

Handwritten Character Recognition Systems Using Image-Fusion and Fuzzy Logic

Rupsa Chakraborty and Jaya Sil

Department of Computer Science and Technology,
Bengal Engineering and Science University, Shibpur,
Howrah -7111 03, India
rupsa_c@yahoo.com, js@becs.ac.in

Abstract. To date, image segmentation is used as a vital step to developing handwritten character recognition systems. In the paper, image segmentation has been avoided by applying image fusion and fuzzy method to recognize handwritten characters. At first, several digital images corresponding to each handwritten character are fused to generate patterns, which are stored in 10×10 matrices, irrespective of the size of the images. The character to be recognized is matched with each pattern and the best match pattern is considered as the most likely choice. In the second phase of the work subjective judgment using fuzzy logic has been incorporated to improve the recognition rate where Gaussian distribution function, with varied mean and standard deviation are considered as fuzzy membership functions. Several binary images have been tested to demonstrate the effectiveness of the proposed system.

1 Introduction

The problem caters to the automated recognition of handwritten characters (hereafter called characters) that require little or no manual intervention and save considerable amount of time, money and energy. To recognize a particular character, traditional image segmentation methods may be used like region-based methods [5]-[9] (e.g., texture analysis) and edge-based methods [2],[9] (e.g., dynamic programming). Some neural-network-based [6] segmentation techniques and intensity based segmentation procedures using Gabor filters [8] are also reported. Segmentation either implemented by arbitrarily subdividing the image into primitives [4] or by computing connected components. The features like the height, width of each segment and gap between two segments are obtained from the segmentation process. In dynamic programming technique, the matcher takes the input image, a lexicon (a list of strings representing all candidate characters), features of each image and returns a confidence value $[0,1]$ indicating the degree of matching to which the image represents that particular character [4]. Fuzzy integral aggregation is a procedure where a rule base of the characters are constructed and stored. The output of the rule base gives the confidence to the extent an image matches with a known character. A set of features is mapped as antecedents of the fuzzy rules while the consequents of the rules generate a numeric confidence that the sub-image belongs to a particular string.

No matter how efficient the technique is, segmentation consumes a lot of time. The methods which tried to avoid segmentation had to deal with lots of features [1] which add complexity in some way or other. The paper aims at developing a character recognition system with improved performance using image fusion and fuzzy logic. The system can be broadly classified into two phases. In the first phase images of a particular character are fused to generate a pattern and stored in a fixed size probability matrix, irrespective of the size of the image. The image to be recognized is then matched with every pattern generating a recognition score for each of the cases. The character that corresponds to the highest score is then accepted as the recognized character. In the second phase, for improving the recognition rate, subjective judgement has been incorporated using fuzzy method.

The paper is divided into four sections. In section 2 the proposed method is described in detail. Results are documented in section 3 while conclusions are summarized in section 4.

2 Proposed Work

Character set Ch: A set containing images of characters to be recognized.

Pattern matrix ($M_p[10][10]$): A matrix of elements within $[0,1]$ representing the probability of occurrence of a black pixel in the respective cell.

Test image: The image fed to the system for recognition.

Training image: The image representing a known character of *Ch* to be fused for generating M_p .

Pattern-matrix table: A file containing information for each character of *Ch*, such as the character itself, number of images till fused (*num*), and M_p .

Recognition score: Value denoting the extent to which the *test image* represents a member of *Ch*.

2.1 Coarse State of Recognition

Generation of Probability Matrix

In the first phase of recognition, following procedures are executed for the training images to generate M_p of each character.

a. Pre-processing: As a preprocessing step noise has been removed from the original image (fig.1(a)) by thresholding operation. Fig.1(b) shows one such image.

b. Thinning: The procedure converts the object in the image (which may be a few pixels thick) to a thickness of one pixel to eliminate redundancy.

c. Image fitting: The image is fitted within a frame that touches the character at its left, right, top and bottom. The time and space complexity is reduced effectively by eliminating redundant/useless information, see fig.1(c).

d. Image reduction: The image is divided into a 10×10 grid structure, shown in fig.1(d). A 10×10 matrix M_{tr} , of training images has been formed which

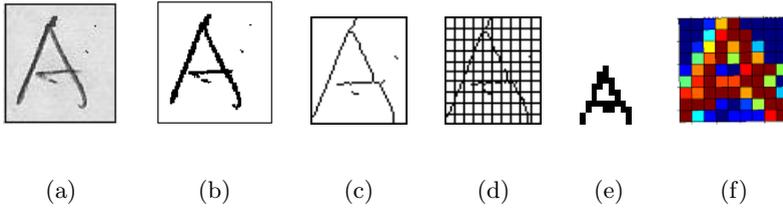


Fig. 1. (a)Original image. (b) Noise-reduced image. (c) After image fitting. (d) A 10×10 grid structure. (e) Reconstructed image (f) A pictorial representation of 'A'.

stores '1' in its (i,j) position provided more than 5 percent of the pixels in the cell(i,j) contains black pixels, else it stores '0'. Effective size of the character is thus reduced significantly and at the same time retaining all the features of the character. Interestingly, it has been observed that the same character is formed from $M_{tr}[10][10]$, only its size is greatly reduced. A reconstructed image obtained from fig.1(e) is shown in fig.1(f).

e. Image fusion: A look-up table (*pattern-matrix table*) stores a pattern matrix M_p corresponding to each element of Ch . Equation (1) explains the formation procedure of M_p , executed for fusing training images. It is worth mentioning that a training image may be fused with the existing pattern at any point of time.

$$M_p^{new} = \frac{1}{num + 1} * (num * M_p^{old} + M_{tr}) \tag{1}$$

Here, M_p^{new} is the pattern matrix obtained after fusing training images contained in M_{tr} and M_p^{old} (already stored in the *pattern-matrix table*). M_p^{new} is copied back to the table along with num increased by one. Finally, every cell of M_p represents the probability of occurrence of a black pixel that maps the *test image* to a typical character. Fig.1(f) gives a pictorial representation of M_p for character 'A'. The content ([0-1])of each cell of M_p is represented by colors ranging from blue to red in the order of VIBGYOR. For example, deep blue and deep red represent probability '0' and '1', respectively. Thus, the overall patterns of each character are restored in the pattern file.

f. Matching: The *test image* is processed similarly up to the *image-reduction* step to generate M_{tst} . M_{tst} is matched with M_p of each character stored in the *pattern-matrix table* and a recognition score is obtained, as suggested by equation (2).

$$RecognitionScore^k = \frac{\sum_{i=0}^{10} \sum_{j=0}^{10} (M_p^k[i][j] * M_{tst}[i][j])}{\sum_{i=0}^{10} \sum_{j=0}^{10} M_{tst}[i][j]} \tag{2}$$

$RecognitionScore^k$ denotes the extent of matching of the k^{th} pattern matrix with that of M_{tst} . The most likely recognized character is the one corresponding to the element with the highest *recognition score*.

2.2 Fine Tuning Using Fuzzy Logic

To improve the recognition rate, fuzzy approach is invoked. An image containing a standard representation of each element of Ch has been reduced (section 2.1) and stored in matrix namely, $st_img[10][10]$. The Gaussian distribution function is mapped as fuzzy membership function(g) and assigned to each column of st_img , shown in fig.2(a).

$$g(r) = e^{-\frac{(r-m)^2}{\delta}} \quad (3)$$

where, r is the row number of a particular column, m and δ denotes the mean and standard deviation, respectively. The function denotes the approximate chance of occurrence of a '1' in that row of the respective column. Membership functions of other columns of the standard image are obtained by changing m and δ only, shown in fig.2(b). A *region* is represented by a set of contiguous rows, for each column, containing '1'(fig 3(a)). A variable, no_reg , denotes the number of such regions in each column having a separate membership function. For each character, a *fuzzy-confidence table* contains no_reg for each column and pair(m,δ) for each region. For recognition, every column of M_{tst} is scanned from row 0 to 9 which generates a confidence value for every element of the matrix st_img and stored in $M_F[10][10]$. While scanning, the number of regions for the column is computed and stored in a variable $test_no_reg$. Here three situations may arise and are tackled in the following manner:

- (i) if $test_no_reg > no_reg$, then the fuzzy membership value (fig.3(b)) for every cell in that row is set to zero.
- (ii) if $test_no_reg = no_reg$ then for each region the fuzzy membership value (fig.3(b)) is computed from the membership function corresponding to that region.
- (iii) if $test_no_reg < no_reg$ then divide the column into number of parts equal to no_reg .

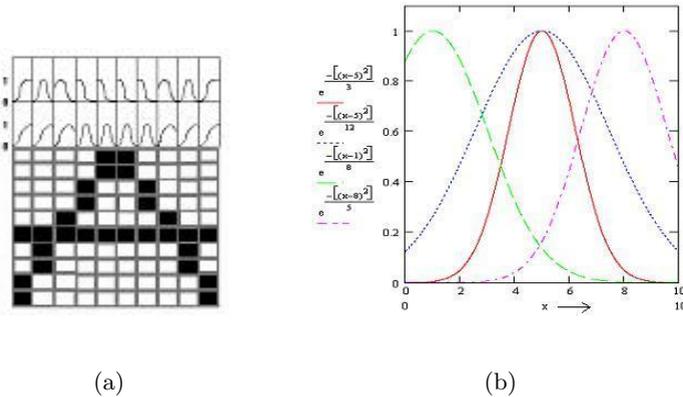


Fig. 2. (a) Membership function of each column. (b) Gaussian distribution functions with varied m and δ .

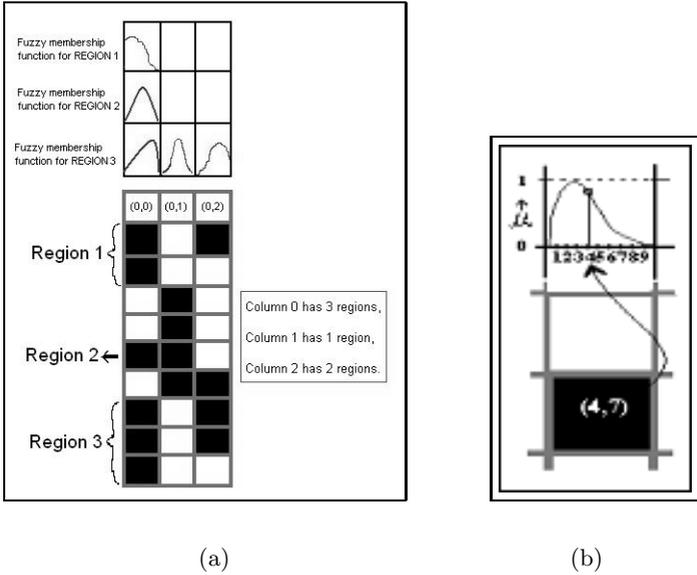


Fig. 3. (a) Illustrating computation of no_reg . (b) Illustrating the computation of g .

After computation of membership value, matching and computation of final recognition score is obtained by equation (4),

$$RecognitionScore^k = \frac{\sum_{i,j=0}^{10} \max(M_{tst}[i][j] * M_F[i][j], M_{tst}[i][j] * M_P[i][j])}{\sum_{i=0}^{10} \sum_{j=0}^{10} M_{tst}[i][j]} \quad (4)$$

The most likely recognized character is the one corresponding to the element with the highest *recognition score*.

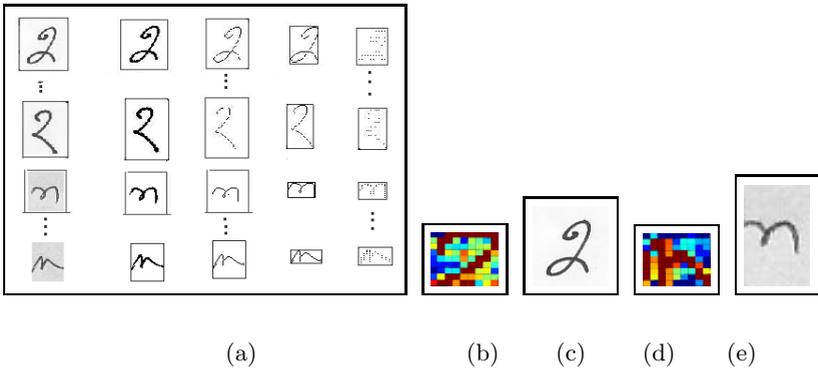


Fig. 4. (a) Image reduction of training images for '2' and 'n'. (b)&(d) pictorial representation of M_p for '2' and 'n' respectively. (c)&(e) Test image of '2' and 'n', respectively.

3 Results

The proposed algorithm has been applied on the elements of $Ch = \{0,2,E,T,S,Y, a,d,n\}$ (two instances shown in fig.4). Fuzzy method has been applied on the same instances which generates the recognition scores 0.6979 & 0.5999 for the test image of fig.4(c) and 4(e), respectively. The result reveals that the recognition rate is around 60 percent for the correct answer to be the one with best match and around 75 percent when the correct answer is either the best or the second best match.

4 Conclusion

The paper presents a generic method for recognition of handwritten characters within a finite character set. More insight in the existing systems shows that segmentation needed for such systems, perhaps, is not a very natural way of proceeding toward the ultimate goal. The performance of the system consists of various salient points: i) little time($O(|C|\log_2|C|)$) and memory requirements, ii) works efficiently irrespective of the size of the image, iii) can be extended to characters with varied shape, iv) avoids segmentation, v) multiple thinning is not required.

References

1. Ariki Y, Motegi Y, IEEE Trans. Pattern Anal., Machine Intell, 1995. Segmentation and recognition of handwritten characters using Subspace Method.
2. Bellmann Y (1957), Dynamic Programming, Princeton Univ. Press.
3. Chiang JH, Gader PD (Nov. 1997), Hybrid Fuzzy-Neural Systems in Handwritten Word Recognition, IEEE Trans. Comm.
4. Gader P, Keller JM, Cai J(Feb. 1995), A Fuzzy Logic System for the Detection and Recognition of Handwritten Street Numbers, IEEE Trans. Comm.
5. Halarick RM and Shapiro LG, Image segmentation techniques, Comp. Vision, Graphics, and Image Proc., vol.29, pp.100-132, 1985.
6. Jackel ID,et.al (1991), A neural network approach to handprint character recognition, IEEE Trans. PAMI.
7. Pal SK and Dutta Majumdar DK (1986), FUZZY Mathematical Approach to Pattern Recognition, Wiley Eastern Limited.
8. Richard Buse, Zhi-Qiang Liu and Jim Bezdek (February 2002), Word Recognition using Fuzzy Logic, IEEE Transactions on Fuzzy Systems, Vol 10, No.1.
9. Sonka M, Hlavac V, and R. Boyle (1998), Image Processing, Analysis, and Machine Vision, 2nd ed. Pacific Grove, CA:PWS.
10. Zamperoni P (1986), Analysis of some region growing operators for image segmentation, in Advances in Image Processing and Pattern Recognition, V. Cappellini and R. Marconi, Eds. Amsterdam: North Holland, pp. 204-208.