A Feature based on Encoding the Relative Position of a Point in the Character for Online Handwritten Character Recognition

Dinesh M, Murali Krishna Sridhar
HP Labs Bangalore, India
{dinesh.mandalapu, mkrishna}@hp.com

Abstract

Feature extraction is a very important step in the process of character recognition. The features extracted from the character should encode the local, global and the structural characteristics of the character shape. In this paper we propose a new feature for recognition of online handwritten characters called the star feature. The star feature encodes the local, global and structural characteristics of a character. The star feature describes every point of the character, in terms of its relative position with respect to the other points in the character. We have evaluated the performance of the star feature on IRONOFF data set and IWFHR-06 Tamil competition data set. The experimental results show that the star feature achieves high accuracy on both the data sets.

1. Introduction

An online handwritten character recognition system includes three steps: preprocessing, feature extraction and classification. Feature extraction is a very important step in online character recognition. The extracted features have to represent the shape in consideration efficiently. The performance of a classifier depends on the features input to it. Depending on the granularity at which the features are extracted from a character, they are classified as local and global features. Local features are extracted at each point of the character. These describe the character at point level and hence fail to capture the global structure of the character. Some examples of local features are $x - y$ coordinates, tangent angle features, etc. Most online character recognition systems use the direction feature and the direction change feature extracted at point level for classification [1].

Global features are extracted at character level or at sub-character level. These features try to describe the shape of the character as a whole and do not capture the local point wise variations of the character. Examples of global features are moments, fourier descriptors, projections, absolute position, accumulated direction change etc. Most online character recognition systems use the direction density information extracted from subparts of the character for classification [7]. As most of the features proposed in the literature are either global or local, they fail to capture the essential information required to represent the shape of the character. To capture local and global variations in the character, methods which use a combination of both local and global features have been proposed. In [8] a combination of peripheral shape features, stroke density features and stroke direction features are used. Though combination schemes try to capture the overall variations in the character, they require a combination parameter to be learnt from the training data. These combination schemes cannot be used in a script independent and writer independent scenario as the optimum combination parameter varies for different writers and different scripts. Global features such as concavity, presence or absence of loop etc., can't be represented with a real value. Parametrization of non-real valued global features is non trivial and hence it is not easy to combine non-real valued global features with real valued local features. Non-real valued global features also cannot be used in statistical classifiers. Methods for parametrization of structural features have been discussed in [4]. Hence there is a need for features that capture both the global and local shape properties of the character.

The shape context feature in [2], which is described as the coarse distribution of the rest of the shape with respect to a given point on the shape, tries to capture both the local and global variations present in the character without any combination. The shape context feature is extracted by populating the r and θ bins placed at each point of the character. Generally an online handwritten character is represented by $[60 - 100]$ points sampled along its trajectory. Due to the less number of points per character, most of the bins in the shape context feature are sparsely populated. Hence, the shape context feature becomes very sensitive to slight variations in the orientation of the character. In this paper we present a feature called the *star feature*, which captures both the local and global information present in the character. The *star feature* describes every point in the character, in terms of its relative position with respect to the other points in the character. We compare the performance of the *star feature* with directional element feature [7] and $x-y$ coordinate feature on both the IRONOFF [10] and IWFHR-06 Tamil competition datasets [5].

The rest of the paper is organized as follows. Section 2 presents the definition and implementation details of the proposed star feature. In Section 3 we present the experimental results and conclude the paper in Section 4.

2. Star Feature

Star feature encodes the relative position of a given point with respect to the other points in the character in the eight directions as shown in Figure 1. It encodes the point of intersection of the eight direction lines originating from a point in the character with the other parts of the character. In this section we explain the *star feature* and present an algorithm for the extraction of star feature.

2.1. Star Feature Definition

Given a point P_i of a character, the star operator shown in Figure 1 is applied at the point P_i to find the point of intersection of the eight direction lines with different parts of the character. The feature vector F_i for each point P_i is an 8 bit binary vector. Each bit in F_i corresponds to each of the eight direction lines of the star operator. A bit in the binary vector is set if there is a point of intersection along that direction. Figure 2 illustrates the eight directional star operator applied at point P_i of the character. For example, to find if the vertical line passing through P_i intersects the character,

1. Find the nearest points to the vertical line in each of the four quadrants formed by the intersection of vertical and horizontal lines at the point P_i.

2. Set the bit corresponding to the direction 3, if the nearest points to the vertical line located in first and second quadrants are adjacent in the temporal order.

3. Set the bit corresponding to the direction 7, if the nearest points to the vertical line located in third and fourth quadrants are adjacent in the temporal order.

The 8 bit binary vector obtained for point P_i shown in Figure 2 is $F_i = [1, 1, 1, 1, 1, 0, 0, 0]$. The last three bits in the vector are not set as there are no points of intersection in those directions. The feature vector F for each character is obtained by concatenating each of the 8 bit binary vectors F_i obtained at each point of the character P_i.

$$F = [F_1, F_2, \ldots, F_N]$$

where, N is the number of points in the character.

The *star feature* is extracted by finding the nearest neighbors of each point in each of the eight directions. Hence, it also captures the direction information in addition to the relative position information. The encoding of the star feature can be used to describe the structural part of the character, to which the corresponding point belongs. For example, the star feature vector F_i for a point P_i inside the loop of a character is [1,1,1,1,1,1,1,1]. Similarly, the feature vectors for a point in the concave or convex part of the character are [1,1,1,1,0,0,0,0] and [0,0,0,0,1,1,1,1] respectively.

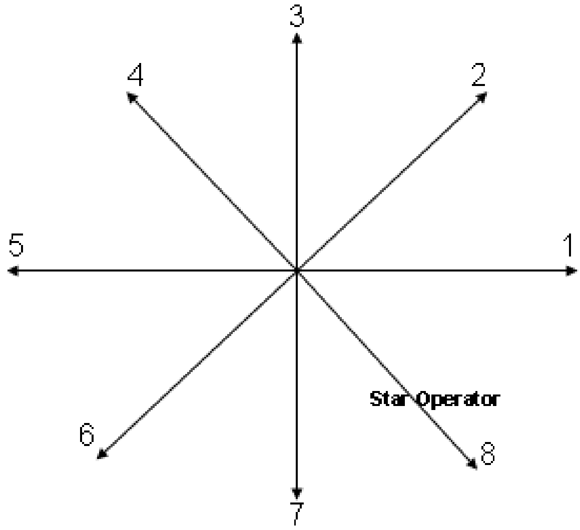

Figure 1. Eight Direction Star Operator

2.2. Star Feature Extraction Algorithm

1. Apply the star operator on each point P_i.

2. Find the point of intersection of each of the 8 direction lines of the star operator and the other parts of the character.

3. Set the bits of the 8 bit vector F_i, corresponding to the directions in which there is a point of intersection.

4. Concatenate each of the 8 bit vectors F_i obtained from each point P_i, to form the feature vector F.

3. Experiments

We evaluated the performance of the proposed feature on IRONOFF dataset (English upper and lowercase characters and numerals) and Tamil dataset (IWFHR-06 Tamil Competition Dataset). We also compared the performance of the proposed feature with the directional element feature described in [7] and the $x-y$ coordinate feature.

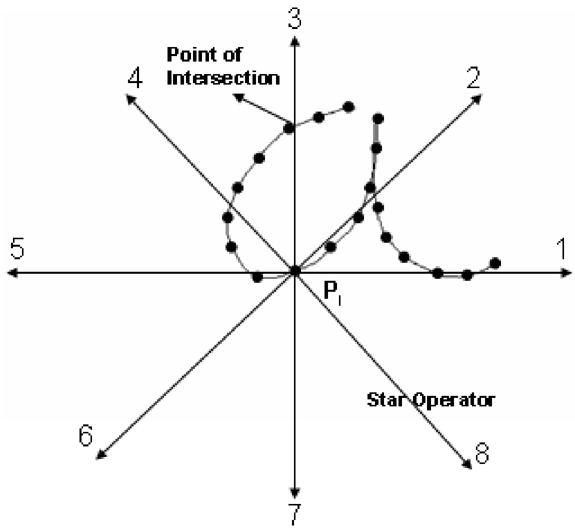

Figure 2. Star Operator Applied at a Point P_i of the Character

3.1. Feature Extraction

Before extraction of features each character is resampled to a fixed length of 60 points per character and then normalized to a standard scale of size 10×10. After normalization, a character sample is smoothened using a moving average filter to remove the random and high frequency noise. A detailed description of the preprocessing can be found in [6].

From the preprocessed sample, the following features are extracted:

1. Normalized $x - y$ coordinates - The dimension of this feature vector is 120 and the components of the feature vector are real valued.

2. Direction element feature(DEF) - The DEF feature is extracted as described in [7]. The online character sample is normalized using the nonlinear normalization method described in [9]. The normalized character is then converted to a binary image of size 64×64. The direction elements are extracted by applying the dot orientation patterns on the contour image. The DEF feature is a vector of dimension 196 and is real valued.

3. Star feature - The star feature is extracted from the preprocessed character sample as described in section 2. The dimension of the star feature is 480 and is binary valued.

Table 1. Results of NN classifier using the three different features

Feature	Classification Accuracy on Different Datasets			
	IRONOFF Data			Tamil Data
	lower	upper	numerals	
x-y	80.1	83.8	90.2	77.6
DEF	81.2	84.3	90.4	76.3
Star	84.9	88.6	93.5	80.2

3.2. Classification and Evaluation

We used Nearest Neighbor classifier to calculate the accuracy of recognition. Euclidean distance metric was used as the measure of dissimilarity between two feature vectors. The evaluation on the IRONOFF dataset is performed by randomly dividing the data set into three folds and then calculating the three fold cross validation accuracy. In the case of Tamil dataset the competition train data has been used for training and the competition test data has been used for testing. The results from the experiments are tabulated in Table 1. It can be seen that the star feature performs better than the other two features for both IRONOFF and Tamil datasets. Moreover, as the *star feature* is binary valued the time complexity of computation of euclidean distance and the memory requirements are low as compared to the DEF and the $x - y$ features.

4. Conclusions

We have presented a new feature for character recognition. This is based on encoding the relative position of every point in the character with respect to the other parts of the character. This feature captures both the local and the global information required to describe the character and hence performs better as compared to the $x - y$ feature and the DEF feature. As the proposed feature vector is binary, it takes less time to calculate the dissimilarity measure and also the memory required to store the feature is quite low.

This is only a preliminary exploration with the star feature. In our future work we will use the star feature for extracting structural information like concavities, loops, cusps etc. from the character. The current evaluation was done using Nearest Neighbor classifier, in future we would evaluate the performance of the *star feature* with other classifiers like SVM, HMMs etc. Currently, the *star feature* is encoded as a binary vector. In our future work we would investigate the advantages of using the $x - y$ coordinates of the points of intersection to form the *star feature* vector. We would also investigate on the performance of the star feature when it is used in combination with the other features.

References

[1] C. Bahlmann. Directional Features in Online Handwriting Recognition. *Pattern Recognition*, 39(1):115–125, January 2006.

[2] S. Belongie and J. Malik. Shape Matching and Object Recognition Using Shape Contexts. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(4):509–522, April 2002.

[3] J. Cai and Z.-Q. Liu. Integration of Structural and Statistical Information for Unconstrained Handwritten Numeral Recognition. *IEEE Trans. Pattern Analysis Machine Intelligence*, 21(3):263–270, March 1999.

[4] L. Heutte and T. Paquet. A Structural/Statistical Feature Based Vector for Handwritten Character Recognition. *Pattern Recognition Letters*, 19(7):629–641, May 1998.

[5] HP Labs Isolated Handwritten Tamil Character Dataset. http://www.hpl.hp.com/india/research/penhw-interfaces-1linguistics.html.

[6] N. Joshi and G. Sita. Comparison of Elastic Matching Algorithms for Online Tamil Handwritten Character Recognition. *Proceedings of the Ninth International Workshop on Frontiers in Handwriting Recognition*, 2004.

[7] N. Kato and S. Omachi. A Handwritten Character Recognition System Using Directional Element Feature and Asymmetric Mahalanobis Distance. *IEEE Trans. Pattern Analysis Machine Intelligence*, 21(3):258–262, March 1999.

[8] Y. Y. Tang and L. T. Tu. Offline Recognition of Chinese Handwriting by Multifeature and Multilevel Classification. *Pattern Analysis Machine Intelligence*, 20(5):556–561, May 1998.

[9] W. Tianlei and M. Shaoping. Feature Extraction by Hierarchical Overlapped Elastic Meshing for Handwritten Chinese Character Recognition. *Proceedings of the Seventh International Conference on Document Analysis and Recognition*, 1:529–533, January 2003.

[10] C. Viard-Gaudin and P. M. Lallican. The IRESTE On/Off (IRONOFF) Dual Handwriting Database. *Proceedings of the Fifth International Conference on Document Analysis and Recognition*, 1999.