Input Fuzzy Modeling for the Recognition of Handwritten Hindi Numerals

M. Hanmandlu
Dept. of Elect. Engg.
I.I.T. Delhi
New Delhi, India
mhmandlu@ee.iitd.ac.in

J. Grover
Dept. of Mathematics
I.I.T. Delhi
New Delhi, India
jyotsana_g2000@yahoo.com

V. K. Madasu
School of Engg. Sys.
Qld.Univ. of Tech.
Brisbane, Australia
v.madasu@qut.edu.au

S. Vasikarla
School of Info. Tech.
American InterCon.Uni.
Los Angeles, USA
svasikarla@la.aiuniv.edu

Abstract

This paper presents the recognition of Handwritten Hindi Numerals based on the modified exponential membership function fitted to the fuzzy sets derived from normalized distance features obtained using the Box approach. The exponential membership function is modified by two structural parameters that are estimated by optimizing the criterion function associated with the input fuzzy modeling. We then utilize a 'Reuse Policy' that provides guidance from past error values of the criteria function to accomplish the reinforcement learning. We will also show how the 'Reuse Policy' improves the speed of convergence of the learning process over other strategies that learn without reuse. There is a 25-fold improvement in training with the use of the reinforcement learning Experimentation is carried out on a limited database of around 3500 Hindi numeral samples. The overall recognition rate is found to be 95%.

Keywords: Membership functions, structural parameters, Reinforcement, input fuzzy modeling

1. INTRODUCTION

Most of the Indian scripts are distinguished by the presence of matras (or, character modifiers) in addition to main characters as against the English script that has no matras. Therefore, the algorithms developed for them are not directly applicable to Indian scripts. Many OCRs for Indian scripts have been reported in literature [1,3,4,5,6]. However, none of these papers have considered Handwritten Hindi (Devanagari) text consisting of composite characters involving matras. In this paper, we present a recognition system for handwritten Hindi numerals. However, the proposed recognition scheme is applicable to composite characters as well as to individual components obtained after decomposition. The system has been tested on handwritten samples collected from several individuals.

Printed Hindi character recognition based on Neural Networks and distance classifiers is reported in [1, 5, 6]. These results are extended to Bengali characters [6], which also have the header line like Hindi characters. Structural features like concavities and inter-sections are used as features. A similar approach is tried for Gujarati [4] with limited success. Reasonable results are reported for Gurumukhi script [5]. Preliminary recognition results are also available in the literature on the two popular scripts in the south India – Tamil and Kannada [3]. Sinha et al. [2, 7] have reported various aspect ratios of Devanagari script recognition. Sethi and Chatterjee [8] have described Devanagari numeral recognition based on the structural approach. The primitives used are horizontal and vertical line segments, right and left slants. A decision tree is employed to perform the analysis depending on the presence/absence of these primitives and their interconnections. A similar strategy is applied to the constrained hand printed Devanagari characters in [9]. Neural network approach for isolated characters is also reported in [10].

Handwritten input usually comes as connected or partially connected strings of characters, and the first stage in the recognition process usually involves 'chopping' the strings into locally separate entities. This process is known as segmentation. Recognition is then performed using these entities. Since segmentation is a hard problem within itself, the assumption of pre-segmented inputs eases the task and schemes have to be developed for isolated input characters.

The perturbations due to writing habits and instruments are taken into account in the recognition of off-line handwritten English numerals in [12]. The back-propagation neural network is used in [11] for the recognition of handwritten characters whereas feature extraction is done using three different approaches, namely, ring, sector and hybrid. The features consist of normalized vector distances and angles. The hybrid approach, which combines the ring and sector approaches, is found to yield the best result. The same features are adopted in [13]. We will follow the same methodology for feature extraction as employed in [13].

We will now discuss the concept of Reuse Policy. A past policy provides a bias to guide the exploration of the environment and speed up the learning of a new action policy. The success of this bias depends on whether the past policy is "similar" to the actual policy or not. In this paper we make use of this concept in devising a new reinforcement learning algorithm that reuses the past errors to bias the learning of a new one.

Reinforcement learning [17] is a widely used tool to solve different tasks in different domains. Here, domain refers to the rules that define how the actions of the learning agent influence the environment, i.e. the state transition

International Conference on Information Technology (ITNG'07)
0-7695-2776-0/07 $20.00 © 2007 IEEE

function. Task is defined as the specific problem that the agent is trying to solve in the domain. The goal of this work is to study how action policies that are learned to solve a defined set of tasks can be used to solve a new and previously unseen task. In this work, we design a new learning scheme that implements the Reuse Policy ideas of [18, 19] efficiently. This learning allows us to reuse the past errors to learn a new one, improving the results of learning from scratch. The improvement is achieved without prior knowledge.

Another important aspect of our recognition scheme is the introduction of input fuzzy modeling where the need for an output is eliminated, thus having a big advantage over existing fuzzy models which are plagued by the need for an output. We also propose a criterion function as part of a fuzzy rule so as to classify the input information in the form of fuzzy sets or granules.

2. PREPROCESSING

The scanned image is first converted into binary image containing '0' and '1' pixels only. Pre-processing techniques like thinning [15], slant correction and smoothing are then applied.

After performing these techniques, there would be extra '0's on all four sides of a numeral. To standardize the numerals, extra rows and columns containing only zeros are removed from all four sides of the numeral. Depending on the Aspect Ratio (AR) of the numeral, categorization of numerals is done. Aspect ratio is the ratio of height to width of the image. Irrespective of the size of a handwritten numeral, this aspect ratio is always found to be nearly constant for all samples. For example, for numerals and it is around 2 while for other numerals it is around 1.5. Hence images having AR greater than 2 are fitted into a window of size 60 x32 while others are fitted into a window of size 42x32.

For validating the recognition scheme, a database of totally unconstrained handwritten numerals is created using samples obtained from handwritten application forms. The numerals are written in varying writing styles, sizes and stroke widths. The database also includes some samples that are difficult to be recognized even by humans. This database is divided into two disjoint sets, one for training and the other for testing.

The training set captures as many variations and styles of a numeral class as possible. In the training phase, we make use of the concept by which each feature when collected over several samples gives rise to a fuzzy set. We then construct a knowledge base (KB) which consists of means and variances of features of all fuzzy sets. The features extracted from the training set are stored in the KB and in the recognition phase, used as reference features for comparing with those of an unknown numeral.

3. FEATURE EXTRACTION

For extracting the features, the Box approach proposed in [13,14] is used. This approach requires the spatial division of the character image. The major advantage of this approach stems from its robustness to small variations, ease of implementation and relatively high recognition rate.

Each character image is divided into 24 boxes so that the portions of a numeral will be in some of these boxes. The choice of the box size is discussed in [16] and reviewed in Section 6 (results). There could be many boxes that are empty, as evident in Fig. 1 in which numeral is enclosed in the 6x4 grid. However, all boxes are considered for analysis in a sequential order. The choice of number of boxes is arrived by experimentation. By considering the bottom left corner as the absolute origin (0,0), the vector distance for the k^{th} pixel in the b^{th} box at location (i,j) is computed using the formula: $d_{kb} = (i^2 + j^2)^{1/2}$ (1)

By dividing the sum of distances of all black pixels present in a box with their total number, a Normalized Vector Distance (γ_b) for each box is obtained as:

$$\gamma_b = \frac{1}{n_b}\sum_{k=1}^{n_b} d_k^b, \quad b=1, 2, \ldots, 24 \qquad (2)$$

where, n_b is number of pixels in b^{th} box.

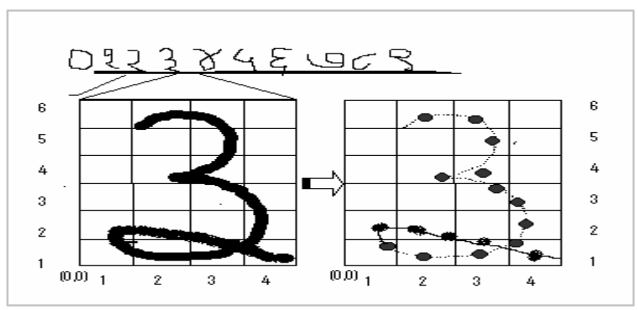


Figure 1. Segmentation of numeral '3' using Box Method

These vector distances constitute a set of features based on distances. Therefore, 24 γ_b's corresponding to 24 boxes will constitute a feature set. However, for empty boxes, the feature value is taken as zero.

4. RECOGNITION SCHEME

In order to recognize the unknown numeral set using fuzzy logic, an exponential variant of fuzzy membership function is selected. The fuzzy membership function is constructed using the normalized vector distance. The concept of a fuzzy set arising from a set of features is as follows. If there are 'n' possible features for each numeral and if there are 'm' such samples, then a particular feature from each of the samples forms a fuzzy set. The means and variances are computed for each of the 24 fuzzy sets and these constitute the knowledge base (KB). Here, we use the training dataset which contains reference numerals for generating the KB.

4.1 Creation of KB and formulation of fuzzy membership function

The means m_i and variance σ^2_i for each of the 24 fuzzy sets of KB are computed from the formulae:

$$m_i = \frac{1}{n}\sum_{j=1}^{N_i} f_{ij} \tag{3}$$

$$\sigma_i^2 = \frac{1}{N}\sum_{j=1}^{N_i} (f_{ij} - m_i)^2 \tag{4}$$

where, N_i is the number of samples in the i^{th} set and f_{ij} stands for the j^{th} feature value of reference numeral in the i^{th} fuzzy set where (i=1,2…24).

For an unknown input numeral x, the 24 features are extracted using the Box method. The membership function is chosen as,

$$\mu_{x_i} = e^{\frac{-|x_i - m_i|}{\sigma_i^2}} \tag{5}$$

where, x_i is the i^{th} feature of the unknown numeral.

If all x_i's are close to m_i's which represent the known statistics of a reference character, then the unknown numeral is identified with this known numeral because all membership function values are close to 1 and hence the average membership function is almost 1. Let, $m_j(r)$, $\sigma^2_j(r)$ belong to the r^{th} reference numeral with r = 0,1…9, we then calculate the average membership as,

$$\mu_{av}(r) = \frac{1}{c}\sum_{j=1}^{c} e^{\frac{-|x_j - m_j(r)|}{\sigma_j^2(r)}} \tag{6}$$

where, c denotes for the number of fuzzy sets. Then $x \in r$ if $\mu_{av}(r)$ is the maximum for r=0,1…9.

It is observed that some of the fuzzy sets have a very small variance and others, a large variance. This spurred the choice of a new membership function in [14] involving the structural parameters s and t given by,

$$\mu_{xi} = e^{-\Delta x'_i / \sigma_i^{2'}} \tag{7}$$

where,

$$\sigma_i^{2'} = (1+t) + t^2\sigma_i^2$$

$$\Delta x_i' = \left|(1-s) + s^2 \Delta x_i\right|$$

$$\Delta x_i = |x_i - m_i|$$

The new mean and the new variance are functions of the mean and variance of the reference fuzzy set. Thus the structural parameters- s, t model the variations in the mean and variance overall 24 boxes. The choice of these parameters has reasoning. That is, if s=1, $\Delta x_i' = \Delta x_i$. Thus, s would be perturbed around 1 to reflect changes in the means. Similarly, if t=-1, then $\sigma_i^{2'} = \sigma_i^2$ thus, t would reflect the changes in the variances.

4.2 Input Fuzzy Modeling

We propose a new input fuzzy modeling method. In this scheme, we treat all fuzzy sets, which are formed from the same feature, as granules. The features considered are the distance features. Each numeral is therefore constituted by 24 granules. In the new input fuzzy model, the need for the consequent part in a fuzzy rule is eliminated. We consider all the fuzzy sets formed out of the same feature as granules. A simple analogy is like flowers (fuzzy sets) strung on a common thread (feature) in a garland (a rule). A fuzzy rule for each numeral is defined as:

IF x is the feature and $x_1 \in A_1, .., x_c \in A_c$ are the granules
DEVISE a Criterion function

(8)

Here 'x' represents the distance feature. The criterion function must be framed to contain the input information. In our earlier work [20,21], we had chosen the following criterion function, G consisting of the Entropy function, E subject to the constraint that the average membership function of all fuzzy sets must be unity :

$$G = E + \lambda J_1 \tag{9}$$

where,

$$E = \frac{1}{C\ln 2}\sum -\left[\mu_{xj}\ln \mu_{xj} + (1-\mu_{xj})\ln(1-\mu_{xj})\right]$$

$$J_1 = [1-J]^2 \text{ and } J = \tfrac{1}{C}\sum_{j=1}^{C}\mu_{xj}$$

We will now eliminate the Lagrangian multiplier, λ in (9) by defining the new criterion function as follows:

$$G = \frac{\sum_{i=1}^{c} -\left[\mu_{xi}\log\mu_{xi} + (1-\mu_{xi})\log(1-\mu_{xi})\right]}{\sum_{i=1}^{c}\mu_{xi}^2} \tag{10}$$

The numerator contains the Shannon entropy, which represents the total uncertainty in all the input fuzzy sets whereas the denominator contains the energy of the fuzzy sets. Taking the inverse of the energy of the fuzzy is a major deviation from the existing fuzzy models. Thus the model in (10) is vested with so much flexibility in the choice of representation of fuzzy sets; which is just unthinkable to derive from the usual fuzzy models. As can be seen from (10), we have removed the restriction in (9). The membership function in (10) is computed using (7).

We have gone one step further from our earlier work [14] to devise this new concept of fuzzy modeling. If there is more than one feature, we need an additional fuzzy rule for each new feature considered. We can have different criterion functions for the same rule. This is a case of multi-classifier problem by which improved recognition rated can be obtained by aggregating the individual recognition rates computed with separate criterion function.

The fuzzy rule in (8) and the criterion function in (10) open new vistas for application involving only the input information. Here, we will not discuss the differences with the Takagi-Sugeno model which has not been found suitable for document processing. This is because this model does not incorporate statistical information in the model through the membership functions unlike our model. Alternatively, we could also take the criterion function as:

$$G = E.J_1 \tag{11}$$

But this still contains the restriction of (7). The advantage with this form is that it is amenable to be extended to the case where the output is available. The discussion of this issue is beyond the scope of this paper. Note that the criterion function in (10) has to be minimized to obtain the structural parameters. That is while matching, the reference numeral with the minimum value of G gives the identity to the unknown numeral. But if we take the energy in the numerator multiplied by the normalized entropy as defined in (12) below, we have to maximize this criterion function for estimating the structural parameters:

$$G = \frac{\sum_{i=1}^{c} -\mu_{xi}^2 \left[\mu_{xi}\log\mu_{xi} + (1-\mu_{xi})\log(1-\mu_{xi})\right]}{\sum_{i=1}^{c} \left[\mu_{xi}\log\mu_{xi} + (1-\mu_{xi})\log(1-\mu_{xi})\right]} \tag{12}$$

The above is in fact the following:

$$G = \sum \textit{Energy} \text{ x } \textit{Normalized-Entropy} \tag{13}$$

Here the maximum value of G of the reference numeral gives the identity to the unknown numeral. In this work, all three criterion functions (10) to (12) are implemented and they give the almost the same recognition rates, but we report results for (10) only. From these criterion functions, it turns out that the definitions revolve around the energy and entropy of the fuzzy sets or the so called granules. Now in order to find the parameters s and t, we use the gradient descent technique,

$$s^{new} = s^{old} - \varepsilon \frac{\partial G}{\partial s}$$
$$t^{new} = t^{old} - \varepsilon \frac{\partial G}{\partial t} \tag{14}$$

The derivatives of G with respect to s, t are:

$$\frac{\partial G}{\partial s} = \frac{(-1)\sum_{i=1}^{c}\left(\log\left(\frac{\mu_{xi}}{1-\mu_{xi}}\right) sum - 2\mu_{xi} su\right)\frac{\partial \mu_{xi}}{\partial s}}{sum^2} \tag{15}$$

$$\frac{\partial G}{\partial t} = \frac{(-1)\sum_{i=1}^{c}\left(\log\left(\frac{\mu_{xi}}{1-\mu_{xi}}\right) sum - 2\mu_{xi} su\right)\frac{\partial \mu_{xi}}{\partial t}}{sum^2} \tag{16}$$

where,

$$su = \sum_{i=1}^{c} (\mu_{xi}\log\mu_{xi} + (1-\mu_{xi})\log(1-\mu_{xi}))$$

$$sum = \sum_{i=1}^{c} \mu_{xi}^2$$

The derivatives of μ_{xi} with respect to s and t are:

$$\frac{\partial \mu_{xi}}{\partial s} = \frac{\mu_{xi}(|1-s+s^2\,|x_i-m_i\,||)(1-2s\,|x_i-m_i|)}{(1+t+t^2\sigma_j^2)(1-s+s^2\,|x_i-m_i|)} \tag{17}$$

$$\frac{\partial \mu_{xi}}{\partial t} = \frac{\mu_{xi}(|1-s+s^2\,|x_i-m_i\,||)(1+2t\sigma_j^2)}{(1+t+t^2\sigma_j^2)^2} \tag{18}$$

The initial values of s and t are taken as 3 and 5 respectively. We will now discuss how to select ε by reinforced learning as proposed in [20].

4.3 Reinforcement Learning

Reusing a defined past policy requires integrating the knowledge of the past policy into the current learning process. Our approach is to bias the exploratory process of the new policy with the past one. In our previous work [16], ε was taken as a constant. In this work the policy reusage concept is used to derive new learning. Instead of taking ε as constant, we make it a variable depending on the past errors. We have used here the sigmoid function for ε in which the cumulative of the past errors is biased by the term k_2 and the slope or gain of the function is changed by the term k_1.

$$\varepsilon = \frac{1}{1+e^{-(k_1\sum err + k_2)}} \tag{19}$$

where,

$$err = J_1^{old} - J_1.$$

In our work, the initial values of k_1 and k_2 are taken both as 0.5. The values of k_1 and k_2 are adjusted as follows:

Algorithm:

1. If $\sum err$ is increasing, then ε must decrease. So k_1 should increase.
2. If $\sum err$ is decreasing, then ε need not change. So k_2 should increase.
3. If $\sum err$ is constant, then ε should not change. So k_1 and k_2 are not changed.

Using this algorithm, the values of k_1 and k_2 have been changed with an increment of 0.1 leading to the convergence of ε as shown in Fig.5. A Genetic algorithm was also implemented to adapt the parameters. But the improvement in the recognition rates is marginal at the cost of enormous computation.

5. RESULTS

We have applied the above recognition strategy on Hindi numerals; a snapshot of numerals is shown in Fig.2. We implemented the proposed recognition method with a variable learning factor ε determined from the reinforcement algorithm. The convergence of structural parameters s and t for constant ε is shown in Fig. 3 and that of variable ε is shown in Fig. 4. Clearly we can see that there is a 25 fold improvement in the speed of convergence during training of λ , s and t. The number of iterations to converge is 601.

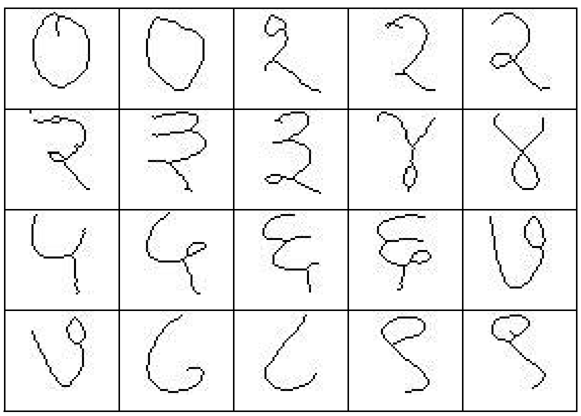


Figure 2. Snapshot of some Database Samples

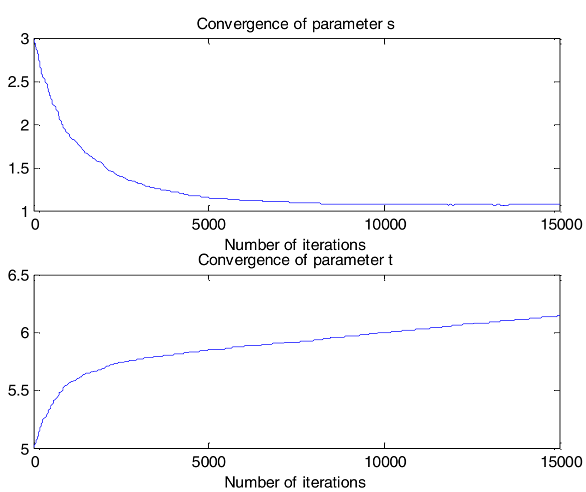


Figure 3. The Case where ε is constant

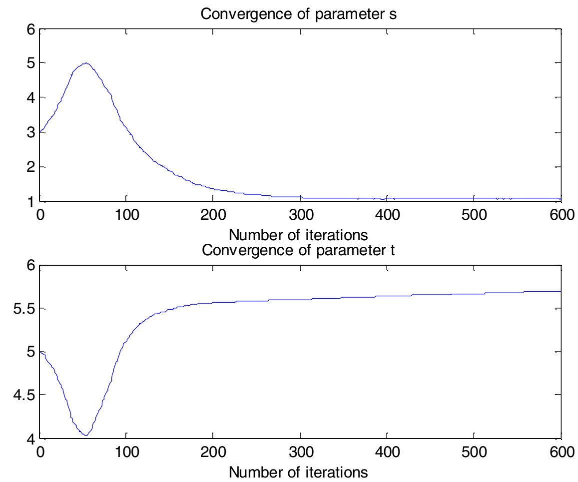


Figure 4. The case where ε is variable

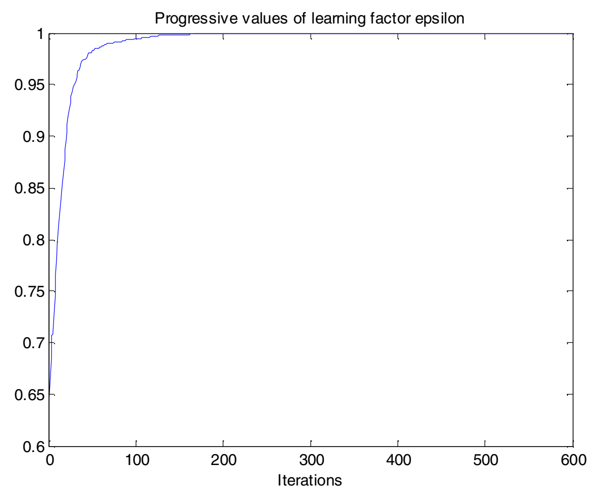
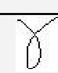


Figure 5. Convergence of ε

Table 1: Parameters & RR of Hindi Numerals

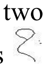
Hindi Numeral	k_1	k_2	*S*	*t*	RR (%)
	31.5	29.5	1.09	10.2	96
	36.5	24.5	1.10	10.54	98
	32.2	28.8	1.04	10.69	90
	37.2	23.8	1.03	10.89	90
	37.5	23.5	1.10	10.89	96
	39.7	21.3	1.09	10.71	100

	39.6	21.4	1.14	10.79	94
	40.4	20.6	1.18	10.61	96
	32.6	28.4	1.00	10.41	96
	31.0	30.0	1.06	10.46	94
Overall recognition rate					**95**

Based on the aspect ratio, we have chosen two windows of sizes: 60x32 to accommodate numerals  and  with an aspect ratio of 1.5 and 42x32 to accommodate the rest of the numerals with an aspect ratio of 1.0. Then window of size 42x32 is divided into 24 boxes of size 7x8 and 60x32 box is divided into 24 boxes of size 10x8. Table 1 shows learning parameters, structural parameters and recognition rates of all Hindi numerals. The overall recognition rate is found to be 95%.

6. CONCLUSIONS

The normalized distance is used as a feature that is found to be effective. A modified membership function is used to improve the recognition. The structural parameters used in the membership function are able to capture the variation in the numerals due to different writing styles. The new input fuzzy modeling making use of a criterion function achieves good recognition rates. The formulation of the relationship between entropy and energy of the fuzzy sets in a criterion function is the major contribution of this paper. This concept will pave the way for new direction in document processing for it simplifies the recognition process.

The aspect ratio of sample image is used in deciding two window sizes for two sets of numerals: 60x32 to accommodate numerals  and  and 42x32 to accommodate the rest of the numerals. We have applied reinforcement learning in training the structural parameters and have seen a 25-fold improvement in the speed of convergence. In document processing where processing time is a major factor, this learning is immensely helpful.

REFERENCES

[1] S. Khedekar, V. Ramanaprasad S. Setlur and V. Govindaraju, "Text - Image Separation in Devanagari Documents", Proc. *Seventh Intl. Conf. on Document Analysis and Recognition*, 2003, pp. 1265-1269.

[2] S.S. Marwah, S.K. Mullick and R.M.K. Sinha, "Character Recognition of Devanagari characters using a hierarchial binary decision tree classifier", Proc. *International Conference on Systems, Man and Cybernetics*, 1984, pp. 414-420.

[3] R. Bajaj, L. Dey and S Chaudhury, "Devnagari numeral recognition by combining decision of multiple connectionist classifiers", *Sadhana*, vol. 27, part 1, 2002, pp. 59-72.

[4] S. Antani and L. Agnihotri, "Gujarati character recognition", Proc. *Fifth International Conference on Document Analysis and Recognition*, 1999, pp. 418–421.

[5] V. Bansal and R.M.K. Sinha, "A Devanagari OCR and a brief review of OCR research for Indian scripts", Proc. *Symposium on Translation Support Systems*, 2001.

[6] B.B. Chaudhuri and U. Pal, "An OCR system to read two Indian language scripts: Bangla and Devnagari (Hindi)", in Proc. *Fourth International Conference on Document Analysis and Recognition*, 1997, vol. 2, pp. 1011–1015.

[7] R.M.K. Sinha and H.N. Mahabala, "Machine recognition of Devanagari script", *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, no. 8, 1979, pp. 435-441.

[8] I.K. Sethi and B. Chatterjee, "Machine recognition of handprinted Devangari Numerals", *Journal of Institution of Electronics and Telecommunication Engineers India*, vol. 22, 1976, pp. 532-535.

[9] I.K. Sethi and B. Chatterjee, "Machine recognition of constrained handprinted Devanagari", *Pattern Recognition*, vol. 9, 1977, pp.69-75.

[10] J.C. Sant and S.K. Mullick, "Handwritten Devanagari Script recognition using CTNNSE algorithm", Proc. *International Conference on Application of Information Technology in South Asia language*, 1994.

[11] M. Hanmandlu, K.R.M. Mohan and H. Kumar, "Neural-based handwritten character recognition", Proc. *Fifth IEEE International Conference on Document Analysis and Recogniton*, 1999, pp. 241-244.

[12] T.M. Ha and H. Bunke, "Offline handwritten numeral recognition by perturbation method", *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 19, issue 5, 1997, pp.535-539.

[13] M. Hanmandlu, K.R.M. Mohan, S. Chakrobarty, S. Goyal and D. Roy Choudhary, "Unconstrained handwritten character recognition based on fuzzy logic", *Pattern Recognition*, vol. 36, issue 3, 2003, pp. 603-623,

[14] M. Hanmandlu, M.H.M. Yusof, V.K. Madasu, "Off-line signature verification and forgery detection using fuzzy modeling", *Pattern Recognition*, vol. 38, issue 3, 2005, pp. 341-356.

[15] R. Stefanelli and A. Rosenfeld, "Some parallel thinning algorithms for digital pictures", *Journal of the ACM*, vol. 18, no. 2, 1971, pp. 255-264

[16] M. Hanmandlu and O.V.R. Murthy "Fuzzy Model Based Recognition of Handwritten Hindi Numerals", Proc. *ICCR 2005*. .

[17] Sutton, R.S. and Barto, A.G. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, Mass., 1998.

[18] Watkins, C.J.C.H. *Learning from Delayed Rewards*. PhD thesis, University of Cambridge, UK, 1989.

[19] F. Fernandez and M. Veloso. *Exploration and policy reuse*. Technical Report CMU-CS-05-172, School of Computer Science, Carnegie Mellon University, 2005.

[20] M. Hanmandlu and O.V.R. Murthy "Reinforcement Learning in the Entropy Based Recognition of Handwritten Hindi Numerals," Proc. 10th World Multi-conference on Systems, Cybernetics and Informatics, 2006.

IEEE COMPUTER SOCIETY