INCORPORATING A NEW RELATIONAL FEATURE IN ARABIC ONLINE HANDWRITTEN CHARACTER RECOGNITION

Sara Izadi and Ching Y. Suen
Centre for Pattern Recognition and Machine Intelligence, 1455 de Maisonneuve Blvd. West, Montreal, Canada

Keywords: Feature extraction, online handwriting, neural network, recognition.

Abstract: Artificial neural networks have shown good performance in classification tasks. However, models used for learning in pattern classification are challenged when the differences between the patterns of the training set are small. Therefore, the choice of effective features is mandatory for obtaining good performance. Statistical and geometrical features alone are not suitable for recognition of hand printed characters due to variations in writing styles that may result in deformations of character shapes. We address this problem by using a relational context feature combined with a local descriptor for training a neural network-based recognition system in a user-independent online character recognition application. Our feature extraction approach provides a rich representation of the global shape characteristics, in a considerably compact form. This new relational feature provides a higher distinctiveness and increases robustness with respect to character deformations. While enhancing the recognition accuracy, the feature extraction is computationally simple. We show that the ability to discriminate in Arabic handwriting characters is increased by adopting this mechanism in feed forward neural network architecture. Our experiments on Arabic character recognition show comparable results with the state-of-the-art methods for online recognition of these characters.

1 INTRODUCTION

For a variety of devices such as hand-held computers, smart phones and personal digital assistants (PDAs), the integrated online handwriting recognition interface is a useful complement to keyboard-based data entry. Variation (the unique way of writing of each individual) and Variability (the changes of a specific individual in producing different samples in different times) are the main challenges for automatic handwriting recognition regardless of the type of application or device. Both improving the recognition performance and extending the number of supported languages for those devices have attracted increasing research interests in the field of online handwriting recognition.

The writing environment in online recognition systems is the surface of a digitizer instead of a piece of paper. The trace of the pen tip is recorded as a sequence of points, sampled in equally-spaced time intervals (x(t),y(t)). Any sequence of points from a pen-down state to the next pen-up state is called a stroke. Dynamic and temporal information in online systems can cause more ambiguity by introducing more inter-class variability. Variations in the number of strokes and the order of strokes make the recognition task more complex than in off-line case. Recognition of scripts with complex characters shapes and a large number of symbols makes it even more difficult. Arabic online handwriting recognition is less explored, due to cursive characteristics of the symbols. Besides having huge variations in different people's writings, there are a lot of similarities between different letters in the alphabet. The Arabic alphabet consists of 28 letters, in 17 main shapes. These shapes are shared between some letters. Figure 1 shows 17 representative shapes from each class of very similar characters when dots are ignored. Dot(s) can appear above, below or inside a letter in Arabic. Various classification methods have been previously studied for online systems, such as neural networks (Verma et al., 2004), support vector machines (Bahlmann et al., 2002), and structural matchings on trees and chains (Oh and Geiger, 2000). However, selecting useful features is more crucial to both learning and recognition than the choice of the classification method. In this paper, we adopt a global feature inspired by shape context representation (Belongie et al., 2002) for similarity measures between images. This method may introduce high dimensional feature vectors in general. However, we investigate ways to adopt this feature for online handwriting applications

Izadi S. and Y. Suen C. (2008).
INCORPORATING A NEW RELATIONAL FEATURE IN ARABIC ONLINE HANDWRITTEN CHARACTER RECOGNITION.
In *Proceedings of the Third International Conference on Computer Vision Theory and Applications*, pages 559-562
DOI: 10.5220/0001087005590562

1	2	3	4	5	6
ا	ٮ	ح	د	ر	س
7	8	9	10	11	12
ص	ط	ع	ڡ ٯ	ک	ل
13	14	15	16	17	
م	ں	ه	و	ى	

Figure 1: The 17 groups of Arabic isolated characters and their assigned class numbers.

efficiently. The proposed technique is computationally simple and provides compact, yet representative, feature vectors. Our empirical evaluation for recognition of isolated online Arabic characters illustrates comparable results with corresponding state-of-the-art.

2 RELATIONAL APPROACH FOR FEATURE REPRESENTATION

Features can be broadly divided into two categories: local, and global. To compute a global feature, the whole sequence of trajectory points is used. Local features are computed by only considering the trajectory points in a certain vicinity of a point. Global features may provide more descriptive information about a character than local features. However, these features usually come with a high computational cost. The choice of effective features is mandatory for reaching a good performance. Our model does not use local features, however, it does use a temporal ordering of observed points. It allows for spatio-temporal data representation and augments the visual realism of the shape of a character. Such contextual modeling is especially used in *shape context* representation methods in machine vision literature.

2.1 Preprocessing

In order to reduce the noise introduced by the digitizing device, we need to preprocess the data. The preprocessing operations applied for our recognition system includes three steps: first *smoothing*, then *de-hooking*, and finally *point re-sampling*. Slight shakes of the hand is the source of the noise that the smoothing operation tries to remove. If each point on the main stroke is expressed as (x_i, y_i) in the Cartesian system, then the trajectory is smoothed by a weighted averaging ($1D$ discrete Gaussian filtering) of each point and its two immediate right and left neighbors:

$$(x_i, y_i) = 1/4(x_{i-1}, y_{i-1}) + 1/2(x_i, y_i) + 1/4(x_{i+1}, y_{i+1}) \quad (1)$$

Figure 2 depicts the Arabic letter "Dal" in its original recorded form and after applying preprocessing operations. The effect of the smoothing operation is illustrated in Figure 2(b). In online handwriting data, a hook-shaped noise often occurs at the start or the end of strokes. Figure 2(a) shows a starting hook. Hooks appear due to digitalizing device inaccuracy in the detection of a pen-down event, or due to fast hand motions when positioning the pen on, or lifting it off of the writing surface. Elimination of this noise is called de-hooking. In this research, for de-hooking, the head or tail of a stroke is considered a hook if its length is short compared to the length of the stroke, and the direction of the writing undergoes a sharp angular change. The (x, y) coordinates obtained from the graphic tablet are originally equidistant in time. However, the distribution of points along the trajectory can be uneven due to variations in the writing speed (see Figure 2(a)). The re-sampling of the data produces points which are equi-distant in space instead of time. This operation is used for either down-sampling or up-sampling, when fewer computations or more data points are desired respectively. Figures 2(d) and 2(e) show examples of re-sampling. We also apply size normalization in the preprocessing stage in order for all characters to have the same heights, while their original aspect ratios remain unchanged.

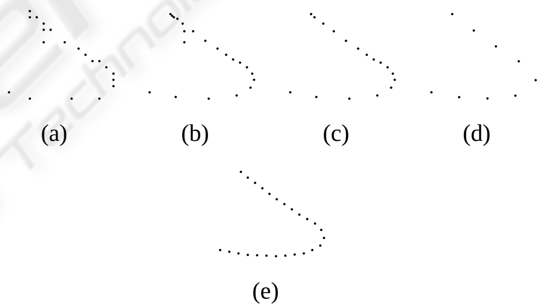

Figure 2: The Arabic letter "Dal" in its (a) original form, and its preprocessed versions when (b)smoothing (c)de-hooking, (d)down-sampling and (e)up-sampling have been applied.

2.2 Feature Extraction

Shape context is a shape descriptor proposed for measuring similarities in the context of shape matching (Belongie et al., 2002). Given that shapes are represented by a set of points sampled from their contours, the shape context represents a coarse distribution of the rest of the shape with respect to a given point in the shape. For a shape with N boundary points, a set of $N-1$ vectors originating from each boundary point to all other sample points expresses the configuration of the entire shape relative to that point. A coarse histogram of the relative coordinates of those remaining $N-1$ points is defined to be the shape context of that particular point. The representation and its matching method is computationally expensive. In applications like online handwriting recognition, all computations need to happen in real time. Therefor, shape context in its original form cannot be applied directly. Our proposed feature is some adaptation of this idea. First, we introduce some notations. We denote the points character trajectory by P and the set of reference points by R. Our feature selection method is explained in Algorithm 1 which selects adequate features of relational type. We call this feature *RF*. In the first part of this algorithm, we select an arbitrary and fixed set R. This set of points must not be outside the bounding box. The points in R may capture some interrelationship structures (for instance, symmetrical corners of the bounding box), or R may contain only one point). The algorithm samples a set of points S which is equi-distanced from the normalized representation of the character, P. The surrounding area of each reference point that falls in the bounding box is divided into bins according to the distance and the angle with respect to the reference point. The values of all the bins are initialized to zero for each reference point. Then, all the points in S are described from the view of each reference point and are placed into the corresponding bins. This is done by computing the distance and angle between the pair of points, $dist(r,p)$ and $angle(r,p)$, and updating the corresponding bin that this pair can be mapped on. After this step, the number of points in the bins provided by all the reference points will give a compact representation of the character.

Algorithm 1 Relational Feature Extraction.

INPUT: A set of re-sampled trajectory points S
OUTPUT: Feature vector V
Select an arbitrary set of reference points R
Select an arbitrary set of r-bins: $r_1, r_2, \ldots, r_n$ and θ-bins: $\theta_1, \theta_2, \ldots, \theta_k$
for all $r \in R$ **do**
  $Initialize(r-bins, \theta-bins)$
  for all $s \in S$ **do**
    Compute $dist(r,s)$
    Compute $angle(r,s)$
    $Assign(r-bins, \theta-bins, r, s)$
    $Update(r-bins, \theta-bins)$
  end for
end for
$V = count(r-bins, \theta-bins, r)$
Return V

In our experiments with a variety of point sets, we noticed that the geometrical center of each character bounding box provides better recognition results, while it keeps the size of the feature vector more manageable. Therefore, we used this point as reference point in a log-polar coordinate system.

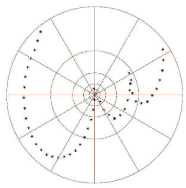

Figure 3: Diagram of the log-polar bins around the center of the bounding box used for relational context feature computation.

Using a log-polar coordinate system makes the descriptor more sensitive to differences in nearby points. Figure 3 shows the log-polar histogram bins for computing the feature of the Arabic character for letter S, pronounced as "Sin". The center of the circles is located at the center of the bounding box of the normalized letter. The template for extracting the context feature for the shown diagram has 5 bins in the tangential direction and 12 bins in the radial direction, yielding a feature vector of length 60. We capture the global characteristics of the character by this feature. We also use a directional feature to extract the local writing directions. The tangent along the character trajectory is calculated as following:

$$Arctan((x_i - x_{i-1}) + j(y_i - y_{i-1})) \quad for \quad i = 2:N, \quad (2)$$

3 RECOGNITION SYSTEM

In this paper, we have used artificial neural networks (ANNs) as our learning classifier method. Our relational feature magnifies the differences between similar characters and improves learning in ANNs. We train a Multi-Layer Perceptrons (MLPs) through a conjugate gradient method for classification using three-layer network with 100 nodes in the hidden layer.

4 EMPIRICAL RESULTS

We used a data set of isolated Arabic letters, courtesy of INRS-EMT Vision Group. In this database, the Arabic letters are divided into 17 classes according to their main shapes (Figure 1). This database was produced by the contribution of 18 writers with a large variety in samples (see (Mezghani et al., 2005) for details). A combination of two separately trained Kohonen Neural Networks in the voting scheme was previously used in the literature for recognizing online isolated Arabic characters (Mezghani et al., 2003). We evaluated the representational power of RF by training the network performing 6-fold cross-validation for 10 times. Previous studies (Mezghani et al., 2003; Mezghani et al., 2005) tested on this database, used 288 and 144 samples of each class for training and testing, respectively. We have used the same 2 out of 3 ratio between the number of training and testing samples to make our results more comparable to those previously reported. We aimed for more confidence in the reported recognition rate by following the experimental set up as explained over ten runs reported the statistics on recognition results with 95% confidence interval. This total recognition is 95.2 ± 0.12.

Table 1: Performance comparison of different recognition systems.

Performance	1-NearestNeighbor	Kohonen memory	RF-Tangent
Recognition Rate	Ref	-1.19	+ 4.22
Training Time	-	2 hrs	7.5 min
Recognition Time	526 s	26 s	23 s

Table 2: Confusion matrix for a sample run using the combination of relational and directional features.

class label	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	Recog rate%
1	143	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	99.31
2	0	137	0	0	0	1	1	0	0	1	0	0	0	4	0	0	0	95.14
3	0	0	144	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100.00
4	0	0	0	130	0	0	0	0	0	1	0	12	0	1	0	0	0	90.28
5	0	2	0	1	140	0	0	0	0	0	0	1	0	0	0	0	0	97.22
6	0	1	0	1	0	137	0	0	0	2	0	0	1	0	0	0	2	95.14
7	0	1	0	0	0	1	142	0	0	0	0	0	0	0	0	0	0	98.61
8	0	0	0	0	0	0	0	144	0	0	0	0	0	0	0	0	0	100.00
9	0	0	0	0	0	0	0	0	141	0	0	0	1	0	1	0	1	97.92
10	0	6	0	0	1	2	1	0	0	133	0	0	0	0	0	0	1	92.36
11	0	0	0	0	0	0	0	0	0	0	144	0	0	0	0	0	0	100.00
12	0	0	0	7	0	0	0	0	0	1	0	133	0	3	0	0	0	92.36
13	0	0	0	0	0	1	0	0	1	0	0	0	139	0	0	3	0	96.54
14	0	0	0	3	0	0	0	0	0	0	0	2	0	132	3	0	4	91.67
15	0	0	0	0	0	1	1	0	0	0	1	0	1	1	139	0	0	96.53
16	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	141	0	97.92
17	0	1	0	0	0	0	0	0	0	1	0	0	0	2	5	0	135	93.75
total rate																		96.16

Table 1 summarizes the results. The first column presents our experiments with the 1-Nearest Neighbor classifier as this classifier is often used as a benchmark. Its asymptotic error rate is less than twice the optimal Bayes rate. However, its asymptotic throughput is zero. Second column shows the results reported in (Mezghani et al., 2005), and the last column shows the results of our experiments with feed forward MLP neural network classifier equipped with the RF feature and the tangent feature, described in Section 3, conducted on Pentium(R)D with 340-MHz CPU. In the first row, the recognition rate of all methods presented. The recognition rate for 1-Nearest Neighbor classifier is considered as the reference point for this comparison. Our method shows 4.22% improvement, on average, compare to the one achieved by 1-Nearest Neighbor classifier. This is while the results in (Mezghani et al., 2005) in the best case can only reach 1.19% less than 1-Nearest Neighbor classifier. The training and recognition times presented in the second and third rows of Table 1. Although the results reported in (Mezghani et al., 2005) are based on slightly faster machine than ours, recognition is faster by our method and the training time is significantly shorter than the ones in (Mezghani et al., 2005). The confusion matrix of a sample run is presented in Table 2. The recognition rate for different classes vary. On average, our recognition rate for different letters was more robust than the best ones reported in (Mezghani et al., 2005). This confirms the high discrimination ability of the feature vectors introduced in our recognition system.

5 CONCLUSIONS

We introduced relational feature for online handwriting recognition and showed the usefulness of this feature in Arabic character recognition. Our experiments suggest that this feature representation improves the state-of-the-art recognition performance. This representation provides a rich enough representational feature for the global shape of these characters. A combination of this global feature, and the local tangent feature which captures the temporal information of the online data, improves the recognition rate compared for the same database. In future, we intend to investigate the use of these features in other supervised learning techniques and also for online word recognition.

REFERENCES

Bahlmann, C., Haasdonk, B., and Burkhardt, H. (2002). On-line handwriting recognition with support vector machines – a kernel approach.

Belongie, S., Malik, J., and Puzicha, J. (2002). Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(4):509–522.

Mezghani, N., Cheriet, M., and Mitiche, A. (2003). Combination of pruned kohonen maps for on-line aracic characters recognition.

Mezghani, N., Mitiche, A., and Cheriet, M. (2005). A new representation of shape and its use for high performance in online arabic character recognition by an associative memory. *IJDAR*, 7(4):201–210.

Oh, J. and Geiger, D. (2000). An on-line handwriting recognition system using fisher segmental matching and hypotheses propagation network. *IEEE Conference on Computer Vision and Pattern Recognition*, 2:343 – 348.

Verma, B., Lu, J., Ghosh, M., and Ghosh, R. (2004). A feature extraction technique for online handwriting recognition. *IEEE International Joint Conference on Neural Networks*, 2:1337 – 1341.