Extended Fuzzy Hyperline Segment Neural Network for Handwritten Character Recognition

Dipti Pawar, *Member, IAENG*

***Abstract*— This paper deals with simple and effective set of features for character representation. These features are computed within regularly placed windows spanning the character bitmap; consist of a combination of average pixel density and measures of local alignment along some directions. NIST database and Devnagari digit databases are used for experimentation. In NIST database, training set consists of 60,000 patterns and testing set consist of 10,000 patterns. Devnagari digit database consist of 500 patterns. It is divided in two parts. Training set consists of 300 patterns and testing set consists of 200 patterns. These features used in conjunction with Extended Fuzzy Hyperline Segment Neural network (EFHLSNN). The performance of EFHLSNN is found to be superior compared to FHLSNN with respect to training time, recall time per pattern and recognition rate.**

***Index Terms*— Fuzzy Neural Network, Feature Extraction, Hyperline Segment, Handwritten Character Recognition**

I. INTRODUCTION

For more than thirty years, researchers have been working on handwritten character recognition. Over the last few years, the number of academic laboratories and companies involved in research on handwriting recognition has continually increased. Simultaneously, commercial products have become available. This new stage in the evolution of handwriting processing results from a combination of several elements: improvements in recognition rates, the use of complex systems integrating several kinds of information, the choice of relevant application domains, and new technologies such as high quality high speed scanners and inexpensive powerful CPUs.

Feature extraction is extracting from raw data the information which is most relevant for classification purpose, in the sense of minimizing within-class pattern variability while enhancing between class variability. There are various methods for extracting the features like Template matching, Deformable templates, Graph Description, Contours, Zernike moments, Fourier Descriptor etc[1]. But these methods are quite complex and not so much suitable for recognizing handwritten character. Template matching in its pure form is not well suited for character recognition. In Deformable templates each template is deformed in no. of small steps to match the candidate input pattern. For Chinese character recognition the graph descriptions are extracted from skeletons as features. Zernike moments have reconstruct ability which moment invariants lack. Fourier descriptors can't be applied to fragmented characters in a meaningful way since this method extract features from one single closed contour or skeleton. The simple approach for extracting handwritten character feature is simple feature extraction for handwritten character recognition using regularly placed windows.

Pattern classification deals with the problem of identifying the underlying structure of data. One important characteristic of human reasoning is the ease with of coping with uncertain or ambiguous data encountered in real life. The traditional (statistical) approaches to pattern classification have been found inadequate in such circumstances. The theory of fuzzy sets was suggested as a way of remedy this difficulty. Interpretation of the structure using fuzzy logic based techniques interest in neurofuzzy pattern recognition systems [4][5]. In this paper I have applied EFHLSNN which is an extension of Fuzzy Hyperline Segment Neural Network (FHLSNN) to the problem of Handwritten Character Recognition [6]. The FHLSNN utilizes fuzzy sets as pattern classes in which each fuzzy set is union of fuzzy set hyperline segments. The hyperline segment is a fuzzy set defined by two end points with membership function. The two end points of fuzzy hyperline segments are determined by FHLSNN algorithm.

Remaining part of the paper is organized as follows. Section II describes Preprocessing and Feature Extraction algorithm. Extended Fuzzy Hyperline Segment neural network algorithm is discussed in section III. Experimental results are tabulated in Section IV. Conclusions are discussed in section V. References are cited at the end.

II. PREPROCESSING AND FEATURE EXTRACTION

Before extracting feature, preprocessing is done on the image. First scaling is done by using zeroth order moment. After scaling translation is done by using first order By using Simple feature extraction technique for handwritten character recognition features are extracted [7]. The features are computed from binary image. First, the input image is partitioned into sub-images (windows) as shown in Figure 1.

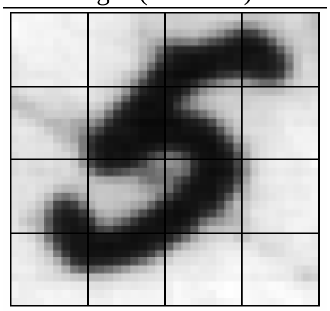

Figure 1. Image partition in Windows

From each window average pixel density and directional features are extracted. A fixed set of operators is applied to each window. First operator is simple bit counter that calculates the average pixel density. Other operators try to

Dipti Pawar is with the Department of Computer Engineering, Sinhgad College of Engineering, Pune, India (phone: 91-9922603212; e-mail: diptishitole@yahoo.co.in).

ISBN: 978-988-19251-1-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

estimate at which extent the meaningful pixels are aligned along some direction. 00, 450 and 900 directions are considered. A set of N equally spaced lines are defined, that span the whole window and parallel to chosen direction.

$$f(n) = \left\{ \begin{array}{cc} 0 & if\ n < 2 \\ 2^{n-2} & if\ n \geq 2 \end{array} \right\} \qquad (1.1)$$

Let's take the example of primitive feature along horizontal direction. After preprocessing image is converted into binary image.

Binary Image							Count (Meaningful pixels)	Quasi-Exponential function f(nj)
0	0	0	0	0	0	1	1	0
0	0	0	0	0	0	1	1	0
0	0	0	0	0	1	1	2	1
0	0	0	0	1	1	1	3	2
0	0	0	1	1	1	0	3	2
0	1	1	1	1	0	0	4	4
1	1	1	1	0	0	0	4	4
								13

Figure 2. Primitive Feature along Horizontal direction

From Binary image count the number of meaningful pixels from each horizontal line. After that apply the quasi-exponential function as shown in equation (1.1) on that count. Summing up all the contributions f(nj) from the N lines, Primitive feature along a given direction is obtained.

III. EXTENDED FUZZY HYPERLINE SEGMENT NEURAL NETWORK

The architecture of Extended Fuzzy Hyperline segment neural network as shown in Figure 3.The FR layer accepts an input pattern and consists of n processing elements, one for each dimension of the pattern. The FE layer consists of processing nodes that are constructed during training. There are two connections from each FR to FE node; one connection represents one end point for that dimension and the other connection represents another end point for that dimension.

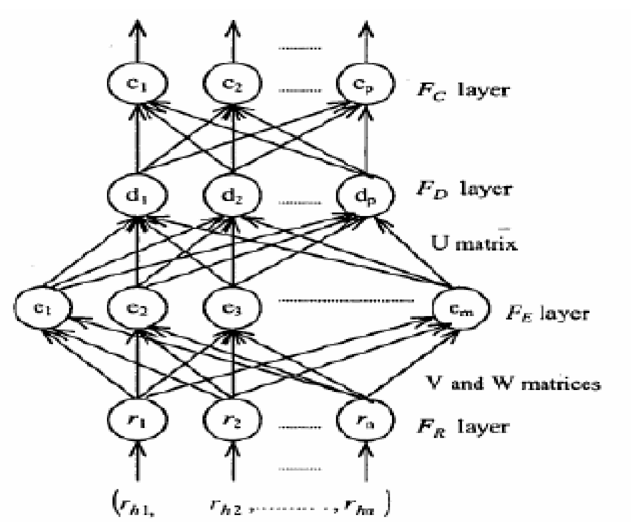

Figure 3.Extended Fuzzy hyperline segment neural network

Let Rh = (rh1, rh2, ..., rhn) represents the hth input pattern, Vj = (Vj1, Vj2, ..., Vjn) is one end point of the hyperline segment ej , and Wj = (Wj1,Wj2, ...,Wjn)is the other end point of ej . Then the membership function of jth FE node is defined as

$$e_j(R_h, V_j, W_j) = 1 - f(x, \gamma, l) \qquad (1.2)$$

In which x=l1+l2, In the FHLSNNN Euclidian distances l1, l2 and l are used those are defined as

$$l_1 = \left[\sum_{i=1}^{n} (w_{ji} - r_{hi})^2 \right]^{1/2} \qquad (1.3)$$

$$l_2 = \left[\sum_{i=1}^{n} (v_{ji} - r_{hi})^2 \right]^{1/2} \qquad (1.4)$$

$$l = \left[\sum_{i=1}^{n} (w_{ji} - v_{ji})^2 \right]^{1/2} \qquad (1.5)$$

Here, in this paper I have used Manhattan distance for computing the values of l1, l2 and l as

$$l_1 = \left[\sum_{i=1}^{n} \left| w_{ji} - r_{hi} \right| \right] \qquad (1.6)$$

$$l_2 = \left[\sum_{i=1}^{n} \left| v_{ji} - r_{hi} \right| \right] \qquad (1.7)$$

$$l = \left[\sum_{i=1}^{n} \left| w_{ji} - v_{hi} \right| \right] \qquad (1.8)$$

.f() is a three-parameter ramp threshold function defined as

$$f(x, \gamma, l) = 0 \text{ if } x = l \text{ otherwise}$$

$$f(x, \gamma, l) = \left\{ \begin{array}{cc} x\gamma & if\ 0 \leq x\gamma \leq 1 \\ 1 & if\ x\gamma > 1 \end{array} \right\}$$

The Manhattan distance has given best performance in terms of training and recall time in comparison with Euclidian distance (1.3),(1.4)and (1.5)

A. Significance of τ and θ

During learning phase maximum hyperline size is controlled by parameter θ (sometimes called as λ). For less value of θ more HLS are created, hence recall time per pattern is getting increased. For large value of θ less HLS are created, hence recall time per pattern is getting decreased. The parameter θ is set in such a way that less

ISBN: 978-988-19251-1-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

number of HLS are created and classification rate should be 100%.

Sensitivity parameter regulates how fast membership value decreases when the distance between Rh and ej increases. Where rh is input pattern and ej is membership of input pattern Rh for hyperline segment j [8].

As sensitivity parameter is getting increased membership value is getting decreased. For less value of τ I get more degree of membership. Sensitivity parameter affects the support of the fuzzy set.

IV. Experimental results

EFHLSNN is implemented using Matlab 7.0. Devnagari digit database consist of 500 characters. The data sets, Set-1 and Set-2 are training and testing sets. Set-1 consists of 300 characters and Set-2 consists of 200 characters. NIST database consists of 1500 characters. The data sets Set-1 and Set-2, are training and testing sets. Set-1 consists of 1000 characters and Set-2 consists of 500 characters.

Devnagari digit image is of size 65*65. Each image is divided into 5*5 i.e. 25 parts. From each part 4 features are extracted, so feature vector is of 100 dimensions. Set 1 is classification and Set 2 is recognition.

Experimentation is done on Set-1 of NIST database. From Set-1, 64 dimension features are extracted. More hyperlines are created as I decrease the value of expansion parameter i.e. θ and vice versa. As I decrease the value of sensitivity parameter i.e. more hyper lines are created and vice versa. I choose the value of θ in such a way that less hyperlines are created and for those hyper-lines classification should be 100%.

Table 1: Experimentation with EFHLSNN and FHLSNN

θ	τ	HLS created with EFHLSNN	HLS created with FHLSNN
0.004	1	840	872
0.003	1	861	910
0.003	0.8	901	951
0.002	1	888	941
0.002	0.8	923	974

As shown in Table 1 for θ=0.004 and τ = 1 hyperlines created are 840 and 872 with EFHLSNN and FHLSNN . As I decrease the value of θ by keeping the same value of τ, more hyperlines are created i.e. for θ=0.002 and τ =1 hyperlines created are 888 with EFHLSS. For θ = 0.003 and τ =1 hyperlines created are 861. For same value of θ and τ more hyperline segments are created with FHLSNN compare to EFHLSNN.

A. Experimental Results of Devnagari Digit database

Devnagari digit image is of size 65*65. Each image is divided into 5*5 i.e. 25 parts. From each part 4 features are extracted by using [1], so feature vector is of 100 dimensions. Set 1 is classification and Set 2 is recognition.

Table 2: Experimental Results with Devnagari Digit database

Classifier	HLS created	Set 1	Set 2
EFHLSS	250	100	70
FHLSS	274	100	61

B. Experimental Results of NIST database

Original size of NIST image is 28*28 . It is divided into 2*2, 4*4 and 8*8 window so I get 16, 64,196 feature vector size [9].

Table 3: Experimental Results of NIST database with EFHLSNN

Feature Vector	HLS created	Set 1	Set 2
16	843	100	53.8
64	650	100	72.6
196	663	100	76.6

Table 4: Experimental Results of NIST database with FHLSNN

Feature Vector	HLS created	Set 1	Set 2
16	884	100	45.6
64	703	100	67.6
196	726	100	70.8

As shown in Table 3 and table 4 as I increase the feature vector size recognition rate is also increased. The recognition rate of EFHLSNN is more than FHLSNN. The timing analysis of training and recall time per pattern are depicted in Table 5.

.

Table 5: Timing analysis with 196 features of NIST Database

Classifier	HLS created	Training time in seconds	Recall time per pattern in seconds
FHLSNN	726	95.235	1.986
EFHLSNN using Manhattan distance	663	93.112	1.777

As shown in Table 5 the training time and recall time per pattern of EFHLSNN is more than FHLSNN. EFHLSS creates less number of hyperline segments than FHLSNN.

ISBN: 978-988-19251-1-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

V. CONCLUSION

The EFHLSNN classifier using Manhattan distance has ability to train and recall patterns faster than FHLSNN using Euclidian distance. Simple feature extraction algorithm for Handwritten Character Recognition a novel feature extraction method that couples good performances with simplicity. Recognition rate is increases with increase in the dimension of feature vector. The parameter which is the maximum length of HLS controls the total number of hyperline segments created. For less value of this parameter more hyperline segments are created and vice versa. For less value of sensitivity parameter more hyperline segments are created and vice versa. Recall time per pattern depend on number of hyperline segments.

REFERENCES

[1] K. Jain, Q . D. Trier and Torfinn Taxt ,"Feature Extraction methods for character recognition-A Survey", July 1995.
[2] U. V. Kulkami, D.D.Doye and T. R. Sontakke, "General Fuzzy Hypersphere Neural Network", accepted for publication in IEEE CNM 2002 to be held atHonolulu, USA, May 2002
[3] Timothy J. Ross "Fuzzy logic and its applications" 1995,second edition,Wiley.
[4] L. Zadeh, "Fuzzy sets" Inform. and control, VOl.8, PP 338-353, 1965.
[5] Lippmann R.P.,"An Introduction to computing with neural nets", IEEE ASSP magazine,vol.4,No.2,1987,pp. 4-22.
[6] U. V. Kulkarni, T. R. Sontakke and G. D. Randale, "Fuzzy hyperline segment neural network for rotation invariant handwritten character recognition. ,IEEE, Trans, on neural network, Vol. 3, No. 2, pp. 241- 251, July 2001.
[7] S. J. Perantonis and P.J.G. Lisboa, "Translation, rotation and scale Invariant pattern recognition by high order neural networks and moment classifiers" IEEE Trans. On neural networks, Vol. 3, No.2, pp.241-251,1992.
[8] Patrick K. Simpson, "Fuzzy Min-Max Neural Networks-Part 1: Classification",IEEE Transactions on Neural Network, vol. 3, pp 776-786, September 1992
[9] The First Census Optical Character Recognition System Conference, Technical Report NISTIR 4912 (1992).

ISBN: 978-988-19251-1-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)