033)-(4621867)
E-mail – shamik_s@hotmail.com

* Author is now with NIIT Limited, 6B Pretoria Street Calcutta 700 071, India.

An MLP using Hough transform based fuzzy feature extraction for Bengali script recognition

1. INTRODUCTION

Optical character recognition (OCR) is one of the most popular areas of research in pattern recognition because of its immense application potential. However, most of the available methods deal with the recognition of the Roman script and some of the oriental scripts like Kanji, Kana, etc (Govindan and Shivaprasad, 1990; Mori et al, 1992; Srihari et al, 1997). There are very few research reports on the recognition of Indian scripts. We present a document processing system for the recognition of an Indian script, namely, Bengali (also called *Bangla*). Bengali, one of the most popular languages in India and national language of Bangladesh, is based on the *Bangla* script. Processing of Bengali is different from that of the Roman script due to a number of reasons. The number of vowels and consonants is large in Bengali although there is no upper and lower case differentiation. More than 200 compound characters (called *yuktakhar*s) are formed by combining two or more consonants. The shape of a *yuktakhar* may be completely different from that of its individual component characters. Vowels and a few consonants act as modifiers when placed next to another consonant. When characters form a word, usually the individual letters get connected by a head line called *matra* or *sirorekha* as shown in Fig. 1. Thus, formation of merged characters is a norm rather than an aberration in this script. Individual characters appear in the upper zone, middle zone as well as in the lower zone of a text line. Further, the same character may appear in one zone in its normal form and in a different zone while acting as a modifier.

The two fundamental approaches to character recognition are feature classification and template matching. Template matching techniques are more sensitive to font and size variations of the characters and hence are not suitable for character recognition from noisy document images. However, selection and extraction of useful features is not always straightforward. Chaudhuri and Pal (1997a) have proposed a complete OCR system for Bengali documents using a combination of feature detection and template matching techniques. A different approach to character recognition is the use of Multilayer Perceptron (MLP) and other neural networks after they are trained with a set of standard patterns by supervised learning. This approach has earlier been successfully applied for the recognition of Roman scripts (Hussain and Kabuka, 1994; Hadar et al, 1995; Sural and Das, 1997). It may be mentioned that human reasoning is somewhat fuzzy in nature, which enables us to combine even visually degraded features in the brain using millions of neurons working in parallel. Fuzzy sets have the ability to model vagueness and ambiguity in data which is encountered in character recognition as well as in other pattern recognition problems. We propose here the application of the neuro-fuzzy technique for the recognition of Bengali (*Bangla*) script. Our method combines the robustness of feature extraction with the speed of operation of neural networks in a framework of fuzzy systems.

2. PREPROCESSING

In document image processing, a paper document is first scanned and binarized using an optical scanner to generate a bitmap image. The scanned image is often skewed due to mis-alignment of the document with the scanner axis. A number of methods have been suggested for the detection and correction of document skew of which the projection profile method (Hou, 1983) and the Hough transform method (Li et al, 1994; Jiang et al, 1997) are the most popular. Chaudhuri and Pal (1997b) have proposed a digital straight line based technique for skew angle detection from digitized Indian script documents and have observed that the projection profile methods are well suited for skew angles less than ± 10 degrees. It has been seen that in most of the practical situations, the skew angle is restricted to this range only. Also, due to the presence of *matra* in Bengali scripts, such documents have a much sharper projection profile than the Roman script documents. We, therefore, use the horizontal projection profile (hpp) method for the detection of skew in scanned Bengali documents. Skew angles even upto 30 degrees have been corrected by this method.

As mentioned earlier, a line of Bengali script can be partitioned into three horizontal zones, namely, upper zone, middle zone and lower zone as shown in Fig. 1. The upper zone is the zone above the *matra*, which usually contains parts of a few characters as well as some of the modifiers. The middle zone contains most of the characters, while the rest of the modifiers are located in the lower zone. A text line is located between scan lines whose horizontal projection profile histogram values are greater than a threshold t_{line}. Each text line is marked by a *start_line* and an *end_line* value. After a text line is detected, its vertical projection profile (vpp) is determined in the range *start_line* to *end_line*. If a number of successive vpp histogram values are greater than a threshold t_{word}, a word is considered to exist between these vertical lines. The scan lines with high hpp values are considered as the *matra*. The zone above the *matra* is detected as the upper zone. Pal and Chaudhuri (1995) have observed that in most of the Bengali scripts, the middle zone is about 52% of the line width. We have used this information to separate the middle and the lower zones. In the present work, we have processed Bengali documents containing only text matter.

Determination of characters within a word is done by a *matra* deletion procedure. Since the individual letters forming a word are usually connected by the *matra*, it is essential to delete the *matra* to separate the characters. After the *matra* is deleted from a word, the position of each character including the modifiers is identified using a recursive algorithm which detects the minimum containing rectangles (MCRs) of connected components of any shape and size. The algorithm starts with a black pixel in a character and recursively determines the max-min co-ordinates of the neighboring black pixels, thus extending the MCR. We have used the algorithm for segmentation of Roman scripts as reported in (Sural and Das, 1997). A segmented image of a part of a scanned Bengali newspaper is shown in Fig. 2. Some of the Bengali characters are formed by adding a dot below another component, which itself is a valid character as shown in Fig. 4. During the MCR detection stage, the main component and the dot, if present, get separated. When such a main component is recognized, the character position including the lower zone is searched for the presence of the dot. The main component and the dot information are combined in a post- processing step to correctly recognize the character.

3. CHARACTER RECOGNITION

After a Bengali document image is segmented, we use a number of fuzzy feature based multilayer perceptrons (explained later in sub-section 3.2) for the recognition of characters including vowels, consonants, modifiers and compound characters. For each character position, the connected components in each zone are processed separately by the MLPs and the results are combined to decide whether it is a normal character, a modifier or a compound character. As an example, Fig. 3 shows a compound character with a consonant in the middle zone, a consonant modifier in the upper zone and a vowel modifier in the lower zone. After zone separation, characters are submitted to an MLP and the results are combined after recognition.

It has been observed through statistical analysis of Bengali scripts that although there are a large number of *yuktakhars*, they occur only sparsely in written text (Pal and Chaudhuri, 1995). We, therefore, employ a hierarchy of MLPs where the first stage detects the vowels, consonants and the modifiers. If the first stage MLP fails to detect an input character, i.e., the outputs fall below a threshold as explained in section 4, it is presented to a second stage MLP trained with sixty commonly used *yuktakhars*. If both the MLP stages fail, the character is submitted to a third stage MLP trained with the rest of the *yuktakhars*. The characters used for training the second stage and the third stage MLPs have been selected in the order of their occurrence in the pioneering book (Vidyasagar, 1983) which is widely used for the learning of Bengali characters. However, the ordering of the characters for detection in the second and the third stage MLPs does not affect the recognition accuracy. All the MLPs are similar in structure and hence, switching over to a different MLP is simply a process of reading a new set of weights for the inter-layer connectors. The characters detected in each stage are shown in Fig. 4. A number of other compound characters are formed by combining some of these characters. Such characters are recognized by identifying their individual components.

In the following two sub-sections, we describe the fuzzy feature extraction technique using Hough transform and the operation of the multilayer perceptron with fuzzy input and output.

3.1 Fuzzy feature extraction

We use Hough transform for the extraction of features from characters in Bengali scripts. Hough transform (Illingworth and Kittler, 1988; Bhandarkar, 1994) is a method for the extraction of lines and curves from images. Han et al (1994) have proposed a fuzzy Hough transform method where an image point is treated as a fuzzy point. An important observation on Hough transform for line detection, which uses the mapping $\rho = x\cos\theta + y\sin\theta$, is that it provides three important characteristics of a straight line in an image pattern. These are the values of ρ, θ and the count of a ρ-θ accumulator cell. ρ and θ specify the position and orientation of a straight line, while the count specifies the length of the line in terms of the number of black pixels lying on it. If an input character pattern is corrupted by noise, some of the features may be missed out due to the thresholding done on the Hough transform accumulator cells. Keeping this in mind, we define a number of fuzzy sets to extract information from the (ρ,θ) accumulator cells. These fuzzy set membership functions are listed in table I for θ values in the first quadrant. Similar membership functions are defined for θ values in the other quadrants also. The fuzzy sets LL and SL of table I extract length information of the different lines in a character pattern. HL, VL and TL represent their skew while NT, NB, NVC, NR, NL and NHC give the spatial distribution of these lines. By this technique, we map the characteristics of the different lines in an image pattern into the properties of these fuzzy sets. The fuzzy sets of table I satisfy the following properties which justify their definition.

i. $\mu_A(x) \in [0,1]$ where A denotes any of the fuzzy sets in table I and $\mu_A(x)$, its membership function.

ii. The height of each fuzzy set, h(A) = 1 and hence each fuzzy set is normal.

iii. The 1-cuts i.e., the core of the fuzzy sets TL, NT, NB, NVC, NR, NL, NHC represent crisp features. The core of the fuzzy sets HL and VL denote the strictly horizontal and strictly vertical lines, respectively, while the core of LL is a diagonal line.

iv. Supports of the fuzzy sets LL and SL represent straight lines with all possible lengths in the pattern.

For a detailed discussion on fuzzy sets and their properties, refer to Klir & Yuan (1995) and Zimmermann (1991). Other fuzzy techniques for pattern recognition have been described by Kandel (1982) and Bezdek & Pal (1992).

Based on the basic fuzzy sets defined in table I, we synthesize additional fuzzy sets to represent each line in a character pattern as a combination of its length, position and orientation. Membership values for the synthesized fuzzy sets are determined for each ρ-θ accumulator cell using t-norms i.e., intersections on the basic fuzzy sets. A fuzzy intersection or t-norm *i* is a binary operation on the unit interval that satisfies the following properties for any p,q,r $\in$ [0,1] where p,q,r denote membership values in arbitrary fuzzy sets.

i. $i(p,1) = p$

ii. $q \le r \Rightarrow i(p,q) \le i(p,r)$

iii. $i(p,q) = i(q,p)$

iv. $i(p,i(q,r)) = i(i(p,q),r)$

A number of t-norms are available as fuzzy intersections of which we use the standard intersection : $i(p,q) = \min(p,q)$. The synthesized fuzzy sets are defined as *Long Slant Line near the Left* (LSL) $\equiv$ *i*(*i*(TL,LL),NL), *Short Slant Line near the Right* (SSR) $\equiv$ *i*(*i*(TL,SL),NR), *Nearly Vertical Long Line near the Left* (VLL) $\equiv$ *i*(*i*(VL,LL),NL), etc.

Similar basic fuzzy sets, namely, *Small Circle* (SC)*, Dense Circle* (DC), *Near Top* (NP), *Near Centre* (NC), etc. and synthesized fuzzy sets, namely, *Small Dense Circle near the Top* (SDT), *Large Dense Circle near the Centre (LDC)*, etc. are defined on (a,b,c) accumulator cells for circle extraction using the Hough transform $c = \sqrt{(x-a)^2 + (y-b)^2}$, where (a,b) denotes the centre of a circle and c, its radius. Since we do not perform any thresholding on the (a,b,c) accumulator cells, even very small circles are retained in the proposed Hough transform method. Membership values of these circles are high in the fuzzy set *Small Circle* while it is low in the fuzzy set *Large Circle*. For other pattern recognition problems, suitable fuzzy sets may be similarly synthesized from the basic fuzzy sets defined on Hough transform.

It should be noted that, a non-null support of a synthesized fuzzy set implies the presence of the corresponding feature in a pattern. We, therefore, choose the height of each synthesized fuzzy set to define a feature element and the set of ‘n’ such feature elements constitute an n-dimensional feature vector for a character. The height of a fuzzy set is the maximum membership grade obtained by any element in that set (Klir and Yuan, 1995). In our system, we extract 20 features for line and circle extraction. Some of the fuzzy feature elements extracted from eight scanned Bengali characters are shown in table II.

3.2 MLP with fuzzy input/output

A multilayer perceptron is a network of simple two-state processors, which are arranged in a manner similar to the biological neurons (Lippmann, 1987). The processors, often called neurons, usually have sigmoid transfer functions and are arranged in a number of layers. The lowermost layer is the input to the MLP and the topmost layer is the MLP output. There can be any number of hidden layers between the input and the output. In multilayer perceptron architecture, there is no interconnection among neurons within a layer while neurons of adjacent layers are fully connected through weighted links. A multilayer perceptron is first trained with a number of sample patterns from different classes of learning and then it acts as a multidimensional pattern classifier. To train the network, the sample patterns are presented to the MLP input one at a time and the corresponding expected outputs are made known to the network. Starting with initial random weights for the inter-layer connectors, the network iterates through a series of weight updation procedures, which minimize the square error between the actual output and the expected output. Weight updation is usually done by a gradient descent rule of which back propagation is the most popular (Rumelhart et al, 1986). This method of training a neural network with sample patterns is called supervised learning. The problem of character recognition where the expected patterns are known, is a typical application of supervised learning of multilayer perceptron acting as a classifier.

An MLP trained with the sample patterns can classify a test pattern by setting one of the outputs high and the rest to low.

When an n-dimensional feature vector is extracted from a degraded character pattern by Hough transform during recognition, the strength of the features in the vector may vary due to the presence of noise. Any optical character recognition system has to differentiate between similar looking patterns and must also recognize a learnt pattern correctly in the presence of noise. To combat the effect of noise, we generate membership values in three linguistic fuzzy sets, namely, *weak*, *moderate* and *strong* from the individual feature elements. The linguistic set membership functions are derived from the Butterworth filter transfer functions (Millman and Halkias, 1972) as shown below.

$$\mu_{\text{weak}}(x) = \left[1 + \left(\frac{x}{a}\right)^{2m}\right]^{-\frac{1}{2}}, \mu_{\text{moderate}}(x) = \left(\left[1 + \left(\frac{x}{a_1}\right)^{2m}\right]\left[1 + \left(\frac{a_2}{x}\right)^{2m}\right]\right)^{-\frac{1}{2}}, \mu_{\text{strong}}(x) = \left[1 + \left(\frac{a}{x}\right)^{2m}\right]^{-\frac{1}{2}}$$

Here x = a (a_1, a_2 for μ_{moderate}) is the cut-off point for all values of ‘m’ where ‘m’ controls the slope of the functions. The n-dimensional feature vectors extracted from Hough transform are thus mapped into 3n-dimensional vectors which form the MLP input both during training and recognition. These membership functions are chosen so that the belongingness of a feature value into each of the linguistic set can be controlled simply by setting two parameter values for μ_{weak} as well as μ_{strong} and three for μ_{moderate}. The advantage of using linguistic features is that, for small variations in the extracted feature values, the linguistic set memberships remain unchanged. The system can then recognize even degraded character patterns.

In a conventional MLP, an input pattern belongs only to a particular output pattern class. We, however, use fuzzy character pattern classes as outputs and the MLP is trained to learn the degree by which a feature vector belongs to each of these classes. For a P-class problem domain with P nodes in the output layer of the MLP, the membership of each input pattern in the output fuzzy classes is determined in terms of the distance between the corresponding feature vectors. Thus, if the patterns, before converting to linguistic sets, are represented by the n-dimensional feature vectors $\overline{F}_i$, i=1,2,..,P, then the Euclidean distance between the feature vectors $\overline{F}_i$ and $\overline{F}_k$ is calculated as follows.

$$d_{ik} \quad = \sqrt{\sum_j \left(F_{ij} - F_{kj} \right)^2} \qquad k = 1,2,..., P. \tag{1}$$

Here the summation is done over all the 'n' feature elements subscripted by j. The membership of the i^{th} character pattern to the k^{th} fuzzy pattern class, and hence the value of the k^{th} expected output of the MLP for the input vector $\overline{F}_i$ is then determined using the following relation.

$$O^i_{k(\exp)} = \mu_k\left(\overline{F}_i\right) = \left[1 + \left(\frac{d_{ik}}{f_{den}} \right)^{f_{pow}} \right]^{-1} \tag{2}$$

Here 'f_{den}' and 'f_{pow}' control the membership grades of the different output fuzzy sets for each input pattern. For all the fuzzy class membership functions, $\mu_k\left(\overline{F}_i\right) \in [0,1]$, $\mu_k\left(\overline{F}_i\right) = \mu_i\left(\overline{F}_k\right)$, $\mu_k\left(\overline{F}_k\right) = 1$, and $d_{ik} \geq d_{il}$ $\Rightarrow \mu_k\left(\overline{F}_i\right) \leq \mu_l\left(\overline{F}_i\right)$. Further, for $f_{den} \rightarrow 0$ and $f_{pow} \rightarrow \infty$, the fuzzy MLP output reduces to a conventional MLP output with $O^i_{k(\exp)} = 1$ for i = k, and 0 otherwise. The effect of the different parameters of eq. (2) on the recognition accuracy is explained later in this section.

Thus, the membership functions satisfy the requisite properties of fuzzy sets and may be considered as a generalization of crisp perceptron outputs. We train the MLP with the input fuzzy feature vectors and fuzzy expected outputs by the back propagation algorithm proposed by Rumelhart et al (1986) and modified by Gupta and Rao (1994) for handling fuzzy inputs and outputs. In this method, arithmetic addition and multiplication are replaced by fuzzy norms and t-norms. There are a few other methods of using fuzzy sets in MLP operations.

Hayashi et al (1993) have proposed a fuzzy neural network with fuzzy signals and weights. Their fuzzy neural network is obtained by directly fuzzifying the classical feedforward neural network with one or more layers. As this fuzzy neural network deals with arithmetic operations on fuzzy numbers, the classification time increases considerably. Pal & Mitra (1992) have proposed an MLP with fuzzy input and output where the operation of each neuron is non-fuzzy in nature.

The fuzzy MLP used in our system is structurally similar to a crisp perceptron with the number of output units, being the same as the number of *distinct* patterns to be classified in each stage. Thus the number of outputs in the three stages of the MLP are 57, 60 and 68, respectively. The number of input units is 60, formed by generating a linguistic feature vector from each set of 20 features extracted by Hough transform. The number of nodes in the hidden layer of the first stage is 30 while that of each of the later two stages is 25. The number of epochs required for training depends on the stage of the MLP and the initial set of weights. A typical set of values is 5000 for the first stage, 7000 for the second stage and 8000 for the third stage. Since the total number of patterns to be recognized is very high, it is impractical to use a single stage MLP to recognize all the different shapes of Bengali script. This limitation coupled with the fact that the occurrence of *yuktakhars* is much less compared to the ordinary characters, has prompted us to consider the multi-stage implementation of the MLP based Bengali character recognition system.

The recognition decision is based on the α-cuts of the output fuzzy sets for α = th_opt, a threshold value for the MLP outputs. The α-cuts are determined by the parameters f_{den} and f_{pow} of eq. (2). For $f_{den} > d_{ik}$, the number of elements in the α-cuts increases with higher values of f_{pow}. For a fixed value of the parameters f_{den} and f_{pow}, if th_opt is low, α-cuts of the output fuzzy sets may contain more than one element while a high value of th_opt results in null α-cuts for some or all of the outputs. If each of the α-cuts contain one element, then during recognition only one of the MLP outputs goes above the threshold. The highest value output is then considered as the detected character. If, however, the membership value is above the threshold for more than one output, indicating a *misclassification*, the character is marked for correction by user intervention. All the high value outputs may also be considered for a dictionary search to uniquely identify the character. The advantage of using fuzzy set membership functions at the MLP output is that, only outputs with high membership values need to be considered for the search. Since the MLP outputs denote their similarities to the different pattern classes, this decision making process is justified. The fuzzy feature based MLP has been found to be stable for different initial random values of the inter-layer connector weights. The feature set has been chosen after a sensitivity analysis of the MLP output for a large number of fuzzy features using the neural network based method proposed by De et al (1997).

4. RESULTS AND DISCUSSIONS

The fuzzy OCR system has been first tested on Bengali document images corrupted by simulated noise. We model a noisy document image as a two-state Markov chain shown in Fig. 5(a) in which a *Random* state 'R' produces errors in the image with a probability 'r' while a *Burst* state 'B' corrupts the image pixels with a probability 'b' where r << b. The system remains in the Random (Burst) state for the next pixel position with probability q (Q), given that it is in the Random (Burst) state for the current pixel position. With a probability 1-q (1-Q for Burst state), it makes a transition to the other state. The model is similar to the Gilbert-Elliott model for noise in communication channels with memory (Schlegel and Herro, 1990). However, propagation of burst is spatial in a noisy document as shown in Fig 5(b) unlike communication channels where it is temporal in nature. The error density ratio is defined as $\Delta = \frac{b}{r}$ and the average burst length, $\lambda = \frac{Q}{1-Q}$. The steady state probabilities of the random state and the burst state are $P_R = \frac{1-Q}{2-Q-q}$ and $P_B = \frac{1-q}{2-Q-q}$. The pixel error probability on the document is $P_e = bP_B + rP_R$.

Fig. 6(a) shows the variation in character recognition error in a Bengali document for different values of λ with b = 0.5, r = 0.0005 and $P_e = 0.005$. It is seen that for small values of λ, the recognition error percentage is high and it goes down with increase in the value of λ. When λ is small, there are a large number of noise bursts, each with a short length. As a result, the number of recognized characters is less, resulting in ambiguous choices. As λ increases, the number of bursts goes down and the affected characters can be correctly recognized. The variation of recognition error for different values of Δ is shown in Fig. 6(b). Here also, $P_e = 0.005$ with $q = 0.999$ and $Q = 0.99$. It is seen that for small values of Δ, unresolved error is high since noise occurs with almost equal probability in the two states. For higher values of Δ, random noise is corrected by the fuzzy MLP itself while errors caused by dense bursts are concentrated in only a few characters, reducing the recognition error percentage. If a dictionary search step is introduced, these errors can be easily corrected.

In the implemented system, a 300 dpi flatbed HP scanner is used to scan the Bengali documents from different sources including newspapers, magazines and novels and an Intel Pentium based machine is used for character recognition. There is a large variation in the printing quality of these documents as the materials are composed manually as well as using monotype and linotype methods. Typical printed Bengali documents have uneven line spacing, varying thickness and they often contain broken fonts. The input and output parameters of the MLP are selected based on the simulation test results. The value of the parameter m is typically chosen as 4 and f_{pow} is set at 0.87. The ratio (d_{ik}/f_{den}) is kept less than unity.

The proposed system has the advantage that the time required for recognition of the vowels, consonants and the modifiers is small, being the propagation time of one stage of MLP after feature extraction. Recognition accuracy of the system for these characters is also very high for documents printed in pica, small pica and long primer. As the fuzzy sets defined in our system does an implicit normalization of the character dimensions, the system is robust enough to handle different point sizes of the same font. Since some of the *yuktakhars* are similar looking as shown in Fig. 4, the MLPs are unable to distinguish them correctly in a few cases. A recognition error occurs either due to *misclassification* in which a character is recognized as a different character or due to *classifier failure*, in which a character is not recognized at all. In the second case, all the MLP outputs fall below the threshold th_opt. It has been observed that classifier failure error is very low except in cases where the characters in a scanned document are completely defaced due to very poor quality of paper. As the *yuktakhars* occur only sparsely in Bengali script, the overall accuracy of the system is about 98%. Table III lists the performance of the system in terms of speed and accuracy of recognition for different groups of Bengali characters. The recognition accuracy will be higher if a post-processing step using dictionary search is introduced to resolve the ambiguous characters based on valid words in the dictionary.

Characters of another Indian script, namely Devnagari, are organized similar to Bengali. A fuzzy OCR system for Devnagari can, therefore, be developed in a similar manner. The recognition accuracy is also expected to be of the same order. The linguistic set definitions and the MLP with fuzzy input/output can be used with other fuzzy feature extraction techniques also. Besides Indian script recognition, fuzzy set generation from Hough transform is useful for a variety of pattern recognition problems including Roman script recognition. Fuzzy features like *thick* lines, *thin* lines, *nearly parallel* lines, lines *slightly above* or *slightly below* fixed lines, similar to those proposed by Krishnapuram et al (1993) may be extracted from document as well as non-document images using this technique.

References

Bezdek, J.C. and S.K.Pal Eds. (1992). *Fuzzy models for pattern recognition - Methods that search for structure in data*. IEEE Press, NY, USA.

Bhandarkar, S.M. (1994). A fuzzy probabilistic model for the generalized Hough transform. *IEEE Trans. on Systems Man and Cybernetics* 24, 745-759.

Chaudhuri, B.B and U.Pal (1997a). A complete printed *Bangla* OCR system. *Pattern Recognition* 30, 531-549.

Chaudhuri, B.B. and U.Pal (1997b). Skew angle detection of digitized Indian script documents. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 19, 182-186.

De, R.K., N.R.Pal and S.K.Pal (1997). Feature analysis : Neural network and fuzzy set theoretic approaches. *Pattern Recognition* 30, 1579-1590.

Govindan, V.K. and A.P.Shivaprasad (1990). Character recognition - A review. *Pattern Recognition* 23, 671-683.

Gupta, M.M. and D.H.Rao (1994). On the principles of fuzzy neural networks. *Fuzzy Sets and Systems* 61, 1-18.

Hadar, I. A., T.A. Diep and H.Garland (1995). High accuracy optical character recognition using neural network with centroid dithering. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 17, 218-224.

Han J.H., L.T.Koczy and T.Poston (1994). Fuzzy Hough transform. *Pattern Recognition Letters* 15, 649-658.

Hayashi, Y., J.J.Buckley and E.Czogala (1993). Fuzzy neural network with fuzzy signals and weights. *International Journal of Intelligent Systems* 8, 527-537.

Hou, H.S. (1983). *Digital Document Processing*. Wiley, New York.

Hussain, B. and M.R.Kabuka (1994). A novel feature recognition neural network and its application to character recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 16, 98-106.

Illingworth, J. and J.Kittler (1988). A survey of the Hough transform. *Computer Vision, Graphics and Image Processing* 44, 87-116.

Jiang, H-F, C-C Han and K-C Fan (1997). A fast approach to the detection and correction of skew documents. *Pattern Recognition Letters* 18, 675-686.

Kandel, A. (1982). *Fuzzy techniques in pattern recognition*. John Wiley & Sons, USA.

Klir, G.J. and B.Yuan (1995). *Fuzzy sets and fuzzy logic - Theory and applications*. Prentice Hall Inc., Englewood Cliffs, NJ, USA.

Krishnapuram, R., J.M.Keller and Y.Ma (1993). Quantitative analysis of properties and spatial relations of fuzzy image regions. *IEEE Trans. on Fuzzy Systems* 1, 222-233.

Li, D.S., G.R.Thoma and H.Wechsler (1994). Automatic page orientation and skew angle detection for binary document images. *Pattern Recognition* 27, 1325-1344.

Lippmann, R.P. (1987). An introduction to computing with neural nets. *IEEE ASSP Magazine*, 4-22.

Millman, J. and C.C.Halkias (1972). *Integrated Electronics : Analog and Digital Circuits and Systems*. McGrawHill, Singapore.

Mori, S., C.Y.Suen and K.Yamamoto (1992). Historical review of OCR research and development. *Proc. IEEE*, 1029-1058.

Pal, S.K. and S.Mitra (1992). Multilayer perceptron, fuzzy sets and classification. *IEEE Trans. on Neural Networks* 3, 683-697.

Pal, U. and B.B.Chaudhuri (1995). Computer recognition of printed Bangla script. *International Journal of System Science* 26, 2107-2123.

Rumelhart, D.E., G.E.Hinton and R.J.Williams (1986). Learning internal representation by error propagation. In: D.E.Rumelhart and J.L.McClelland, Eds., *Parallel Distributed Processing : Explorations in the microstructure of cognition, Vol. 1 : Foundations*, Chapter 8. MIT Press.

Schlegel, C.B. and M.A.Herro (1990). A burst error correcting Viterbi algorithm. *IEEE Trans. on Communication* 38, 285-291.

Srihari, S.N., T.Hong and G.Srikantan (1997). Machine printed Japanese document recognition. *Pattern Recognition* 30, 1301-1313.

Sural, S. and P.K.Das (1997). A document image analysis system on parallel processors. *Proc. Fourth International Conference on High Performance Computing*, IEEE Computer Society Press, California, USA, 527-532.

Vidyasagar, I. (1983). *Vidyasagar's Varna Parichaya (Part Two) Nirmal Edition*. Nirmal Book Agency, Calcutta, India.

Zimmermann, H.-J. (1991). *Fuzzy set theory and its applications*. Kluwer Academic publishers, Norwell, Massachusetts, USA.

Table I. Membership functions of fuzzy sets defined on Hough transform accumulator cells. X and Y denote the height and the width of each character pattern.

Fuzzy Set	Membership Function	Notation
Long Line	$\left(\frac{\text{count}}{\sqrt{X^2 + Y^2}}\right)$	LL
Short Line	$2\left(1-\frac{\text{count}}{\sqrt{X^2 + Y^2}}\right)$ if count $> \frac{\sqrt{X^2 + Y^2}}{2}$ $2\left(\frac{\text{count}}{\sqrt{X^2 + Y^2}}\right)$ if count $\leq \frac{\sqrt{X^2 + Y^2}}{2}$	SL
Nearly Horizontal Line	$\left(\frac{\theta}{90.0}\right)$	HL
Nearly Vertical Line	$\left(1-\frac{\theta}{90.0}\right)$	VL
Slant Line	$\left(\frac{2\theta}{90.0}\right)$ if $\theta \leq 45.0$ $2\left(1-\frac{\theta}{90.0}\right)$ if $\theta > 45.0$	TL
Near the Top	$\left(\frac{\rho}{X}\right)$ if HL > VL 0 otherwise	NT
Near the Bottom	$\left(1-\frac{\rho}{X}\right)$ if HL > VL 0 otherwise	NB
Near the Vertical Centre	$\left(\frac{2\rho}{X}\right)$ if (HL > VL and $\rho \leq \frac{X}{2}$) $2\left(1-\frac{\rho}{X}\right)$ if (HL > VL and $\rho > \frac{X}{2}$) 0 otherwise	NVC
Near the Right Border	$\left(\frac{\rho}{Y}\right)$ if VL > HL 0 otherwise	NR
Near the Left Border	$\left(1-\frac{\rho}{Y}\right)$ if VL > HL 0 otherwise	NL
Near the Horizontal Centre	$\left(\frac{2\rho}{Y}\right)$ if (VL > HL and $\rho \leq \frac{Y}{2}$) $2\left(1-\frac{\rho}{Y}\right)$ if (VL > HL and $\rho > \frac{Y}{2}$) 0 otherwise	NHC

Table II Some of the fuzzy features extracted by Hough transform from eight scanned Bengali characters

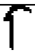
Character Pattern	Hough cell size ($\Delta\rho$, $\Delta\theta$) = ($0.5, 1^0$), (Δa, Δb, Δc) = ($1,1,0.8$)				
	Values of the feature elements				
	LSL	**SSR**	**VLL**	**SDT**	**LDC**
	0.32	0.81	0.19	0.80	0.31
	0.42	0.45	0.25	0.94	0.52
	0.63	0.83	0.21	0.43	0.13
	0.63	0.96	0.13	0.11	0.35
	0.25	0.53	0.12	0.20	0.27
	0.36	0.14	0.88	0.46	0.14
	0.29	0.78	0.25	0.55	0.35
	0.28	0.32	0.27	0.91	0.62

Table III Performance of the fuzzy OCR system in terms of speed and accuracy of recognition

Group of characters	Speed of recognition (Characters / minute)	Recognition accuracy (%)
Vowels	165	99.2
Consonants	163	99.0
Modifiers	154	98.8
First set of compound characters	135	88.6
Second set of compound characters	102	82.2
Overall	**160**	**98.1**

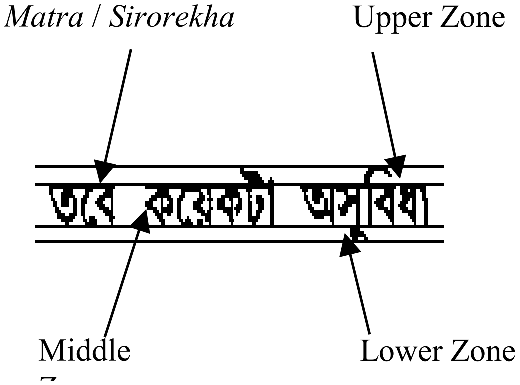
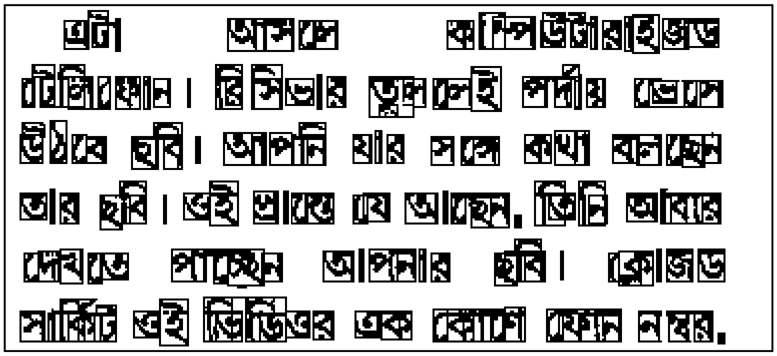


Figure 1



এটা আসলে কম্পিউটারাইজড
টেলিফোন। রিসিভার তুললেই পর্দায় ভেসে
উঠবে ছবি। আপনি যার সঙ্গে কথা বলছেন
তার ছবি। ওই প্রান্তে যে আছেন, তিনি আবার
দেখতে পাচ্ছেন আপনার ছবি। ক্লোজড
সার্কিট ওই ভিডিওর এক কোণে ফোন নম্বর,

Figure 2

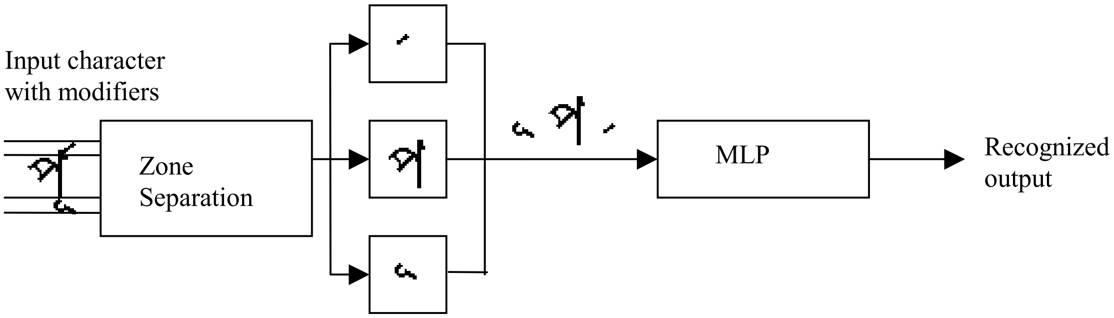
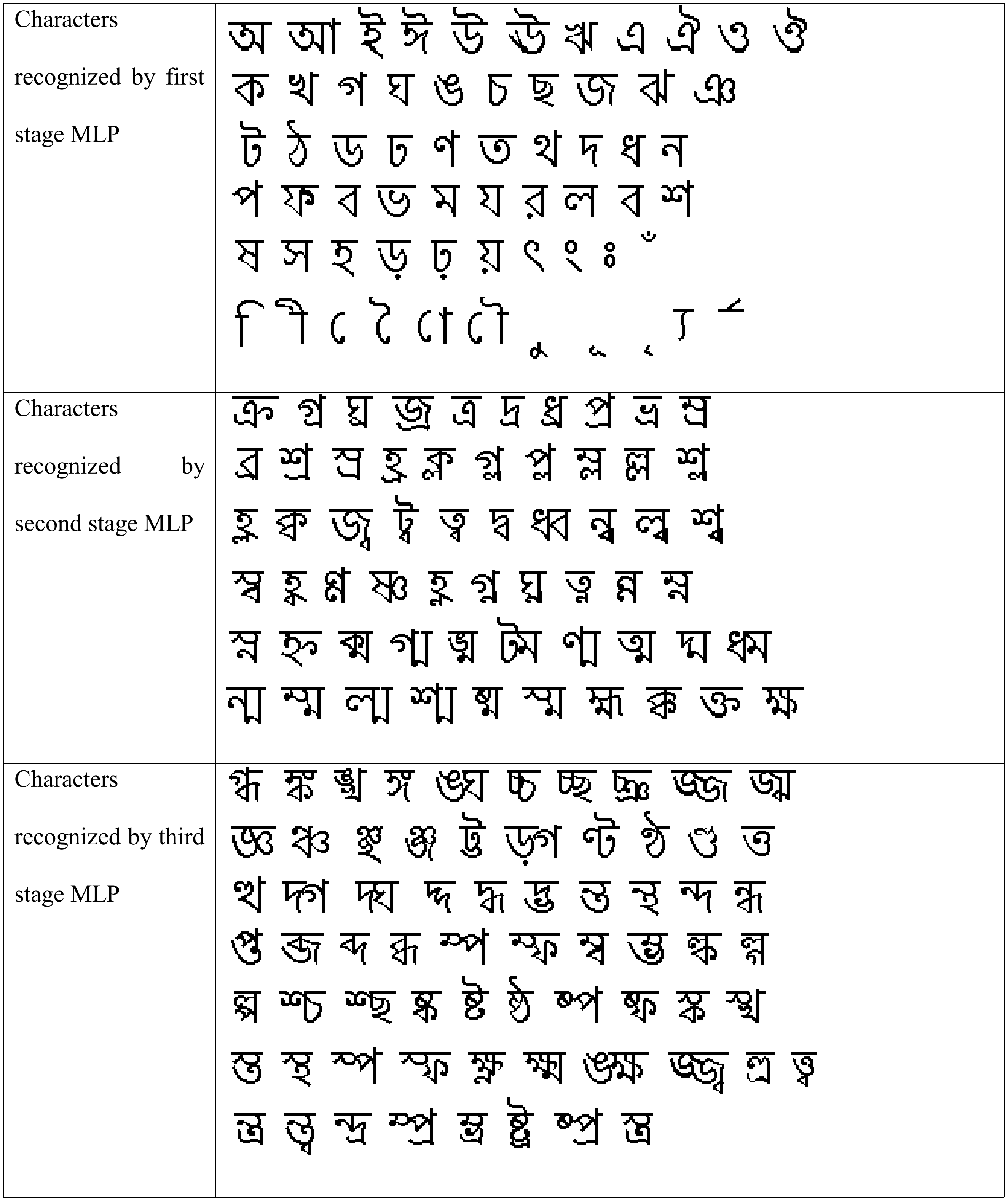

Input character
with modifiers
Zone
Separation
MLP
Recognized
output

Figure 3



Characters recognized by first stage MLP	অ আ ই ঈ উ ঊ ঋ এ ঐ ও ঔ ক খ গ ঘ ঙ চ ছ জ ঝ ঞ ট ঠ ড ঢ ণ ত থ দ ধ ন প ফ ব ভ ম য র ল ব শ ষ স হ ড় ঢ় য় ৎ ং ঃ ঁ ি ী ে ৈ ো ৌ ু ূ ৃ ্য র্
Characters recognized by second stage MLP	ক্র গ্র ঘ্র জ্র ত্র দ্র ধ্র প্র ভ্র ম্র ব্র শ্র স্র হ্র ক্ল গ্ল প্ল ম্ল ব্ল শ্ল হ্ল ক্ব জ্ব ট্ব ত্ব দ্ব ধ্ব ন্ব ল্ব শ্ব স্ব হ্ব ন্ন ষ্ণ হ্ন গ্ন ঘ্ন ত্ন ম্ন স্ন স্ন হ্ণ ক্ম গ্ম ঙ্ম ট্ম ণ্ম ত্ম দ্ম ধ্ম ন্ম স্ম ল্ম শ্ম ষ্ম স্ম হ্ম ক্ক ক্ত ক্ষ
Characters recognized by third stage MLP	ঙ্খ ঙ্ক ঙ্খ ঙ্গ ঙ্ঘ চ্চ চ্ছ চ্ছ্ব জ্জ জ্ঝ জ্ঞ ঞ্চ ঞ্ছ ঞ্জ ট্ট ড্গ ণ্ট ণ্ঠ ণ্ড ত্ত ত্থ দ্গ দ্ঘ দ্দ দ্ধ দ্ভ ন্ত ন্থ ন্দ ন্ধ প্ত ব্জ ব্দ ব্ধ ম্প ম্ফ ম্ব ম্ভ ল্ক ল্প ল্ফ শ্চ শ্ছ ষ্ক ষ্ট ষ্ঠ ষ্প ষ্ফ স্ক স্খ স্ত স্থ স্প স্ফ ক্ষ্ণ ক্ষ্ম ঙ্ক্ষ জ্জ্ব ন্ড্র ত্ত্ব ন্ত্র ত্ত্ব ন্দ্র স্প্র স্ত্র ষ্ট্র ষ্প্র স্ত্র

Figure 4

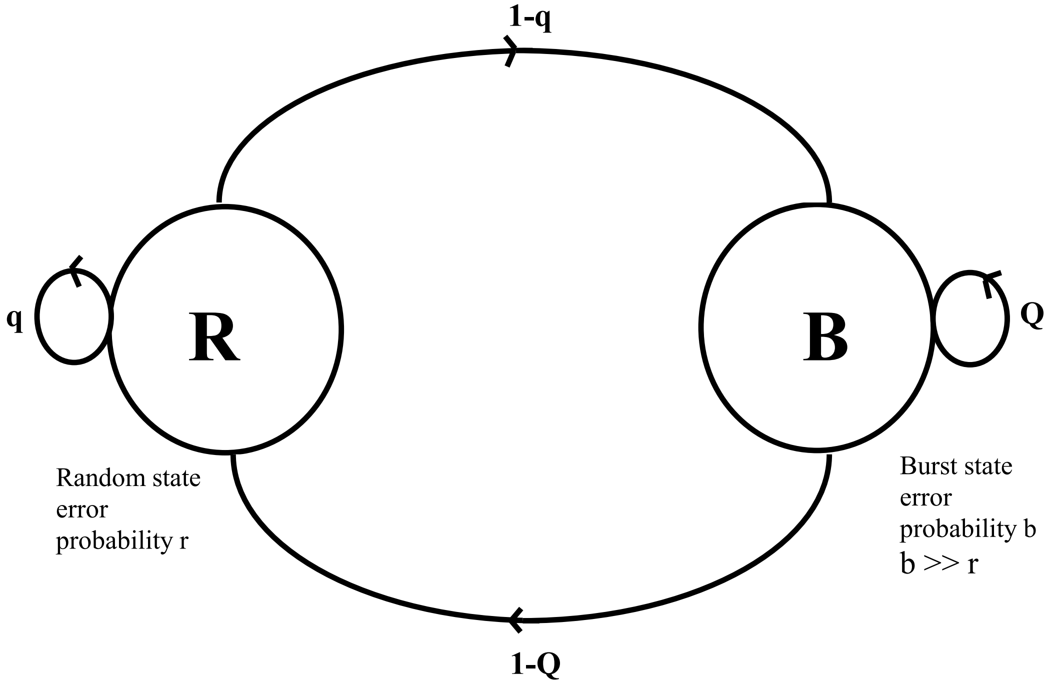

1-q
q
R
B
Q
Random state
error
probability r
Burst state
error
probability b
b >> r
1-Q

Figure 5(a)

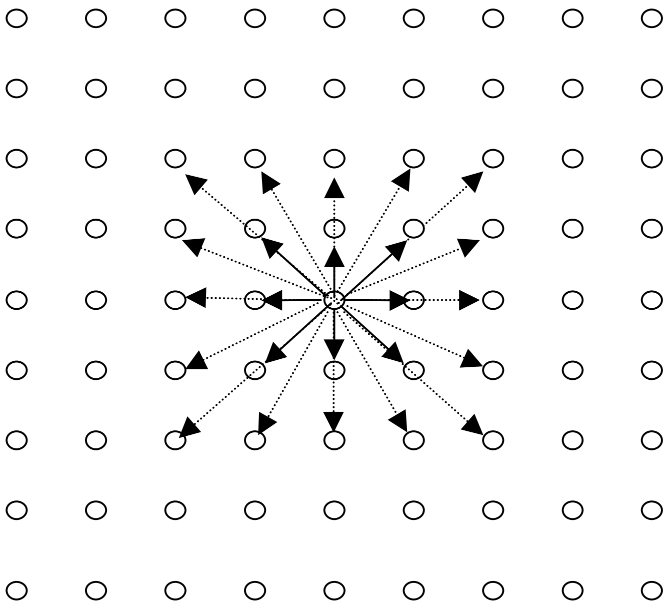


Figure 5(b)

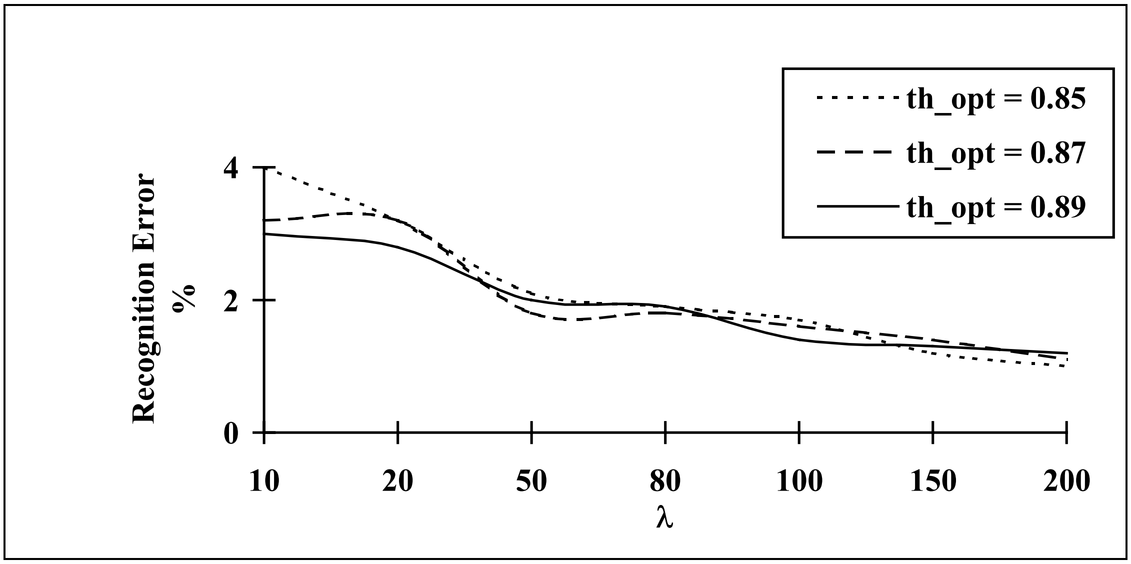

th_opt = 0.85
th_opt = 0.87
th_opt = 0.89
4
2
0
Recognition Error
%
10
20
50
80
100
150
200
λ

Figure 6 (a)

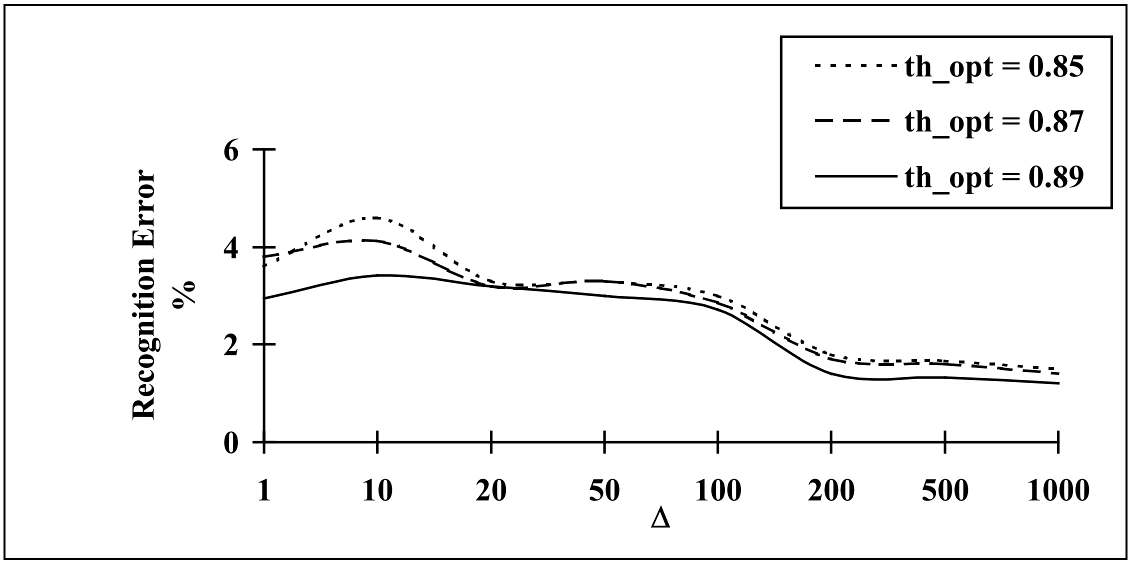

th_opt = 0.85
th_opt = 0.87
th_opt = 0.89
Recognition Error
%
6
4
2
0
1
10
20
50
100
200
500
1000
Δ

Figure 6(b)

Figure Captions

Figure 1 Different text zones in a line of scanned Bengali script.

Figure 2 Segmented image of a part of a scanned Bengali newspaper.

Figure 3 Recognition of a character with modifiers from a Bengali script.

Figure 4 Characters of Bengali script recognized by different stages of MLP.

Figure 5 (a) Two-state Markov chain model of a noisy document image.

(b) Spatial propagation of noise burst in a document image.

Figure 6 (a) Variation of character recognition error in Bengali scripts with average burst length ‘λ’.

(b) Variation of character recognition error in Bengali scripts with error density ratio ‘Δ’.