Fuzzy Hough Transform and an MLP with Fuzzy Input/Output for Character Recognition

Abstract

A neuro-fuzzy system for character recognition using a fuzzy Hough transform technique is presented in this paper. For each character pattern, membership values are determined for a number of fuzzy sets defined on the standard Hough transform accumulator cells. These basic fuzzy sets are combined by t-norms to synthesize additional fuzzy sets whose heights form an n-dimensional feature vector for the pattern. A 3n-dimensional fuzzy linguistic vector is generated from the n-dimensional feature vector by defining three linguistic fuzzy sets, namely, weak, moderate and strong. The linguistic set membership functions are derived from the Butterworth polynomials and are similar to the gain functions of low pass, band pass and high pass filters, respectively. A multilayer perceptron (MLP) is trained with the fuzzy linguistic vectors by the back propagation of errors. The MLP outputs represent fuzzy sets denoting similarity of an input feature vector to a number of character pattern classes. Recognition accuracy of the system is more than 98%.

Keywords : Pattern Recognition, Fuzzy Hough Transform, Linguistic Sets, Multilayer Perceptron, Character Recognition

1. INTRODUCTION

Recognition of characters from scanned documents is a key step in document image processing. Many methods have been suggested for solving the problem of optical character recognition (OCR) [4,15]. 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. Multilayer perceptron (MLP) and other neural networks are also often used for character recognition after they are trained with a set of standard patterns by supervised learning. Lippmann [13] has presented a comprehensive introduction to neural networks while MLP based character recognition systems have been proposed in [5,7,21]. An advantage of using neural networks is that a high computation rate can be achieved through their massive parallelism. It may be mentioned that human beings are more efficient than computers in handling complex recognition problems including character recognition from document images. Human reasoning is somewhat fuzzy in nature, which enables us to combine even visually degraded features in the brain using the 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 [2,8,10,23]. Thus, to enable an OCR system to recognize characters even from degraded text images, it is felt necessary to incorporate fuzzy feature extraction concepts in a neural network. Our approach combines the robustness of feature extraction with the speed of operation of neural networks in a framework of fuzzy systems.

Hough transform is a method for the extraction of lines and curves from images [1,22]. Fuzzy probabilistic concepts have been introduced by some authors to generalize the basic Hough transform technique [3]. A fuzzy Hough transform method has been presented in [6] where an image point is treated as a fuzzy point. We propose a Hough transform technique in which a number of fuzzy set membership functions are defined on the standard Hough transform accumulators. When applied to the problem of character recognition, these fuzzy sets are combined to generate individual feature elements for each character. A set of such feature elements form a feature vector and the feature vectors from all the standard character patterns are used to train a multilayer perceptron by supervised learning. The expected MLP outputs denote membership values by which an input pattern belongs to a number of fuzzy sets representing similarity to the different character pattern classes. The pattern corresponding to the highest value MLP output is normally selected as the detected character during recognition. Due to degradation of the image by noise, more than one MLP output may have high fuzzy set membership values. A dictionary search for ascertaining the correct choice is made with these possible characters only. The search is thus confined to a small subset, which greatly reduces the search time.

This paper is structured into sections. The proposed fuzzy Hough transform method for feature extraction is described in the next section. The MLP with fuzzy input and output is covered in section 3. Section 4 presents a two-state Markov chain model of a noisy document image along with the simulation test results. In the last section, we present the implementation results and draw conclusions from our work.

2. FUZZY FEATURE EXTRACTION USING HOUGH TRANSFORM

Hough transform is used to detect lines and curves from an image. For line detection, it uses a parameterization to map an arbitrary straight line in the image plane to a point in the parameter space. A straight line is parametrically represented by the relation $\rho = x\cos\theta + y\sin\theta$, where ρ is its distance from the origin, and θ is the angle of its normal with the x-axis. Hough transform, in effect, maps the problem of finding collinear points in the image plane into one of detecting concurrent curves in the (ρ-θ) plane. To determine the concurrent curves, the ρ-θ plane is quantized into quadruled grids forming a two-dimensional array of accumulators. For each black pixel point in the input image, ρ values are computed for all the quantized values of θ. Accumulator cell count is incremented for each ρ-θ combination so obtained. When all the black pixels are transformed and the accumulator array updated, the count of a given cell in the two-dimensional accumulator holds the total number of curves through the ρ-θ values represented by it. If the count is greater than a threshold value, a line is detected in the input pattern.

An important observation on Hough transform is that it provides three important characteristics of a straight line in an image. These are the values of ρ, θ and count of a ρ-θ accumulator cell. ρ and θ specify the position and orientation of a straight line, while 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.

The first step of fuzzy feature extraction is to map all the black pixels in an input character pattern to the ρ-θ plane. The ρ-θ accumulator cells are accordingly updated. The membership values for the fuzzy sets of table I are then determined for each accumulator cell. Based on the fuzzy sets defined on the Hough transform accumulator, a number of fuzzy sets are next synthesized using t-norms to represent each line in the image as a combination of its length, position and orientation. A number of t-norms are available as fuzzy intersections of which we use the standard intersection: i(p,q) = min(p,q). Similar basic fuzzy sets and synthesized fuzzy sets are defined on the (a,b,c) accumulator cells for circle extraction using the transform $c = \sqrt{(x-a)^2 + (y-b)^2}$ where (a,b) denotes the centre of a circle and c, its radius. The fuzzy sets defined on (a,b,c) accumulators for circle extraction are listed in table II. The synthesized sets for both line and circle extraction are defined in table III. For other pattern recognition problems, suitable fuzzy sets may be similarly synthesized from the basic sets of fuzzy Hough transform.

From the fuzzy set membership function definitions, it is seen that a non-null support of a synthesized fuzzy set implies the presence of the corresponding feature in a character 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 the character. Thus, if $\overline{F}_1, \overline{F}_2, \ldots, \overline{F}_P$ denote P feature vectors derived from P input characters using fuzzy Hough transform, the feature elements of each pattern $\overline{F_i}$ are defined as $F_{i1} = h(LSL) = \underset{(\rho\text{-}\theta)}{\mathrm{Sup}}\ LSL$, $F_{i2} = h(SSL) = \underset{(\rho\text{-}\theta)}{\mathrm{Sup}}\ SSL$, etc, where Sup denotes Supremum of a fuzzy set. The feature vectors extracted from all the character patterns are used to form the input of a classifier. We use a multilayer perceptron as the classifier in our system as described in the next section.

3. MULTILAYER PERCEPTRON WITH FUZZY INPUT AND OUTPUT

A number of fuzzy perceptrons and other fuzzy neural networks have been described in the literature [9,12,17,18]. We use a multilayer perceptron with fuzzy feature vectors as inputs and fuzzy pattern class memberships as outputs. The MLP is structurally similar to a crisp perceptron with supervised learning [13]. In this section, we discuss the fuzzy MLP input and output as well as the process of fuzzy character recognition with the MLP.

3.1 Fuzzy MLP Input

When a feature vector as defined in section 2, is extracted from a degraded character pattern for recognition, the strength of the features in the vector may vary due to 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 to form the MLP inputs. Different membership functions may be considered for these linguistic sets. A π membership function has been used in [16] to generate linguistic sets for MLP inputs. In our system, we derive the linguistic set membership functions from the Butterworth polynomials for low pass, band pass and high pass filter transfer functions [14] as shown below.

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

Here, a, a_1 and a_2 determine the cut-off points and m controls the slope of the membership functions. A processing step before the actual MLP input layer converts each feature element into membership values in these linguistic sets, generating a 3n-dimensional fuzzy feature vector from the n-dimensional feature vector. These 3n-dimensional vectors form the input pattern for the MLP. The advantage of using linguistic features is that, for small variations in the feature values, the linguistic set membership remains unchanged. The system can thus recognize even degraded character patterns.

3.2 Fuzzy MLP Output

In a conventional MLP, an input pattern belongs only to a particular output pattern class. We, however, use the outputs to represent fuzzy pattern classes and the MLP is trained to learn the degree by which an input linguistic feature vector belongs to each of these classes. The output pattern classes are defined as "similar to character 'A' ", "similar to character 'B' ", etc and represented by C_1, C_2, ..., etc. When the MLP is trained with sample patterns, the expected outputs corresponding to each input pattern is computed based on a distance measure between the input feature vector and the feature vector of the character represented by the particular output unit. The membership function denoting belongingness to the different character pattern classes for an input pattern is determined as explained below.

Consider a P-class problem domain with P nodes in the output layer of the MLP. Each pattern, before converting to linguistic sets, is represented by the feature vector $\overline{F}_i$, i = 1,2,..., P. The Euclidean distance between $\overline{F}_i$ and other feature vectors is calculated as follows.

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

We use eq. (1) to calculate the distances of all the P patterns from the i^{th} input pattern where the summation is done over all the feature elements subscripted by j. The degree of membership of the i^{th} input character pattern to the k^{th} fuzzy pattern class C_k, and hence the value of the k^{th} expected output of the MLP for the input vector $\overline{F}_i$ is 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 the parameters f_{den} and f_{pow} control the degree of membership by which the pattern $\overline{F}_i$ belongs to the different output fuzzy sets. It is seen that, $\mu_k(\overline{F}_i) \in [0,1]$, $\mu_k(\overline{F}_i) = \mu_i(\overline{F}_k)$, $\mu_k(\overline{F}_k) = 1$, and $d_{ik} \geq d_{il} \Rightarrow \mu_k(\overline{F}_i) \leq \mu_l(\overline{F}_i)$. 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. Distance measures other than the Euclidean distance may also be considered in a similar manner. The MLP is trained with the input fuzzy feature vectors and fuzzy expected outputs by the back propagation algorithm [19].

A block diagram of the complete fuzzy character recognition system is shown in figure 1. During recognition only the path marked with bold lines is followed. The dotted path is additionally required in the learning phase for generating expected outputs, error calculation and weight updation by error back propagation.

3.3 Fuzzy Character Recognition

As discussed above, each MLP output represents the degree of membership by which an input pattern belongs to the corresponding fuzzy character pattern class. The recognition decision is based on the α-cuts of the output fuzzy sets for $\alpha = \tau$, 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 τ is low, α-cuts of the output fuzzy sets may contain more than one element while a high value of τ 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. However, due to the presence of noise, all the outputs may fall below the threshold or more than one output may go above the threshold, depending on the values of τ, f_{den} and f_{pow}. If the membership value is above the threshold for more than one output, it indicates a possibility of *misclassification*. We then consider all these high value outputs for a dictionary search to ascertain the character. In the dictionary search step, a lexicon is consulted to correctly determine the character using word level knowledge. The advantage of using fuzzy set membership functions at the MLP output is that the search space is considerably shortened and only the outputs with high membership values are considered for the search. Since the MLP outputs denote their closeness with the correct pattern class, this decision making process is justified. In the third possibility, if all the MLP outputs fall below the threshold τ for an input pattern, it indicates a *classifier failure*. The dictionary search is then made with a wildcard at the unknown character position, i.e., the search space includes all the pattern classes, which increases the search time. If more than one choice of possible characters form a valid word in the dictionary, the characters are considered to be unresolved and are marked for identification by user intervention.

The values of τ, f_{den} and f_{pow} are chosen by simulation in such a way that each of the α-cuts contains more than one element and the combined error due to *misclassification* and *classifier failure* is minimized. The noise model used for simulation along with the simulation results is discussed in the next section.

4. NOISE MODEL AND SIMULATION RESULTS

Noise in a document image occurs in the form of bit reversal which can be random or burst in nature. We create such noisy documents following the Gilbert-Elliott model for communication channel noise [20]. A noisy document image is modeled as a two-state Markov chain 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’. Here r << b. The state transition probabilities are ‘q’ and ‘Q’ where q is the conditional probability that the image remains in the Random state for the next pixel position, given that it is in that state for the current pixel. With probability (1-q), it makes a transition to the Burst state. Q is also defined similarly for the Burst state. The steady state probabilities of the random state and the burst state, P_R and P_B are as follows.

$$P_R = \frac{1-Q}{2-Q-q}, \quad P_B = \frac{1-q}{2-Q-q} \tag{3}$$

The average pixel error probability P_e on the document is $P_e = bP_B + rP_R$. In channel models, a Burst is defined as a sequence in which contiguous transmitted bits have a higher probability of error. However in a document image, which is inherently two-dimensional in nature, we define a Burst start as an event when pixels lying on the k-neighbors of the current pixel are affected with higher error probability, for different values of ‘k’. Thus, propagation of burst is spatial in a noisy document image unlike communication channels where it is temporal in nature. The extent of a Burst around a pixel is controlled by setting the parameter k. The error density ratio, $\Delta = \frac{b}{r}$, is an indicator of the severity of the bursts in the image. The average burst length, λ is defined as the average number of pixels for which the image remains in the burst state and is given by $\lambda = \frac{Q}{1-Q}$.

Extensive simulation has been done using the model to determine the dependence of the OCR recognition performance on the parameter m of the linguistic set definitions and the parameter f_{pow} of the MLP output fuzzy set as well as on the noise parameters λ and Δ. The ratio (d_{ik}/f_{den}) is kept less than unity. It has been observed that *misclassification* error increases while recognition error due to *classifier failure* goes down drastically with increasing values of m. This is because, for higher values of m, the inputs tend to look similar to the MLP. Keeping the value of m constant, if f_{pow} is decreased, *misclassification* is reduced while error due to *classifier failure* increases. The reason is that, for lower values of f_{pow}, the membership values $\mu_k(\overline{F}_i)$ of eq. (2) are reduced except when i = k. Thus, decreasing the value of f_{pow} has the effect of enhancing the ‘contrast’ in the output pattern classes, while increasing the value of m has the effect of reducing the contrast among the input patterns.

The requirement of an OCR system to correctly differentiate between similar looking patterns (thus reducing *misclassification* error), and to correctly recognize a pattern in the presence of noise (thus reducing *classifier failure*), are mutually contradictory in nature. We, therefore, choose the values of m and f_{pow} from the simulation results so that the total error of the fuzzy OCR system is minimized. Since, this set of simulations is done to test the sensitivity of the classifier to the input and output parameters, we have not used dictionary search to aid the recognition process in this case.

The ability of the fuzzy OCR system to correctly identify a character through dictionary search depends on the noise distribution in the document image. In the second set of simulations we have used dictionary search to determine the variation of the overall OCR error with the noise parameters λ and Δ. It has been observed that for small values of λ, the recognition error percentage is higher 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 short length, affecting more than one character position in a word. The number of recognized characters in a word is, therefore less, resulting in ambiguous choice from the dictionary search. As λ increases, the number of bursts goes down and the affected characters are identified by the dictionary search, reducing the overall recognition error. Recognized characters in their positions within the word and unrecognized possibilities in their positions are considered for this search.

It has also been observed that for small values of Δ, unresolved error is higher. The error percentage then decreases and remains almost constant for higher values of Δ. For small values of Δ, error occurs with almost equal probability in both the random state and the burst state, increasing the number of unresolved errors. For higher values of Δ, error due to random noise is corrected by the fuzzy MLP itself while the errors caused by dense bursts are resolved in the dictionary search step.

5. IMPLEMENTATION RESULTS AND CONCLUSIONS

The fuzzy OCR system has been implemented for character recognition from a large number of printed documents of different types and quality. A 300-dpi HP scanner is used to scan the documents and generate bitmap images. The input and output parameters of the MLP are selected based on the simulation test results. The linguistic set parameter m and the MLP output parameter f_{pow} are typically chosen as 4 and 0.87, respectively. As mentioned before, the ratio (d_{ik}/f_{den}) is kept less than unity. The character recognition efficiency of the system is more than 98% for single font documents when the system is trained with a particular font. When trained with more than one font, the recognition efficiency of the system is in the range of 96-98% for multi-font documents.

The advantage of fuzzy sets is harnessed in two stages in the proposed fuzzy character recognition system. First, the fuzzy Hough transform does not reject any feature since thresholding is not done on the Hough transform accumulators cells. All the pattern features are, therefore, retained for decision making at a higher level. Secondly, the output fuzzy sets of the MLP enable the system to consider characters which are suitable candidates for final selection from word level knowledge. Here also, we do not use the usual *winner-take-all* logic of a crisp perceptron.

The fuzzy Hough transform technique proposed here is useful for a variety of pattern recognition problems other than character recognition. Fuzzy features like *thick* lines, *thin* lines, *nearly parallel* lines, lines *slightly above* or *slightly below* fixed lines similar to those presented in [11] may also be extracted from document as well as non-document images using the proposed method. The present work can be extended to include fuzzy rule based systems to combine features extracted by Fuzzy Hough transform instead of using a fuzzy MLP.

Acknowledgment

The authors would like to thank Dr. N.R.Pal of the Indian Statistical Institute, Calcutta for his valuable suggestions and interest in the work.

References

[1] D.H.Ballard, Generalizing the Hough transform to detect arbitrary shapes, Pattern Recognition 13 (1981) 111-122.

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

[3] S.M.Bhandarkar, A fuzzy probabilistic model for the generalized Hough transform, IEEE Trans. on SMC 24 (1994) 745-759.

[4] V.K.Govindan and A.P.Shivaprasad, Character recognition - A survey, Pattern Recognition 23 (1990) 671-683.

[5] I.A.Hadar, T.A.Diep and H.Garland, High accuracy OCR using neural network with centroid dithering, IEEE Trans. on PAMI 17 (1995) 218-224.

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

[7] B.Hussain and M.R.Kabuka, A novel feature recognition neural network and its application to character recognition, IEEE Trans. on PAMI 16 (1994) 98-106.

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

[9] J.M.Keller and D.J.Hunt, Incorporating fuzzy membership functions into the perceptron algorithm, IEEE Trans. on PAMI 7 (1985) 693-699.

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

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

[12] H.K.Kwan and Y.Cai., A fuzzy neural network and its application to pattern recognition, IEEE Trans. on Fuzzy Systems 2 (1994) 185-202.

[13] R.P.Lippmann, An introduction to computing with neural nets, IEEE ASSP magazine (1987) 4-22.

[14] J.Millman and C.C.Halkias, Integrated Electronics : Analog and Digital Circuits and Systems (McGrawhill, Singapore, 1972)

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

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

[17] W. Pedrycz, Neurocomputations in relational systems, IEEE Trans. on PAMI 13 (1991) 289-297.

[18] W.Pedrycz and A.F.Rocha, Fuzzy set based models of neurons and knowledge based networks, IEEE Trans. on Fuzzy Systems 1 (1993) 254-266.

[19] D.E.Rumelhart, G.E.Hinton and R.J.Williams, 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, 1986).

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

[21] S.Sural and P.K.Das, A document image analysis system on parallel processors, Proc. International Conf. on High Performance Computing (1997) 527-532.

[22] S.Y.Yuen and C.H.Ma, An investigation of the nature of parameterization for the Hough transform, Pattern Recognition 30 (1997) 1009-1040.

[23] H.-J.Zimmermann, Fuzzy set theory and its applications, Second edition (Kluwer Academic publishers, Norwell, Massachusetts, 1991).

Table I. Fuzzy set membership functions defined on Hough transform accumulator cells for line detection from a pattern of height X and width Y.

Fuzzy Set	Membership Function	Notation
Long line	$\left(\frac{\text{count}}{\sqrt{X^2+Y^2}}\right)$	LL
Short line	2LL if count $\leq \frac{\sqrt{X^2+Y^2}}{2}$ 2(1-LL) if count $> \frac{\sqrt{X^2+Y^2}}{2}$	SL
Nearly horizontal line	$\left(\frac{\theta}{90.0}\right)$	HL
Nearly vertical line	1-HL	VL
Slant line	2HL if $\theta \leq 45.0$ 2(1-HL) if $\theta > 45.0$	TL
Line near top border	$\left(\frac{\rho}{X}\right)$ if HL $>$ VL 0 otherwise	NT
Line near bottom border	1-NT if HL $>$ VL 0 otherwise	NB
Line near vertical centre	2NT if (HL $>$ VL and $\rho \leq \frac{X}{2}$) 2(1-NT) if (HL $>$ VL and $\rho > \frac{X}{2}$) 0 otherwise	NVC
Line near right border	$\left(\frac{\rho}{Y}\right)$ if VL $>$ HL 0 otherwise	NR
Line near left border	1-NR if VL $>$ HL 0 otherwise	NL
Line near horizontal centre	2NR if (VL $>$ HL and $\rho \leq \frac{Y}{2}$) 2(1-NR) if (VL $>$ HL and $\rho > \frac{Y}{2}$) 0 otherwise	NHC

Table II. Fuzzy set membership functions defined on Hough transform accumulator cells for circle detection from a pattern of height X and width Y.

Fuzzy Set	**Membership Function**		**Notation**
Large circle	$\left(\frac{c}{X/2}\right)$		LC
Small circle	2LC 2(1-LC)	if $c \leq (X/4)$ if $c > (X/4)$	SC
Centre near right border	$\left(\frac{a}{Y}\right)$		CRB
Centre near left border	1-CRB		CLB
Centre near horizontal mid-point	2CRB 2(1-CRB)	if $a < (Y/2)$ otherwise	CHM
Centre near top border	$\left(\frac{b}{X}\right)$		CTB
Centre near bottom border	1-CTB		CBB
Centre near vertical mid-point	2CTB 2(1-CTB)	if $b < (X/2)$ otherwise	CVM
Centre near mid-point	(2CHM)CVM		CMP
Dense circle	$\left(\frac{\text{count}}{2\pi c}\right)$		DC
Sparse circle	2DC 2(1-DC)	if Count $\leq \pi c$ Otherwise	PC

Table III. Synthesized fuzzy set definitions using t-norms.

Synthesized Fuzzy Set	**Definition** ($i \equiv$ **t-norm)**	**Notation**
Long slant line	i(TL,LL)	LSL
Short slant line	i(TL,SL)	SSL
Nearly horizontal short line near vertical centre	i(HL,SL,NVC)	HSVC
Nearly vertical long line near left border	i(VL,LL,NL)	VLL
Nearly vertical long line near right border	i(VL,LL,NR)	VLR
Nearly horizontal long line near top border	i(HL,LL,NT)	HLT
Nearly horizontal long line near bottom border	i(HL,LL,NB)	HLB
Nearly vertical long line near horizontal centre	i(VL,LL,NHC)	VLHC
Nearly vertical short line near horizontal centre	i(VL,SL,NHC)	VSHC
Large dense circle with centre near mid-point	i(LC,DC,CMP)	LDM
Large sparse circle with centre near mid-point	i(LC,PC,CMP)	LPM
Large sparse circle with centre near bottom border on horizontal mid-point	i(LC,PC,CBB,CHM)	LPBM
Small sparse circle with centre near left border on vertical mid-point	i(SC,PC,CLB,CVM)	SPLM
Small dense circle with centre near top border on horizontal mid-point	i(SC,DC,CTB,CHM)	SDTM
Small sparse circle with centre near top left border	i(SC,PC,CTB,CLB)	SPTL
Small sparse circle with centre near top right border	i(SC,PC,CTB,CRB)	SPTR
Small sparse circle with centre near bottom border on horizontal mid-point	i(SC,PC,CBB,CHM)	SPBM
Small sparse circle with centre near mid-point	i(SC,PC,CMP)	SPM
Small dense circle with centre near mid-point	i(SC,DC,CMP)	SDM

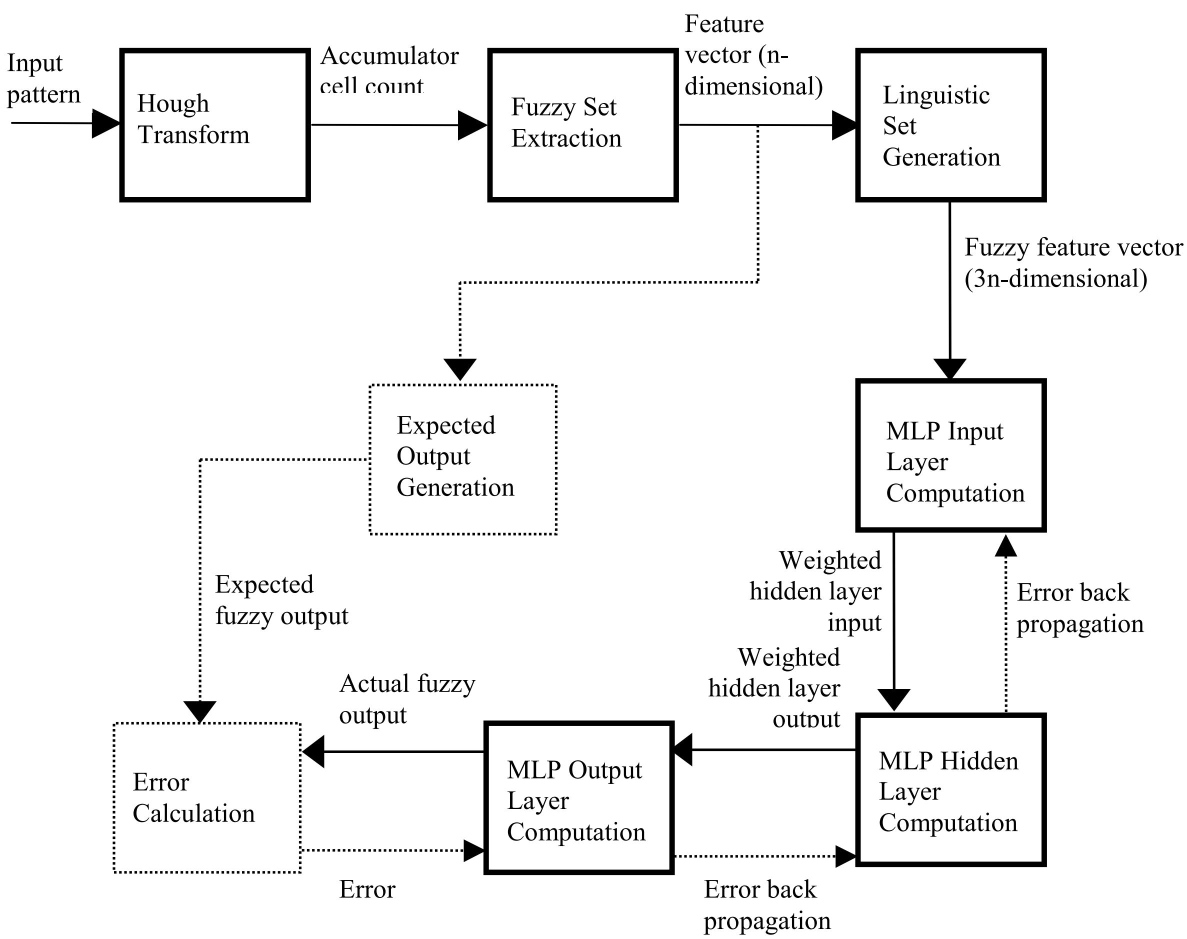
Input
pattern
Hough
Transform
Accumulator
cell count
Fuzzy Set
Extraction
Feature
vector (n-
dimensional)
Linguistic
Set
Generation
Fuzzy feature vector
(3n-dimensional)
Expected
Output
Generation
MLP Input
Layer
Computation
Weighted
hidden layer
input
Error back
propagation
Expected
fuzzy output
Weighted
hidden layer
output
Actual fuzzy
output
Error
Calculation
MLP Output
Layer
Computation
MLP Hidden
Layer
Computation
Error
Error back
propagation

Figure 1

Figure Caption

Figure 1 Complete block diagram of the fuzzy character recognition system