A Soft Computing Approach to Character Recognition

Shamik Sural, NIIT Limited, 6B Pretoria Street, Calcutta 700 071, India

P.K.Das, Department of CSE, Jadavpur University, Calcutta 700 032, India

Abstract

A character recognition system using soft computing techniques is presented in this paper. We define fuzzy sets on the Hough transform of each character pattern pixel and synthesize additional fuzzy sets by t-norms. The heights of these t-norms form an n-dimensional feature vector for the character. A 3n-dimensional vector is then generated from the n-dimensional feature vector by defining three linguistic fuzzy sets, namely, weak, moderate and strong for each feature element. These 3n-dimensional vectors for all the character patterns form a multilayer perceptron (MLP) input for training by the back propagation of errors. The fuzzy feature set is chosen after performing a sensitivity analysis of the multilayer perceptron outputs to the input features by a genetic algorithm. The MLP outputs also represent fuzzy sets denoting the belongingness of each input pattern to a number of fuzzy pattern classes. During recognition, outputs with high fuzzy set membership values are considered for a dictionary-based search to identify the ambiguous characters using word level knowledge. The system has been implemented for character recognition from printed English documents.

Introduction

Optical character recognition (OCR) is one of the most popular areas of research in pattern recognition because of its immense application potential. The two fundamental approaches to OCR are template matching and feature classification. In the template matching approach, recognition is based on the correlation of a test character with a set of stored templates. In the feature classification method, features are extracted from a standard character image to generate a feature vector. A decision tree is formed based on the presence or absence of some of the elements in the feature vector. When an unknown character pattern is encountered, this tree is traversed from node to node till a unique decision is reached. The template matching techniques are more sensitive to font and size variations of the characters than the feature classification methods. However, selection and extraction of useful features is not always straightforward.

Multilayer perceptron (MLP) and other neural networks are often used for the recognition of characters after they are trained with a set of standard patterns by supervised learning. See (Lippmann 1987) for an introduction to neural networks and (Hussain and Kabuka 1994, Hadar, Diep and Garland 1995, Sural and Das, 1997) for neural network based character recognition techniques. It may be mentioned that human reasoning is somewhat fuzzy in nature, which enables us to combine 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. Our soft computing approach to character recognition combines the robustness of feature extraction with the speed of operation of neural networks in a framework of fuzzy systems. The difficulties in feature selection have been overcome by the use of a genetic algorithm.

This paper is organized into a number of sections. We describe the proposed fuzzy feature extraction method using Hough transform in the next section. The operations of a multilayer perceptron with fuzzy input/output is explained in the following section. We then discuss the genetic algorithm based feature selection process. The implementation results and conclusions are given in the last section.

Hough Transform and Fuzzy Feature Extraction

Hough transform is a method for the detection of lines and curves from images (Illingworth and Kittler 1988, Yuen and Ma 1997). Some authors have introduced fuzzy probabilistic concepts to generalize the basic Hough transform technique (Bhandarkar 1994). (Han, Koczy and Poston 1994) present a fuzzy Hough transform technique where the image points are treated as fuzzy points. 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 line in an image pattern. ρ and θ specify the position and orientation of the line, while count of a (ρ,θ) accumulator cell used in Hough transform implementation specifies the number of black pixels lying on it. Keeping this in mind, we define a number of fuzzy sets on the ρ-θ accumulator cells. These fuzzy set definitions are shown in table I for θ values in the first quadrant. The definitions are extended for other values of θ. Two of these fuzzy sets, namely, *Long Line* (LL) and *Short Line* (SL) extract length information of the different lines in a pattern. *Nearly Horizontal* (HL), *Nearly Vertical* (VL) and *Slant Line* (TL) represent their skew while *Near Top* (NT), *Near Bottom* (NB), *Near Vertical Centre* (NVC), *Near Right* (NR), *Near Left* (NL) and *Near Horizontal Centre* (NHC) extract the position information of these lines. Characteristics of the different lines in an image pattern are thus mapped into the properties of these fuzzy sets. For a detailed discussion on fuzzy sets and their properties, readers may refer to (Klir and Yuan 1995).

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

Table I. Fuzzy set membership functions defined on Hough transform accumulator cells for line detection. X and Y denote the height and the width of each character pattern.

Based on the basic fuzzy sets defined above, we synthesize additional fuzzy sets to represent each line in a pattern as a combination of its length, position and orientation using t-norms. The synthesized fuzzy sets are defined as *Long Slant Line* (LSL) ≡ *i*(TL,LL), *Short Slant Line* (SSL) ≡ *i*(TL,SL), *Nearly Vertical Long Line near the Left* (VLL) ≡ *i*(VL,LL,NL), etc., where *i* denotes a t-norm.. Similar basic fuzzy sets (*Large Circle, Dense Circle*, Centre *Near Top*, etc.) and synthesized fuzzy sets (*Small Dense Circle near the Top*, *Large Dense Circle near the Centre*, etc.) are defined on (a,b,c) accumulator cells for circle extraction using the Hough transform c = $\sqrt{(x-a)^2 + (y-b)^2}$. For a circle extraction, (a,b) denotes the origin, c denotes the radius while count specifies the number of pixels lying on the circle.

A number of t-norms are available as fuzzy intersections of which we use the standard intersection : i(p,q) = min(p,q). For other pattern recognition problems, suitable fuzzy sets may be similarly synthesized from the basic sets of fuzzy Hough transform. 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.

MLP with Fuzzy Input/Output

The n-dimensional feature vectors as described above may be used to train a multilayer perceptron. However, when such an n-dimensional feature vector 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. The linguistic set membership functions are derived from the Butterworth filter transfer functions (Millman and Halkias 1972) as shown below.

$$\mu_{\text{weak}}(x) = \left[1+\left(\frac{x}{a}\right)^{2m}\right]^{-\frac{1}{2}} \qquad (1)$$

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

$$\mu_{\text{strong}}(x) = \left[1+\left(\frac{a}{x}\right)^{2m}\right]^{-\frac{1}{2}} \qquad (3)$$

The membership value is 0.7 for x = a (a_1, a_2 for μ_{moderate}), the cut-off point, for all values of m where m controls the slope of the functions. The 3n-dimensional vectors, thus generated, form the MLP input both during training and recognition for each character pattern. 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. (Pal and Mitra 1992) have incorporated similar concepts of fuzzy sets in MLP operation for object classification. They have used a π−function for linguistic set membership generation.

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 Euclidean distance between each input vector $\overline{F_i}$ and other feature vectors is calculated as follows.

$$d_{ik} = \sqrt{\sum_j \left(F_{ij} - F_{kj} \right)^2} \qquad k = 1,2,..., P. \qquad (4)$$

The summation is done over all the feature elements subscripted by j. The membership of the ith character pattern to the kth fuzzy pattern class, and hence the value of the kth expected output of the MLP for the input vector $\overline{F_i}$ is determined using the following relation.

$$O^i_{k(exp)} = \mu_k\left(\overline{F_i}\right) = \left[1 + \left(\frac{d_{ik}}{f_{den}} \right)^{f_{pow}} \right]^{-1} \qquad (5)$$

Here 'f_{den}' and 'f_{pow}' control the membership grades in the different output fuzzy sets for each input pattern. The following properties are satisfied by the fuzzy class membership functions of eq. (5).

i. $\mu_k\left(\overline{F_i}\right) \in [0,1]$

ii. $\mu_k\left(\overline{F_i}\right) = \mu_i\left(\overline{F_k}\right)$

iii. $\mu_k\left(\overline{F_k}\right) = 1$

iv. $d_{ik} \geq d_{il} \Rightarrow \mu_k\left(\overline{F_i}\right) \leq \mu_l\left(\overline{F_i}\right)$

v. For $f_{den} \rightarrow 0$ and $f_{pow} \rightarrow \infty$, the fuzzy MLP output reduces to 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 feature vectors to learn the fuzzy expected outputs by the back propagation algorithm (Rumelhart, Hinton and Williams 1986).

Recognition decision of the MLP is based on the α-cuts of the output fuzzy sets for α = τ, a threshold value for the outputs. For a fixed value of the parameters 'f_{den}' and 'f_{pow}' if τ is low, the α-cuts contain more than one element while a high value of τ results in null α-cuts for some of the outputs. Conversely, in the presence of low level noise in a test character, one of the MLP outputs goes above the threshold and the others take on low values. In this case, the highest value output is considered to be the unknown character. If, however, the membership value is above the threshold for more than one output, indicating a possibility of misclassification, a dictionary search is used 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 elements constituting the feature vector is chosen after performing a sensitivity analysis of the MLP output for different input feature vectors. The feature selection method is explained in the next section.

Feature Selection Using Genetic Algorithm

A number of neural network and fuzzy set theoretic approaches have been proposed for feature analysis (Ruck, Rogers and Kabrisky 1990, Pal 1992, Belue and Bauer 1995). (De, Pal and Pal 1997) have suggested the use of a feature quality index (FQI) for the ranking of features. Their feature ranking process is based on the concept that the influence of a feature on an MLP output is related to the importance of the feature in discriminating among classes. The impact of the qth feature on the MLP output out of a total of 'p' features is measured by setting the feature value to zero for each input pattern $\mathbf{x_i}$, i = 1,2,...,n. FQI is defined as the deviation of the MLP output with qth feature value set to zero from the output with all features present. Thus,

$$FQI_q = \frac{1}{n} \sum_{i=1}^{n} \left\| \mathbf{O_i} - \mathbf{O_i}^{(q)} \right\|^2 \qquad (6)$$

Here $\mathbf{O_i}$ and $\mathbf{O_i}^{(q)}$ are the output vectors with all the 'p' features present and with the qth feature set to zero, respectively. The features are ranked according to their importance as $q_1, q_2, ..., q_p$ if $FQI_{q1} > FQI_{q2} > ... > FQI_{qp}$. In order to select the best $p^/$ features from the set of p features, $\binom{p}{p^/}$ possible subsets are tested, one at a time. The quality index $FQI_k^{(p^/)}$ of the kth subset S_k is measured as :

$$\mathrm{FQI}_k^{(p^{/})} = \frac{1}{n}\sum_{i=1}^{n}\left\|\mathbf{O_i} - \mathbf{O_i}^k\right\|^2 \qquad (7)$$

Here $\mathbf{O_i^k}$ is the MLP output vector with $\mathbf{x_i^k}$ as the input. $\mathbf{x_i^k}$ is derived from $\mathbf{x_i}$ as follows :

$$x_{ij}^{\;k} = \begin{cases} 0 & \text{if } j \in S_k \\ x_{ij} & \text{otherwise} \end{cases} \qquad (8)$$

A subset S_j is selected as the optimal set of features if $\mathrm{FQI}_j^{(p^{/})} \geq \mathrm{FQI}_k^{(p^{/})} \;\; \forall k;\; k \neq j$.

An important observation on this method of feature selection is that the value of $p^{/}$ should be pre-determined and that $\begin{pmatrix} p \\ p^{/} \end{pmatrix}$ number of possible choices are to be verified to arrive at the best feature set. It is evident that no *a priori* knowledge is usually available to select the value of $p^{/}$ and an exhaustive search is to be made for all values of $p^{/}$; $p^{/} = 1,2,..., p$. The number of possible trials then becomes $(2^p - 1)$ which is prohibitively large for high values of p.

To overcome the drawbacks of the above method, we select the best feature set by the use of genetic algorithm (Goldberg 1989). We define a mask vector **M** where $M_i \in \{0,1\}$; i=1,2,...,p and each feature element q_i, i = 1,2,...,p is multiplied by the corresponding mask vector element before reaching the MLP input. The MLP inputs may then be written as :

$$I_i = q_i M_i \,;\; i = 1,2,...,p. \qquad (9)$$

$$= \begin{cases} 0 & \text{if } M_i = 0 \\ q_i & \text{otherwise} \end{cases} \qquad (10)$$

Thus, a particular feature q_i reaches the MLP if the corresponding mask element is one. To find the sensitivity of a particular feature q_j, we have to set the mask bit M_j to zero. In light of the above discussions, when we select the k^th subset of the feature set $\{q_1, q_2, ..., q_p\}$, all the corresponding mask bits are set to zero and the rest are set to one. When the feature set multiplied by these mask bits reaches the MLP we get the effect of setting the features of the subset S_k to zero and calculate the value of $\mathrm{FQI}_{k.}$ Note that the k^th subset thus chosen may contain any number of feature elements and not a pre-specified $p^{/}$ number of elements.

Starting with an initial population of strings representing the mask vectors, we use a genetic algorithm with *reproduction*, *crossover* and *mutation* operators to determine the best value of the objective function. The objective function is the FQI value of the feature set S_k selected with the mask bits set to zero for the selected features and is given by :

$$\mathrm{FQI}_k = \frac{1}{n}\sum_{i=1}^{n}\left\|\mathbf{O_i} - \mathbf{O_i}^k\right\|^2 \qquad (11)$$

In this process, we solve both the problems of pre-determining the value of $p^{/}$ and searching through the $\begin{pmatrix} p \\ p^{/} \end{pmatrix}$ possible combinations for each value of $p^{/}$.

In the genetic algorithm implementation, we start with 20 features generated from the fuzzy Hough transform as explained earlier so that the number of elements in the mask vector is also 20. After running the genetic algorithm for a sufficiently large number of generations, the mask string with the best objective function value is determined. The feature elements corresponding to the mask bits zero are chosen as the selected set of features. The parameters for the genetic algorithms are as follows :

Chromosome Length : 20
Population Size : 20
Mutation Probability : 0.002
Crossover Probability : 0.67

No. of Generations	**Objective Function Value**	**Best Fit String**
10	1.690104	01010100001000100000
20	1.759118	01100101001000100000
50	1.759118	01100101001000100000
100	1.790014	01000101001000000000
200	1.795768	01010101001000000000
300	1.820325	01010101000101000100
400	1.820325	01010101000101000100
500	1.820325	01010101000101000100
600	1.895224	01010001000100000100
700	1.899543	01000001000101000100
800	1.899543	01000001000101000100
900	1.899543	01000001000101000100
1000	1.899543	01000001000101000100
1500	1.899543	01000001000101000100
2000	1.903486	01010001000010000100
2500	1.917685	01010001000001000100
3000	1.917685	01010001000001000100
4000	1.917685	01010001000001000100

Table II. Performance of the genetic algorithm over different generations for the feature selection process.

The performance of the genetic algorithm over different generations is listed in table II. The selected set of features is ($q_1q_3q_5q_6q_7q_9q_{10}q_{11}q_{12}q_{13}q_{15}q_{16}q_{17}q_{19}q_{20}$). The MLP is next trained with only this set of features for classification. The number of features is reduced from 20 to 15 which is a reduction of 25%. The time and resource requirements are, thereby, greatly reduced. A schematic representation of the genetic algorithm based feature selection method is shown in figure 1.

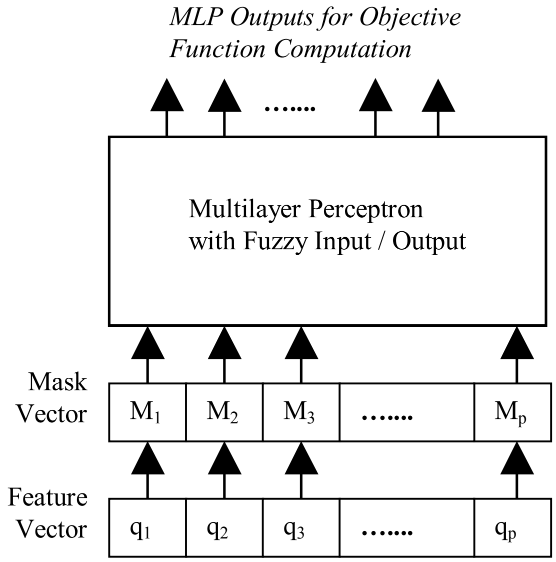

Figure 1. Schematic representation of the feature selection process using genetic algorithm.

Results and Discussions

We have implemented the character recognition system using soft computing techniques for the recognition of characters from scanned English documents. In the implemented system, a 300 dpi HP scanner is used to scan the documents and generate the bitmap images. The linguistic set parameter m and the MLP output parameter 'f_{pow}' are typically chosen as 4 and 0.87, respectively. The ratio d_{ik} / f_{den} is kept less than unity. The system can correctly recognize 98% of the learnt patterns for single font documents. When trained with more than one font, the recognition efficiency of the system is in the range of 96-98% for multifont documents.

The advantage of soft computing is harnessed in three stages in the proposed character recognition system. First, the proposed fuzzy Hough transform method 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 *winner-take-all* logic of crisp perceptrons. Finally, the genetic algorithm for feature selection makes the process fast and elegant.

The linguistic set definitions and the MLP with fuzzy input/output can be used with other fuzzy feature extraction techniques also. Fuzzy features like *thick* lines, *thin* lines, *nearly parallel* lines, lines *slightly above* or *slightly below* fixed lines, similar to those proposed by (Krishnapuram, Keller and Ma 1993) may also be extracted from document as well as non-document images using this technique. The genetic algorithm based feature analysis may be applied to a wide variety of pattern recognition problems including the example problems cited by (De, Pal and Pal 1997). The present work can be extended to include fuzzy rule based systems to combine features extracted by fuzzy Hough transform instead of using the fuzzy MLP.

References

Belue, L.M., and Bauer, J.K.W. 1995. Determining input features for multilayer perceptrons. *Neurocomputing* 7: 111--121.

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

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

Goldberg, D.E. 1989. *Genetic algorithms in search, optimization and machine learning*. Reading, mass.:Addison-Wesley.

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

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

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

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

Klir, G.J., and Yuan, B. 1995. *Fuzzy sets and fuzzy logic*. N.J.USA:Prentice-Hall Inc.

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

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

Millman, J., and Halkias, C.C. 1972. *Integrated Electronics: Analog and Digital Circuits and Systems*. Singapore:McGraw Hill.

Pal, S.K. 1992. Fuzzy set theoretic measures for automatic feature evaluation: II. *Information Sciences* 64:165--179.

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

Ruck, D.W., Rogers, S.K., and Kabrisky, M. 1990. Feature selection using a multilayer perceptron. *Journal of Neural Network Computing*:40--48.

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

Sural, S., and Das, P.K. 1997. A document image analysis system on parallel processors. In *Proceedings of the Fourth International Conference on High Performance Computing*, 527--537. Bangalore, India:IEEE Computer Society Press.

Yuen, S.Y., and Ma, C.H. 1997. An investigation of the nature of parametrization for the Hough transform. *Pattern Recognition* 30:1009--1040.