

A NEW REPRESENTATION OF CHARACTER SHAPE AND ITS USE IN ON-LINE CHARACTER RECOGNITION BY A SELF ORGANIZING MAP

N. Mezghani, A. Mitiche

INRS-EMT
800, de la Gauchetière Ouest, Suite 6900
Montréal QC, H5A 1K6 Canada.

M. Cheriet

École de Technologie Supérieure
1100 rue Notre-Dame Ouest
Montréal QC, H3C 1K3 Canada

ABSTRACT

The purpose of this study is to investigate a new representation of shape and its use in handwritten on-line character recognition by a Kohonen associative memory. This representation is based on the empirical distribution of features such as tangents, and tangent differences at regularly spaced points along the character signal. Recognition is carried out by a Kohonen neural network trained using the representation. In addition to the traditional Euclidean distance, functions such as the Kullback Leibler divergence and the Hellinger distance are investigated to evaluate similarity of feature vectors because these functions provide measures of distance between distributions. We report on extensive experiments using a database of on-line Arabic characters produced without constraints by a large number of writers. Comparative results show the representation relevance and the superior performance of the scheme.

1. INTRODUCTION

Character recognition is a fundamental, challenging problem in pattern recognition, and one with numerous useful applications ranging from electronic archival of scanned text to human-machine interfaces.

Of the two major issues in character recognition, character shape representation and category assignment, the latter has been the focus of most studies. The underlying hypothesis was that representation by curvature, tangents, moments of various kind, and others, of long standing use in pattern classification at large, were good enough, and that classification was the crucial, complex task to focus on. However, notwithstanding the complexity of the category assignment problem, a good representation is manifestly as important as a good classifier.

The document analysis literature abounds of character recognition methods and there are several review papers on the subject [1, 2, 3]. By and large, most of these methods pertain to off-line data from scanned paper documents or other such digital pictures. Significantly fewer methods deal with on-line data because affordable practical applications of on-line character recognition are much more recent. However, the basic concepts of pattern representation and classification apply to both input modalities, the major difference being the constraint of real-time response imposed by most applications of on-line character recognition.

The latest on-line character recognition methods implement algorithms of the major pattern classification paradigms [4, 5, 6, 7, 8, 9]. Prominent among these are neural networks methods because of their short time of development, their classification potency, and real-time response to classification requests. In previ-

ous studies, we investigated neural networks for both off-line and on-line character recognition. A multilayer perceptron [10], and Kohonen associative memories were trained to recognize printed characters from a large number of fonts [11], and off-line handwritten characters [12]. Finally, we investigated Kohonen memories and combination of Kohonen memories as on-line character classifiers [13, 14].

Pattern representation has not been the main explicit issue in most on-line character recognition studies, as these concentrated more on classification algorithms. Decomposition of character patterns into characteristic strokes has been most often used [15, 16, 17]. The $x - y$ coordinate string of the input signal [8], global shape descriptors such as Fourier coefficients [13, 18], and local geometric descriptors such as tangents, have all been in use. There have also been efforts to model pen-tip movements [19] to extract features, such as curvilinear and angular velocities. This study is along the vein of our previous ones on neural networks as pattern classifiers. The purpose is to investigate a new representation of shape and its use in a Kohonen associative memory for superior performance in on-line handwritten character recognition.

The representation is based on the empirical distribution (histograms) of tangents, and tangent differences at regularly spaced points along the character signal. Empirical distributions of characteristics features, which can be seen as marginal distributions of the underlying distribution [20], provide powerful discriminative statistics. Classification is carried out by a Kohonen neural network trained using the representation. However, in addition to the traditional Euclidean distance, functions such as the Kullback Leibler divergence and the Hellinger distance are investigated to evaluate similarity of feature vectors because these functions are more appropriate measures of distance between distributions. We report on extensive experiments using a large database of on-line Arabic characters produced without constraints by a large number of writers. Comparative results show the pertinence of the representation and the superior performance of the scheme.

The remainder of this paper is organized as follows: Section 2 describes the representation, Section 3 the Kohonen associative memory and the feature vectors distances used. Sections 4 and 5 give a detailed account of the experiments and results. Finally, Section 6 contains a conclusion.

2. FEATURE REPRESENTATION

We investigate a representation based on feature empirical distributions (histograms). The features considered are tangents, and tangent differences at regularly spaced points along the character

signal.

Let $\Gamma(s)$ be the arc length parametric representation of the curve of a character, and N equidistant points on $\Gamma(s)$:

$$((x_0, y_0)(x_1, y_1) \dots (x_k, y_k) \dots (x_{N-1}, y_{N-1}))$$

Let $\phi^{(0)}$ be the feature the measurements of which are the tangent angles defined by:

$$\theta_k = \arctan\left(\frac{y_{k+1} - y_k}{x_{k+1} - x_k}\right) \quad k \in \{0, 1, \dots, N-1\} \quad (1)$$

For $\alpha \in \mathbb{N}, 1 \leq \alpha \leq N-1$, let $\phi^{(\alpha)}$ be the feature the measurements of which are the tangent angle differences defined by:

$$\beta_k^{(\alpha)} = \theta_{(k+\alpha) \bmod N} - \theta_k \quad k \in \{0, 1, \dots, N\} \quad (2)$$

Therefore, for each shape Γ , we have a set of features:

$$\Phi = \{\phi^{(\alpha)}, \alpha = 0, 1, 2, \dots, N-1\}$$

The tangent angle ($\phi^{(0)}$) is invariant under translation and scaling and the tangent angle difference is invariant to rotation as well.

Now, we compute statistics for each feature $\phi^{(\alpha)}$, namely the empirical distribution (histograms) of features. In the continuous case the histogram of $\phi^{(\alpha)}$ on shape $\Gamma(s)$ is defined as [20]:

$$H^{(\alpha)}(\Gamma, z) = \int \delta(z - \phi^{(\alpha)}(s)) ds \quad \alpha = 0, 1, \dots, N-1 \quad (3)$$

In the above definition, z is a continuous variable for the feature, e.g $H^{(0)}(\Gamma, 0)$ is the number of points on Γ that have zero angle. δ is the Dirac delta function with unit mass at zero and $\delta(x) = 0$ for $x \neq 0$. For discretized variables we have:

$$H^{(\alpha)}(\Gamma, z) = \frac{1}{N} \sum_{j=1}^N \delta(z - \phi^{(\alpha)}(s_j)) \quad (4)$$

In practice, we discretized a histogram $H^{(\alpha)}(\Gamma, z)$ into m bins as shown in figure 1. Therefore, for each feature $\phi^{(\alpha)}$ the histogram is an m -dimensional vector:

$$H^{(\alpha)}(\Gamma) = (H_1^{(\alpha)}, H_2^{(\alpha)}, \dots, H_m^{(\alpha)})$$

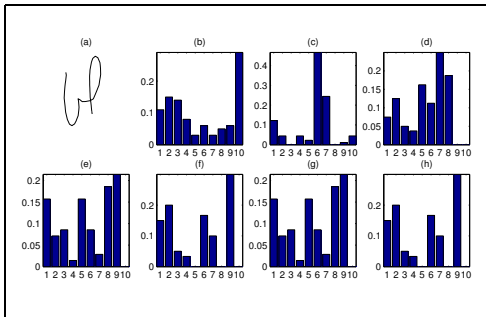


Fig. 1. (a) The letter ‘‘Sad’’ (b) Histogram of tangent angle $H^{(0)}$ (c) - (h) Histograms of tangent angle differences corresponding to $H^{(10)}, H^{(20)}, H^{(30)}, H^{(40)}, H^{(50)}, H^{(60)}$.

Finally, we compose a vector of representation using histograms of all the features:

$$\mathcal{H} = (H^{(0)}, H^{(1)}, \dots, H^{(N-1)}) \quad (5)$$

3. THE KOHONEN MEMORY

The Kohonen neural network [21] (also called Kohonen self organizing map, or SOM), implements an algorithm of the clustering paradigm similar to K-means [22]. It can also be seen as a vector quantizer, mapping data patterns onto a computed set of patterns representative of pattern categories. The nodes in a Kohonen network are organized in a one- or two-dimensional array as shown in Figure 2. The network is an associative memory which encodes the input patterns in the form of weight vectors of the same dimension and nature as the input patterns, stored at the nodes of the network. A characteristic of the Kohonen associative memory is its *self-organizing ordering*: neighboring nodes encode neighboring weight values, creating a ‘topological order’ among nodes.

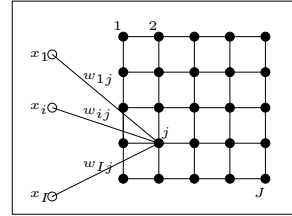


Fig. 2. A two-dimensional Kohonen memory of J nodes.

The version of the training algorithm we used [23] is shown below. Its output is the set of weight vectors $W_j = (w_{1j}, \dots, w_{Ij})$ stored at nodes $j = 1, \dots, J$. After the weights are initialized at small random values, the process consists of finding the node, j^* , that contains the weight vector closest to the current input, X , and updating the weight vector at each node j of the memory by an amount that is a function of the grid distance to the node j^* . Function h^{j,j^*} which defines the influence of node j^* on node j during update at j , decreases with increasing grid distance between nodes j^* and j . It depends on parameter σ which decreases with the number of iterations between the value σ_i and σ_f . ϵ scales weight change and varies with the number of iterations from ϵ_i to ϵ_f . Parameters $\sigma_i, \sigma_f, \epsilon_i$ and ϵ_f must be chosen appropriately to obtain convergence of the algorithm and topological ordering of the network. The Kohonen memory training algorithm can be summarized as follows (Tab.1):

- Initialize weights W_j^0 to small random values, $j \in [1, J]$.
- Get new input $X^n = (x_1^n, \dots, x_I^n)^T$, and compute distances $d(X, W_j)$ to all weight vector.
- Find node j^* with smallest distance.
- Update weights:

$$w_{ij}^{n+1} = w_{ij}^n + \epsilon_n h_n^{j,j^*} (x_i^n - w_{ij}^n) \quad (6)$$

$$\epsilon_n = \epsilon_i \left(\frac{\epsilon_f}{\epsilon_i}\right)^{\frac{n}{n_{max}}}, \quad \sigma_n = \sigma_i \left(\frac{\sigma_f}{\sigma_i}\right)^{\frac{n}{n_{max}}} \quad (7)$$

$$h_n^{j,j^*} = \exp - \frac{\|j - j^*\|^2}{2\sigma_n^2} \quad (8)$$

Table 1. Kohonen memory training algorithm

The Euclidean distance has been traditionally used in the Kohonen memory algorithm. However, we investigated some other more appropriate measures because they measure distance between distributions. The Kullback-Leibler divergence also known as cross entropy and relative entropy is defined by:

$$d_{KL}(X, W_j) = \sum_{i=0}^I x_i \log \frac{x_i}{w_{ij}}$$

The Hellinger distance is given by:

$$d_H(X, W_j) = \sum_{i=0}^I (\sqrt{x_i} - \sqrt{w_{ij}})^2$$

4. EXPERIMENTAL RESULTS

4.1. Data collection and preprocessing

Data collection was done using a digital Wacom Graphire tablet, with a resolution accuracy of 23 points/cm. It has a sampling frequency of 100 points/sec. Arabic has 28 letters in the alphabet based on 18 distinct shapes that vary according to their connection to preceding or following letters (Fig.3). Using a combination of dots and symbols above and below these shapes, the full complement of 28 consonants can be constructed. Our system recognizes 17 distinct classes of shapes because the “Fa” letter and “Qaf” letter are the same except for their position with respect to the base line. The database contains 432 samples of each character, written by 18 writers without constraint, leading to a wide variety of size and orientation. This is by far the largest database of on-line Arabic characters we know of. We are using our database, rather than others, to be able to draw meaningful conclusions regarding the performance of our recognition system.

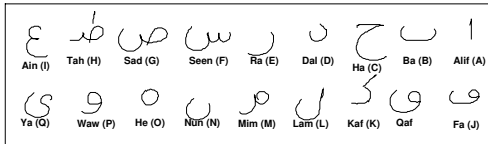


Fig. 3. The 18 shapes of Arabic isolated characters and their assigned labels.

The on-line signal is smoothed and resampled to have equidistant points. Smoothing consists of averaging a points with its neighbors, we used the 3-points average. Resampling is a processing step implemented in almost every on-line handwritten system. In general, points recorded during writing are equidistant in time but not in space. Hence, the number of captured points varies depending on the velocity of writing. To normalize the number of points, the sequence of captured points is replaced with a sequence of points having the same spatial distance. Therefore, all characters will have the same number of points.

The database is divided into two distinct sets. The training set contained 4896 samples and the testing set 2448 samples (which corresponds to 288 samples of each character for training and 144 samples of each character for testing).

4.2. Selection of the number of features

Using all the features $\phi^{(\alpha)}$, $\alpha \in \{0, 1, \dots, N - 1\}$, results in a vector of significantly high dimension (e.g for $N = 100$ and $m = 10$, the dimension is 1000). Therefore, we use a subset of these features that we have choose experimentally. We tested these using memories of 900 nodes (this is a sufficient size, also determined experimentally). Memories were been trained using all the three distances (Section 3). Fig. 4 shows the recognition rates for α 's multiples of 10. The tendency is for the rate, as a function of α , to grow to maximum and then decrease. Following these first experiments, we retained the features for $\alpha = 0, 10, 20, 30, 40$, to compose the vector of representation which, in this case, is of dimension 50.

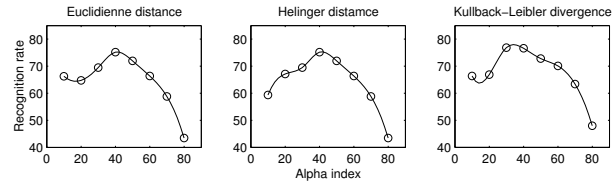


Fig. 4. Recognition rate vs. index features (rates obtained for each histogram individually)

4.3. Recognition rates

In the second experiment, we optimize the number of iterations and the number of nodes. Table 2 summarizes the recognition rates and the optimal parameters for each distance measures. The recognition rate with the Hellinger distance and the Kullback-Leibler divergence are close, and better than the recognition rate with the eucliden distance. With the Kullback-Leibler divergence, significantly less training iterations and memory nodes are required. Therefore, and particulary in applications when training delay and memory size are important considerations, the Kullback-Leibler divergence is to be favored. The Kohonen map trained with the Kullback-Leibler divergence is shown in Fig.5. The characters in this map are those of the training sample whose representation is closest to the memory content. For lack of space we are not showing the two other memories.

Measures	Number of iterations	Number of nodes	Recognition rate
Eucliden	60	361	94.07 %
Hellinger	80	400	94.56 %
Kullback-Leibler	20	256	94.48 %

Table 2. Optimized parameters in Kohonen memories

4.4. Comparison to the nearest neighbor classifier

The nearest neighbor classifier is often used as benchmark. It has excellent practical performance, at least as good as that of neural networks [24]. Its asymptotic error rate is less than twice the optimal Bayes rate. However, *its asymptotic throughput is zero*. Although heuristic pruning algorithms can reduce the size of a reference data set, the nearest neighbor classifier remains significantly slower than neural networks.

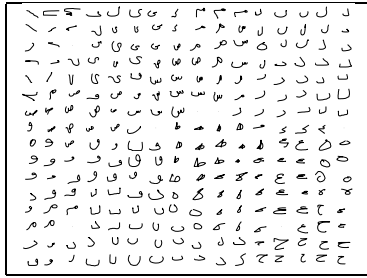


Fig. 5. The trained Kohonen map with the KL divergence

We implemented a nearest neighbor classifier using the three measures of distances: the Euclidian distance, the Kullback-Leibler divergence, and the Hellinger distance. We ran it using the same database as before. The database is divided, as before, into two distinct sets. The set of prototypes contained 4896 samples and the testing set 2448 samples. The recognition rates obtained with our system for the Kullback-Leibler divergence and the Hellinger distance (Table 2) are only slightly lower. Considering memory requirement and speed of execution, these rates support the conclusion that the proposed classifier has excellent performance.

Measures	Recognition rates
Euclidien	95.26
Hellinger	95.09
Kullback-Leibler	95.34

Table 3. Recognition rates using the nearest neighbor classifier

5. CONCLUSION

The aim of this paper was to develop a new representation of shape and to use it in handwritten on-line Arabic character recognition. We investigated statistics of feature based on histograms of tangent angles and tangent angle differences. Recognition was carried out by an associative memories in which we investigated several distance measures. Experimental results show the high performance of the proposed scheme and the pertinence of the representation.

6. REFERENCES

- [1] A. Amin, "Off line arabic character recognition - a survey," in *ICDAR*, Germany, August 1997, vol. 1, p. 596.
- [2] A. Amin, "Off-line arabic character-recognition: The state of the art," *PR*, vol. 31, no. 5, pp. 517–530, May 1998.
- [3] C.C. Tappert, C.Y. Suen, and T. Wakahara, "The state of the art in on-line handwriting recognition," *IEEE Trans. on PAMI*, vol. 12, no. 8, pp. 787–808, August 1990.
- [4] E. Anquetil, *Modélisation et reconnaissance par la logique floue: application la lecture automatique en-ligne de l'écriture manuscrite omni-scripteur*, Ph.D. thesis, Université de Rennes I, 1997.
- [5] T. Hasegawa, H. Yasuda, and T. Matsumoto, "Fast discrete hmm algorithm for on-line handwriting recognition," in *ICPR*, Barcelona, Spain, September 2000, vol. 4, p. 4535.

- [6] T. Klassen, "Towards neural network recognition of handwritten arabic letters," M.S. thesis, Dalhousie University, Halifax, 2001.
- [7] X. Li and D.-Y. Yeung, "On-line handwritten alphanumeric character recognition using dominant point strokes," *Pattern recognition*, vol. 30, no. 1, pp. 675–684, June 1998.
- [8] M. Schenkel, I. Guyon, and D. Henderson, "On-line script recognition using time delay neural networks and hidden markov models," vol. 2, pp. 637–640, 1994.
- [9] M.C. Su, E. Lai, and C.Y. Tew, "A som-based fuzzy system and its application in handwritten digits recognition," in *International Symposium on Multimedia Software Engineering*, Taipei, Taiwan, 2000, p. 253.
- [10] M. Sabourin and A. Mitiche, "Optical character recognition by a neural network," *Neural Networks*, , no. 5, pp. 843–852, 1992.
- [11] M. Sabourin and A. Mitiche, "Modeling and classification of shape using a kohonen associative memory," *Neural Networks*, , no. 6, pp. 275–283, 1993.
- [12] N. Mezghani and A. Mitiche, "Reconnaissance des caractres manuscrits par rseau neuromimétique de kohonen," *Journal Marocain d'Automatique, d'Informatique et de Traitement du Signal*, vol. 3, no. 2, 2001.
- [13] N. Mezghani, A. Mitiche, and M. Cheriet, "On-line recognition of handwritten arabic characters using a kohonen neural network," *Proc. 8th IWFHR*, pp. 490–495, 2002.
- [14] N. Mezghani, M. Cheriet, and A. Mitiche, "Combination of pruned kohonen maps for on-line arabic characters recognition," *7th ICDAR*, vol. 2, pp. 900–905, august 2003.
- [15] T.S. Al-Sheikh and S.G El-Taweel, "Real-time arabic handwritten character recognition," *PR*, vol. 23, no. 12, pp. 1323–1332, 1990.
- [16] S. Marukat, R.Sicard, T. Artires, and P. Gallinari, "A flexible recognition engine for complex on-line handwritten character recognition," in *7th ICDAR*, Edinburgh, Scotland, august 2003, vol. II, pp. 1048–1051.
- [17] G. Seni, *Large Vocabulary Recognition of On-line Handwritten Cursive Words*, Ph.D. thesis, the state university of New York at Buffalo, 1995.
- [18] N. Mezghani, A. Mitiche, and M. Cheriet, "Reconnaissance en-ligne de caractères arabes manuscrits par un rseau de kohonen," *Proc. VI 2002*, pp. 186–191, 2002.
- [19] R. Plamondon, "Handwriting generation: the delta lognormal theory," *Proc. of IWFHR*, pp. 1–10, 1994.
- [20] C.S. Zhu, N.Y. Wu, and D. Mumford, "Minimax entropy principle and its application to texture modeling," *Neural Computation*, vol. 9, no. 8, November 1997.
- [21] T. Kohonen, *Self organizing maps*, Springer, Berlin, 1995.
- [22] P. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, New York, 2001.
- [23] H. Ritter and K. Schulten, "Kohonen's self-organizing maps: exploring their computational capabilities," in *Proc. IEEE Int. Joint Conf. On Neural Networks*, San Diego, USA, 1988, pp. 109–116.
- [24] A. Mitiche and J.K. Aggarwal, "Pattern category assignment by neural networks and the nearest neighbors rule," *IJPRAI*, vol. 10, pp. 393–408, 1996.