Invariant Primitives for Handwritten Arabic Script: A Contrastive study of four feature sets

Sofiene Haboubi
Samia Maddouri
Noureddine Ellouze
Image Processing and Pattern Recognition Unit
National Engineering School of Tunis (ENIT)
BP 37, Belvedere 1002, Tunis, Tunisia.
sofiene.haboubi@istmt.rnu.tn;
{samia.maddouri,noureddine.ellouze}@enit.rnu.tn

Haikal El-Abed
Institute for Communications Technology
Department of Signal Processing for Mobile
Information Systems
38106 Braunschweig,Germany
elabed@tu-bs.de

Abstract

The choice of relevant features is very decisive in handwriting recognition rate. Our aim is to present some useful structural and statistical features and see their degree of variability. In this paper, we start with a description of the variability of the Arabic handwriting and the way how to reduce it. Four kinds of feature sets used by our handwriting systems are then presented evaluated and discussed. The comparison is carried on a database of images from IFN/ENIT databases. The Neural Network Multilayer perceptrons is our method of classification. A contrastive study of these primitives is done according to recognition their time and memory consuming and their variability degree.

1 Introduction

The choice of primitives is very crucial for the handwriting recognition step. The main purpose is to and an optimal set of features that would properly partition the space and achieve classification. A compromise should be respected during primitive extraction: The extractor should provide uniformed primitives for different kinds of scripts. These primitives should describe all details necessary for recognition process. Further, it should not be time or memory space consuming [3]. Several structural primitives extraction methods are presented in literature as [9, 11]. These primitives are based on three principal methods: image projection [6, 12], skeletization [1, 10, 14] and contour function [13, 17, 18, 19]. Projection method is the simplest and the least time and memory space consuming for images with a little size but it is very sensitive to slant and it can not give a precise description of the image. Skeletization is generally used for script segmentation in uniform graphemes. Cursive Script segmentation difficulties reduce the efficiency of skeleton description. The contour which globally describes the image is easy to be detected. By ulterior processing steps we can extract different kind of structural, analytic, global or local primitives from the image contour. This kind of description is most used in our proposed structural and analytic or statistical descriptors. Handwriting Arabic words recognition becomes a complex task because of the diversities of script styles on the one hand and the great quantities of information given by the Arabic word on the other hand. In this paper we present in section 2 a state of the art on the main useful primitives for handwritten Arabic script. In section 3 we describe four primitives' extraction methods. In section 4 we evaluate and compare the extraction rate of each method on IFN/ENIT database [20].

2 Invariance of descriptors

A printed text without noise can be considered as an invariant model of characters. While the handwriting is regarded as a high variability model, there are several types of variations which depend on position, size, rotation, slope and distortions of the form. This variability is one of the greatest difficulties of the handwritten script description and recognition systems. The normalization methods have an important role in the limitation of these variations. A second solution to reduce the variability of the handwriting is to generate or choose invariant descriptors. An invariant descriptor is denned by having the same descriptors from two different objects of the same pattern. Indeed, an object preserves its form when it changes positions and orientation. This concept of invariance has been the subject of several works [16]. The rebuilding of the forms by invariant de-

978-0-7695-3725-2/09 $25.00 © 2009 IEEE
DOI 10.1109/ICDAR.2009.281

scriptors is significant in some applications. A case in point, the regeneration of standardized images allows us to reduce the cost of training step in the recognition systems sensitive to variations.

2.1 Handwritten Arabic script features

Primitives can be classified into two categories: global features and local ones. Each kind of these categories has their advantages and disadvantages.

Global primitives The global observation of images is inspired from human perception. It reduces the difficult problem of handwritten script segmentation. Based on an adequate choice of global descriptors and a robust extraction method, global primitives can be insensitive to noise and script variability. This is one of the advantages of word global description by visual primitives because the word is considered as one entity. During our bibliographic study on global primitives, the most used primitives are structural ones, namely, ascender, descender, loop and the number as well as the position of diacritic points [1, 16]. The problem with this reduced number of global structural primitives is that some words do not contain any of these primitives, which makes their processing by the associated recognition system impossible.

Local primitives The solution proposed by some researchers is to add local primitives as valleys, contour or skeleton and concavity number. The problem with these secondary primitives lies in the difficulty of their extraction, particularly for handwritten script. Local description is necessary when the character does not hold any of the above primary features, or if these features are wrongly detected (such as the open or hidden loop). However, the main problem with local processing is the pre-segmentation step [8]. Handwritten words can be described globally by taking into consideration all pixels of the image, or by the use of some statistical descriptors such as Fourier descriptors, Gabor filter, wavelet, to name but a few. The invariability of these primitives is assured by pre-processing and normalization step.

2.2 Normalization methods

The recognition system is characterized by three stages: pre-processing, feature extraction, recognition and post-processing, in some cases. The image normalization can be introduced in pre-processing or post-processing step. It consists in ensuring a strong correlation between images containing the same pattern. Indeed, the majority of the recognition methods require a training step followed by a recognition step. The training step needs a great number of images of the same pattern to be recognized. The greater the number of these images is, the better the recognition rate. And the more these forms are strongly correlated, the less the recognition errors. In literature, several normalization methods are developed such as linear and non-linear normalization, HMM normalization, geometrical normalization and Fourier Transform normalization.

Several standardization methods exist in literature. These various techniques tend to standardize isolated forms such as Asian characters, Latin printed number or characters. Standardization, in this case, can be independent of the context. In Arabic and Latin scripts, normalization tends to reduce distance between new form and an already known reference. The aim is to reduce the variability of handwriting for a better recognition rate.

3 Features extraction methods

Features extraction methods can be divided into two categories: structural features and statistical features. Structural features highly tolerate distortions and variations in writing styles but their extraction from images is not always easy, requiring further research endeavors. Statistical features can be, in general, easily and quickly extracted from a text image. They can tolerate moderate noise and variation, and the system can be trained automatically.

3.1 Structural features

The structural features that are already used comprise: ascenders, descenders and loops. We add the position of diacritic dots so as to take into account the specificity of the Arabic script. The shape of these primitives depends on their position in the word. This position is also considered as a feature and added to one of the four previous primitives. When we do not detect any of the described characteristics, we generate the primitive ”Nothing” which is illustrated by ”R” and considered useful to process words without global feature. The primitive ascender is described by the letter ”H”, the descender by the letter ”J”, the loop by ”B”, the upper diacritic point corresponds to letter ”P” and lower diacritic point to letter ”Q”. Four primitive positions are designed by ”D” for beginning, ”M” for middle, ”F” for the end position and ”I” for Isolated. The number and the order of the PAW in the word are also considered as structural features that can contribute to the recognition step.The detection particularities of these features are summarized in the following points:

Baseline detection In order to improve the usual methods, we add a global normalization step based on FT (Fourier Transform) to correct the slant [15]. In the case of ambiguities of the projection method caused by the existence of many ascenders, descenders or diacritic points, we combine it with the local minima and maxima method. Extrema,

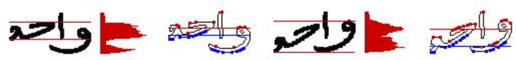
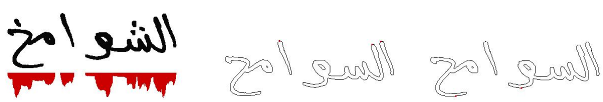


Figure 1. Improvement of the baseline projection extraction method by the minima and maxima of the contour

above lower baseline or below upper baseline deduced from the horizontal projection, are eliminated. From the others extreme points, new baselines are generated. Figure 1 illustrates the result automatically generated by our system.

Ascender and descender extraction Here, ascender and descender correspond to the global minima and maxima of outer boundary points as shown in figure 2. The idea is not to segment the word in letters but to delimit only key letters having apparent primitives such as ascender and descender. The Maxima and Minima and the projection methods are used in this step. A hybrid method is also proposed. The projection method has a problem with the existence of diacritic points and overlapping between descenders. However, it is used as a first estimation method of the letter limit zones. In order to resolve these problems, we start with the elimination of the diacritic points. The problem of overlapping is reduced by the use of the method based on the minima and maxima of the contour.

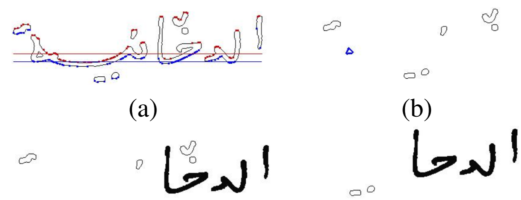


(a) (b) (b)

Figure 2. (a) Estimation of ascender and descender position by projection methods (b) extraction of ascender by global maxima (c) extraction of descender by global minima

Loops and diacritic dots A classical contour extraction method is used for loops and diacritic. The distinction between both is based on the use of a threshold for the boundary size, their relative position compared to the baseline and the verification of the inclusion between closed contours for the characterization of loops. Figure 3 is a clear illustration of the extraction process of diacritic signs above and below baseline as well as a loop in the middle of the word.

Position of primitives (characters) The shape of an Arabic character depends on its position in the PAW. This position is detected during feature extraction step. Indeed, extracted zones are already delimited by local minima. These minima are deduced from vertical projection and minima of the contour. The number of black pixels is computed at the limit of feature extracted zones and between the two baselines. According to this number, we can deduce the position of the primitive. By a way of illustration, if this number is equal to 0 at the two limit zones, the letter is considered as isolated. Due to the overlapping between letters, extraction of structural features and their positions is separately carried out on each PAW. Figure 4 depicts this process.



(a) (b)

(c) (d)

Figure 3. (a) Baseline extraction (b) loop extraction (c) distinction between loops and upper diacritic dots (d) distinction between loops and lower diacritic dots

(a) (b) (c) (d)

Figure 4. (a) Original image (b) Structural feature extraction of the first PAW: PM and JF (c) Structural feature extraction of the second PAW : BPI (d) superposition of feature extraction of PAWs.

3.2 Statistical Features

Statistical features are numerical measurements of spatial distribution of image pixels [5]. They are computed over images or regions of images.They include Fourier descriptors, Gabor filter, wavelets [4], pixel densities [8], histograms of chain code directions, among many others. These features can also be deemed global if they are used to describe the whole word or a set of PAWs. They can be considered as local descriptors in case they are used to describe details after a segmentation step. Two statistical features are used in our work, namely, normalized Fourier descriptors and Gabor filter.

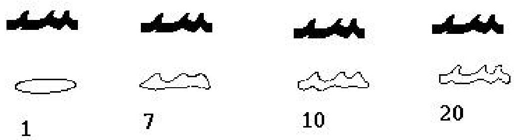


Figure 5. Superposition of Fourier Descriptors for character "ssin".

Fourier Descriptors Fourier Descriptors (FD) can be normalized to be in- variant to translation, rotation, size and starting point [2]. The normalization approach uses the boundary function of the characters pre-segmented from the image. It is based on the method developed by Kuhl. Freeman chain code is generated from the closed boundary function. A Fast Fourier Transform (FFT) algorithm calculates the FDs (a_n, b_n, c_n, d_n, A_0 and C_0) of a chain-coded contour defined by equation (1), with N being the number of boundary points [13]:

$$\begin{aligned} X_N(k) &= A_0 + \sum_{n=1}^{N} a_n \cos\frac{2n\pi k}{N} + b_n \sin\frac{2n\pi k}{N} \\ Y_N(k) &= C_0 + \sum_{n=1}^{N} c_n \cos\frac{2n\pi k}{N} + d_n \sin\frac{2n\pi k}{N} \end{aligned} \tag{1}$$

From this equation, we can deduce that the original contour is approximated by the addition of the elliptic loci of the projections points ($X_n(k)$, $Y_n(k)$), as presented in figure 5.
The normalization is performed according to various elliptic properties of the FDs.

- Position normalization: FDs are invariant to translation if we ignore the bias terms A_0 and C_0.
- Starting point transform: The starting point angular rotation θ_1 is determined from the point (x_1; y_1) with elliptic loci by:

$$\begin{aligned} x_1 &= a_1 \cos\theta + b_1 \sin\theta, \\ y_1 &= c_1 \cos\theta + d_1 \sin\theta, \\ \theta &= 2\pi k/N \end{aligned} \tag{2}$$

The magnitude of the first harmonic phase is:

$$E = \sqrt{x_1^2 + y_1^2} \tag{3}$$

By differentiating the magnitude of the first harmonic phase and setting the derivative equal to zero, we obtain:

$$\theta_1 = \frac{1}{2}\arctan\left[\frac{2(a_1 b_1 + c_1 d_1)}{a_1^2 + b_1^2 - c_1^2 - d_1^2}\right] \tag{4}$$

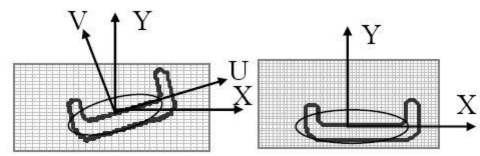


Figure 6. Harmonic phase normalization.

This expression locates the first semi-major axis to occur moving away from the starting point in the direction of the rotation around the contour [15, 13]. Harmonic angular normalization:

- To have FD invariant to rotation, we should move the first harmonic angular until it is aligned with the semi-major axis of its loci. Figure 6 illustrates this step of normalization. The spatial rotation is determined from the Fourier coefficients a_1^* and c_1^* described in formula (5) and which are corrected for starting point displaced θ_1 given by formula (4).

$$\begin{aligned} x_1(k^*) &= a_1^* \cos(2\pi k^*/T) + b_1^* \cos(2\pi k^*/T) \\ y_1(k^*) &= c_1^* \cos(2\pi k^*/T) + d_1^* \cos(2\pi k^*/T) \end{aligned} \tag{5}$$

The harmonic phase Ψ_1 is obtained as:

$$\Psi_1 = \arctan\frac{y_1^*(0)}{x_1^*(0)} = \arctan\frac{c_1^*}{a_1^*} \tag{6}$$

- Size normalization: The normalization of the size can be made by dividing each of the coefficients by the magnitude of the semi-major axis defined as:

$$E^*(0) = \sqrt{(x_1^*(0))^2 + (y_1^*(0))^2} = \sqrt{(a_1^*)^2 + (c_1^*)^2} \tag{7}$$

The proposed transformation tries to reduce the first harmonic size of all letters to 1.

Gabor filter A Gabor filter is a sinusoidal function modulated by a Gaussian envelope. Such a sinusoidal function is characterized by its frequency and its orientation [21, 22]. Thus, a Gabor filter can be seen as a detector of particular edges of orientation since it reacts to the edges perpendicular to the direction of propagation of the sine [23]. Filtering by Gabor preserves the temporal and frequencies aspects signal. In the space field, the application of the filters of Gabor is carried out by calculating the convolution of the image with a function regulated to one of textures. The process is shown in equation (8).

$$q(x,y) = p(x,y) \times h(x,y) \quad (8)$$

$q(x,y)$ is the filtered pixel of the image result; $h(x,y)$ is the Gabor filter; $p(x,y)$ is the original pixel of the image.

Theoretically, we can make all calculations in the frequencies field; the product of convolution is reduced to a simple multiplication transforms of Fourier. The process is shown in equation (9).

$$q(x,y) = TF^{-1}(P(u,v)H(u,v)) \quad (9)$$

with $P(u,v) = TF(p(x,y))$ and $H(u,v) = TF(h(x,y))$

The 2d Gabor filter is represented by equation (10).

$$h(x,y) = g(x',y') \exp(2\pi i f x') \quad (10)$$

with $g(x',y')$ is the 2d Gaussian function given by formula (11).

$$g(x',y') = \exp\left[\frac{-\frac{1}{2}(x'^2 * y'^2)}{\sigma^2}\right] \quad (11)$$

With $(x',y') = (x\cos\theta + y\sin\theta, -x\cos\theta + y\sin\theta)$, the turned (x,y) co-ordinates of an angle θ.

The parameters F and θ represent the frequency and the sinusoidal orientation of the signal and constitutes the parameter of the space of the Gabor filter. In the space field, the Gabor function is represented by equation (12).

$$\begin{aligned} H(u,v) = \exp\left[-2\pi^2\sigma^2((u-f)-v^2)\right] \\ +\exp\left[-2\pi^2\sigma^2((u+f)+v^2)\right] \end{aligned} \quad (12)$$

3.3 Pixel Representation

The pixel representation is generated after normalization steps. The first step is a rotation of the word resulting in a horizontal baseline. Subsequently a vertical height normalization is done using a linear characteristic between topline and baseline and a nonlinear characteristic elsewhere, this results in constant heights for ascender and descender regions and thus in a fixed total height of the resulting skeleton graph. Next a horizontal width normalization with a linear characteristic is performed, yielding a word with constant average character width. The line thickness is normalized during the generation of the skeleton. Finally a rethickening is done by a Gaussian filtering of the normalized skeleton image, resulting in a gray level image.

The pixels describing an image is generated using a sliding windows. This window is shifted in respect to the Arabic writing direction from right to left across the normalized gray level script image and generates a feature vector (frame).

4 Evaluation and Comparison

The evaluation stage is undertaken on time and memory consuming, the kind of recognizer, the complexity and the variability degrees of each of the four primitives already described in section 3. Figure 3 presents pixels description of a handwritten word coming from IFN/ENIT database. This image is described by (433*132)=57156 pixels. IFN/ENIT contains only binary images. So, each 8 pixels are described by one byte. The image size of figure 3 is then (57156/8)= 7145 bytes as presented in table 4. The size of the header file is neglected.

In table 1, we give a structural description of the same handwritten word in figure 3. This description starts with the number of PAWs or connected components (4). Then, we assign the combination of structural description for all the letters of each PAW preceded by the PAW order.

Table 2 presents the 48 statistical descriptors deduced from the average and the variance type of 4 frequencies and 6 orientations which explain the number 48=2*6*4. To have the size in bytes, we should multiply 48 by the number of bytes needed for the representation of oat value (4 bytes). The result of this multiplication is given in table 4. The choice of the number of frequencies and orientation depends on the complexity of the image to recognize. For IFN/ENIT database words, the preceding numbers are sufficient. Table 3 describes the complexity of Fourier Descriptors based on the application of the Fast Fourier Transform (FFT) on a closed contour. So, for Arabic word such as figure 3 each connected component or PAW is processed separately. We have 4 coefficients (a_n, b_n, c_n, d_n) corresponding to four PAWs of figure 3. The number of harmonic depends on the complexity of the image to describe. For IFN/ENIT database we have used 32 harmonics. In table 3 we give only the first 16 harmonics to reduce the size of this table. The number of bytes needed to describe an IFN/ENIT database image is given by: 4PAWs * 4coefficients * 32harmonics * 4bytes. Table 4 presents the obtained result.

From these features we can see the complexity and the memory consuming of these primitives. Structural primitives have the least size. However, the least time consuming corresponds to the FFT but the size is greater than Gabor Filter descriptors. Gabor Filter Has the greatest time consuming.

As mentioned earlier, recognition systems rate and rapidity depend on the quality of feature set. Overall, structural and statistical features are drawn upon by all recognition systems.

The base of documents used in the phase of training and

Figure 7. 433×132 **pixels representation of a handwritten Arabic word.**

Table 1. Structural description of the image of the word in figure 7

PAW Nb	PAW1	PAW2	PAW3	PAW4
4	HI	HD HM BJF	HI	PD BPF

test, is that of the IFN/ENIT [20]. From this base we used 4 sets (Set a, b, c and d).

The 4 sets give 26461 images. This base contains a samples of 945 names of Tunisian Towns. Considering the importance of the number of samples for each Town name in the phase of training and test, we give a statistical figure showing the number of samples in this database (Figure 8). We notice in the database, some number of towns not have sufficient samples. For example, there are 31 towns have in the maximum 5 samples, and at the same time, 24 towns have there more than 200 samples. In the phase of training, certainly, the number of samples for each towns has a significant influence on the results of test. For this reason, we choose the town names which have a number of samples more than 50. We found 122 town names, with 16107 images. We given 90% (14497 images) of this database for training and 10% (1610 images) for test. The Neural network is the used method for training and test. The used model of Neural Network is Multilayer perceptrons (MLPs) trained with static backpropagation. This network contains 3 layers: - Input layer: a many neurons equalize with the dimension of the feature vector; - Hidden layer: 122 neurons; - Output layer: 122 neurons. The results of test are posted in the table 5. Currently, We do not have yet results for Pixel Representation with Neural network, but with HMM gives a rate of 87.1%* [7]. Table 5 shows that the Gabor Filter does not give interesting characteristics for recognition (19.2%). For Fourier Descriptors and Structural Approach gives 79.2% and 71.1% respectively.

Table 2. 48 Gabor Filter descriptors of the image of figure 7

0.002	0.021	0.020	0.000	0.020	0.021	0.058	0.050
0.043	0.048	0.071	0.063	0.008	0.000	0.000	0.008
0.000	0.000	0.048	0.010	0.011	0.029	0.014	0.019
0.007	0.000	0.000	0.008	0.000	0.000	0.026	0.003
0.023	0.004	0.006	0.000	0.000	0.000	0.000	0.000
0.000	0.007	0.001	0.001	0.007	0.001	0.001	0.000

Table 3. 64 Normalized Fourier descriptors of the first PAW of figure 7

	PAW 1							
a_n	-1.00	-0.66	-0.29	-0.27	-0.03	-0.11	-0.17	-0.09
	-0.14	-0.10	-0.09	-0.09	-0.13	-0.12	-0.08	-0.14
b_n	0.00	0.47	0.44	0.34	0.12	0.11	0.14	0.19
	0.05	0.04	0.08	0.03	0.09	0.02	-0.01	0.02
c_n	0.18	-0.02	-0.26	-0.43	-0.49	-0.49	-0.46	-0.46
	-0.48	-0.50	-0.51	-0.50	-0.50	-0.51	-0.53	-0.54
d_n	6.56	2.95	1.76	1.26	1.03	0.87	0.72	0.58
	0.47	0.40	0.35	0.29	0.23	0.17	0.11	0.08

Table 4. 256 Normalized Fourier descriptors of the image of figure 7

Feature sets	Memory consuming		Time consuming	
	1Word	IFN/ENIT	1Word	IFN/ENIT
Pixel Rep.	7145	230		
Struct. Feat.	32	2	1.38	12.5
Gabor Filter	192	6	3.74	48
Fourier Desc.	1024	33	0.89	8
Unit	Mega Bits		Second	Hours

Table 5. Results of test

	Correct Classified	Incorrect Classified
Structural Approach	71.1%	28.9%
Fourier Descriptors	79.2%	20.8%
Gabor Filter	19.3%	80.7%
Pixel Representation	87.1%*	12.9%*

5 Conclusion

The aim of this paper is to compare between some features sets. In pursuit of this end, we select four feature sets already used by our systems. These features are based on a direct pixel description (PD), structural description (SD) and two statistical ones GF (Gabor Filter) and NFD (Fourier Descriptors). The third feature set is the direct use of pre-processed pixels of the image PR. SD is based on a combination between 10 character (H, J, B, P, Q, D, M, F, I, R) describing structural shapes of handwriting.

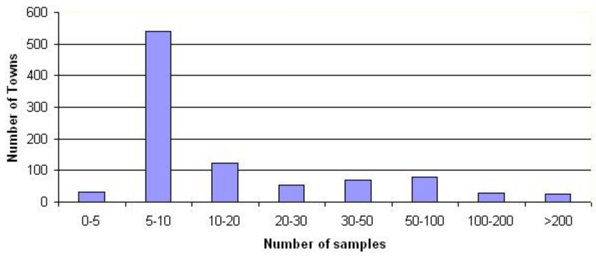


Figure 8. Number of samples by town.

For GFF, we need 48 oat values and for NFD we use 128 descriptors for each PAW of the word. SD needs less memory space and NFD are deduced in the less time. The obtained rates for SD, NFD and PD are interesting, and the three approaches are too different. This factor encourages us to think of a system: who combines between these three.

As perspectives, we need to see other features set and the contrastive study could be done on the impact of each feature set on the recognition rate of a same classifier. We can also see the impact of the pre-processing step on the rate accuracy and deduce which features set tolerate distortions more.

References

[1] A. Amin. Recognition of hand-printed characters based on structural description and inductive logic programming. *Pattern Recogn. Lett.*, 24(16):3187–3196, 2003.

[2] K. Arbter. Affine-invariant fourier descriptors. from pixel to features. *Elsevier Science Publisher B.V (North-Holland)*, 1989.

[3] V. Bouletreau, N. Vincent, R. Sabourin, and H. Emptoz. Synthetic parameters for handwriting classification. *ICDAR '97*, 0:102, 1997.

[4] S. J. V. M. Broumandnia, A. Persian/arabic handwritten word recognition using m-band packet wavelet transform. *Image Computer*, 2007.

[5] A. Chhabra, Z. An, D. Balick, G. Cerf, K. Loris, P. Sheppard, R. Smith, and B. Wittner. High-order statistically derived combinations of geometric features for handprinted character recognition. *ICDAR '93*, pages 397–401, 1993.

[6] M. Coté, M. Cheriet, E. Lecolinet, and C. Y. Suen. Automatic reading of cursive scripts using human knowledge. In *ICDAR '97*, pages 107–111, Washington, DC, USA, 1997. IEEE Computer Society.

[7] H. El Abed and V. Margner. Comparison of different preprocessing and feature extraction methods for offline recognition of handwritten arabicwords. In *ICDAR '07: Vol 2*, pages 974–978, Washington, DC, USA, 2007. IEEE Computer Society.

[8] R. El-Hajj, L. Likforman-Sulem, and C. Mokbel. Arabic handwriting recognition using baseline dependant features and hidden markov modeling. In *ICDAR '05*, pages 893–897, Washington, DC, USA, 2005. IEEE Computer Society.

[9] N. Farah, L. Souici, and M. Sellami. Classifiers combination and syntax analysis for arabic literal amount recognition. In *Engineering Applications of Artificial Intelligence 19*, 2006.

[10] M. Hamanaka, K. Yamada, and J. Tsukumo. On-line japanese character recognition experiments by an off-line method based on normalization cooperated feature extraction. *ICDAR '93*, 200-203, 1993.

[11] M. Khorsheed. Recognising handwritten arabic manuscripts using a single hidden markov model. *Pattern Recognition Letters 24*, 14:2235–2242, 2003.

[12] S.-Y. Kim and S.-W. Lee. Nonlinear shape normalization methods for gray-scale handwritten character recognition. In *ICDAR '97*, pages 479–482, Washington, DC, USA, 1997. IEEE Computer Society.

[13] F. Kuhl. Elliptic fourier features of a closed contour. *Computer Graphics and Image processing 18*, pages 236–258, 1982.

[14] L. Lam and C. Y. Suen. Structural classification and relaxation matching of totally unconstrained handwritten zipcode numbers. *Pattern Recogn.*, 21(1):19–32, 1988.

[15] S. S. Maddouri, H. Amiri, and A. Belaid. Local normalization towards global recognition of arabic handwritten script. *DAS '00*, 2000.

[16] S. S. Maddouri, E. Yahia, and H. Amiri. Les descripteurs invariants de l'ecriture arabe manuscrite. *TAIMA '01*, pages 29–37, 2001.

[17] S. Madhvanath and V. Govindaraju. Contour-based image preprocessing for holistic handwritten word recognition. In *ICDAR '97: Proceedings of the 4th International Conference on Document Analysis and Recognition*, pages 536–539, Washington, DC, USA, 1997. IEEE Computer Society.

[18] S. Madhvanath and V. Krpasundar. Pruning large lexicons using generalized word shape descriptors. In *ICDAR '97*, pages 552–555, Washington, DC, USA, 1997. IEEE Computer Society.

[19] H. Nishida. A structural model of continuous transformation of handwritten characters. In *ICDAR '95(Volume 2)*, page 685, Washington, DC, USA, 1995. IEEE Computer Society.

[20] M. Pechwitz, S. Maddouri, V. Margner, N. Ellouze, and H. Amiri. Ifn/enitdatabase of handwritten arabic words. *CIFED '02*, pages 127–136, 2002.

[21] T. N. Tan, H. E. S. Said, and K. D. Baker. Personal identification based on handwriting. In *National Laboratory of Pattern Recognition , Institue of Automation, Chinese Academy of Sciences*, China, December 2000.

[22] T. N. Tan, H. E. S. Said, and K. D. Baker. Personal identification based on handwriting. In *National Laboratory of Pattern Recognition , Institue of Automation, Chinese Academy of Sciences*, China, December 2000.

[23] L. Vincent. Texture segmentation using gabor filters. In *Center For Intelligent Machines, McGill University*, December 2000.