

# Spatial and Textural Aspects for Arabic Handwritten Characters Recognition

Y. Boulid<sup>1\*</sup>, A. Souhar<sup>2</sup>, Mly M. Ouagague<sup>1</sup>

<sup>1</sup> Research and Development Unit, Maxware Technology, Kénitra (Morocco)

<sup>2</sup> Department of Computer Science, Faculty of Science, University Ibn Tofail, Kénitra (Morocco)

Received 4 September 2017 | Accepted 30 November 2017 | Published 19 December 2017



## ABSTRACT

The purpose of the present paper is the recognition of handwritten Arabic characters in their isolated form. The specificity of Arabic characters is taken into consideration, each of the proposed feature extraction method integrates one of the two aspects: spatial and textural. In the first step, a modified Bitmap Sampling method is proposed, which converts the character's images into a binary Matrix and then constructs a Mask for each class. A matching rate is used between the input binary matrix and the masks to determinate the corresponding class. In the second step we investigate the use of an Artificial Neural Network as classifier with the binary matrices as features and then the histograms of Local Binary Patterns to capture the texture aspect of the characters. Finally, the results of these two methods are combined to take into consideration both aspects at the same time. Tested on the Arabic set of the Isolated Farsi Handwritten Character Database, the proposed method has 2.82% error rate.

## KEYWORDS

Handwritten Recognition, Arabic Documents, Local Binary Patterns, Modified Bitmap Sampling.

DOI: 10.9781/ijimai.2017.12.002

## I. INTRODUCTION

**A**RCHIVES constitute the human documentary production; they reflect the activity of individuals and organizations in time.

The conservation of these archives is one of the major goals of the nations since they show their evolution. The use of the information contained in these archives are still a challenge since most documents are still in their original formats (paper) which makes searching and retrieving information a very hard task, even for scanned documents (saved as images on electronic media) because of the need to manually search through images one by one to find a specific information.

With the scientific progress in terms of computing resources of machines and the development of sophisticated algorithms, converting scanned documents to electronic versions has become possible, allowing indexing and searching content using queries feasible. Systems allowing the conversion of printed documents into an editable text are called Optical Character Recognition (OCR).

The conversion of historical handwritten documents into editable text is more challenging due to irregularity and diversity of handwritten styles, and also to the poor quality of the paper. The systems that correspond to this category are known as Intelligent Character Recognition (ICR).

As many object recognition problems, the process of hand written recognition often consists of five stages executed in a linear way: preprocessing, segmentation, data representation (feature extraction), classification/recognition and post-processing.

Each of these stages consists of a number of techniques and methods adapted to specific problems, such as: noise suppression, skew

and slope correction, text lines segmentation, words segmentation, characters recognition, lexical, syntax and semantic verification.

In addition to the complexity of a document, the language of which it is written with has an influence in the performance of the recognition result. For instance recognizing Latin script is much easier than Arabic or other cursive scripts.

Today, Arabic language is spoken by over 300 million people. The written direction is from right to left and the script is semi-cursive which means that some letters may connect to each other.

Depending on their position in the word the letters have shapes that may change, as they are preceded and/or followed by other letters or isolated.

The information about diacritical points is important in differentiating between letters sharing similar shape. The number and/or positions of these points differ, as for the three different letters 'ba', 'taa', 'thaa' (ب, ت, ث) that have the same basic shape but differ in the position and the number of points.

Arabic characters have different spatial disposition, which could be horizontal, vertical, diagonal or other. The texture of a character may change along its shape. The fact that Arabic script is written from right to left influences on the texture of the shape in the beginning which differs from the one in the ending.

We think that in the design of feature extraction methods for Arabic characters, one must take into consideration these two aspects spatial and textural.

The present paper is an extension of our work in [1]. We will first compare the recognition accuracy of two feature extraction methods respecting the spatial and textural aspects, namely modified Bitmap Sampling method and Local Binary Patterns (LBP). In the second step the combination of these two methods is investigated in order to take into consideration both aspects.

\* Corresponding author.

E-mail address: y.boulid@gmail.com

The rest of the paper is organized as follows: Section 2 provides a brief overview of some related works on the recognition of Arabic letters. Section 3 describes the used feature extracted methods, provides results of each one and also investigates their combination, and compares the performance of the proposed method with existing works on IFHCDB dataset. Finally, section 4 concludes the paper.

## II. RELATED WORKS

Generally, in handwritten recognition there are two category of feature extraction: either statistical [2] such as histograms of transition and projection profile, histograms of gray level distribution, Fourier descriptors and chain code; or structural [3, 4] such as the presence of loops, number and position of diacritical points and orientation of curves. Table I presents an overview of some related works dealing with the recognition of isolated Arabic and Farsi handwritten characters.

## III. FEATURES EXTRACTION METHODS WORKS

The challenge in the recognition of the Arabic letters is that the used descriptor must take into consideration the existence of different characters written in a similar way, for example, some character shares the same body shape but differs only in the diacritical points. In the following, we will give details and results of the used feature extraction methods.

### A. Dataset

In [19] a dataset called Isolated Farsi Handwritten Character Database (IFHCDB) was introduced and it was used in the ICDAR 2009 competition [15]. The distribution of characters is not uniform.

In this work, we are interested only in the Arabic portion, which represents approximately 97% of all character set (Arabic and Farsi), and which is divided into 35989 images for learning and 15041 for testing.

### B. Modified Bitmap Sampling (MBS)

In order to take into consideration the spatial aspect of the characters, the first method of feature extraction is called Bitmap Sampling, which allows to sub-sample the character's image in a smaller dimension in order to be used as input into a classifier.

We consider the character as a set of filled and empty cells distributed along a grid. The grid corresponds to the smallest bounding box enclosing the character and its diacritics.

In order to make the descriptor invariant to the size of the characters, the number of lines and columns in the grid is fixed for all characters (Fig. 1).

TABLE I  
AN OVERVIEW OF SOME METHODS OF ARABIC/FARSI HANDWRITTEN CHARACTER RECOGNITION

Ref	Classifiers	Features	Dataset	Recognition Rate
[1]	Artificial Neural Network	Local Binary Pattern, Spatial distribution of pixels	IFHCDB dataset	96.31%
[5]	Inductive learning program to generate Horn clauses	Structural features (lines, curves and loops)	30 samples for training and 10 samples for testing from each character	86.65%
[6]	Back propagation neural network	Statistical and Morphological and features from the main body and secondary components	CENPRMI	88%
[7]	Non-dominated Sorting Genetic Algorithm	Normalized central and Zernike moments features	Isolated handwritten characters of 48 persons	90%
[8]	Neural network	Wavelet coefficients	Isolated Arabic letters from more than 500 writers	88%
[9]	Vector quantization	Statistical feature such as distributive and concavity	3000 of Farsi letters	85.59%
[10]	Support Vector Machine	Fourier descriptor of the projection profile, Moments and centroid distance	1000 Arabic isolated characters	96.00%
[11]	Artificial Neural Network, Support Vector Machine, and Logistic Regression	HOG, GIST, LBP, SIFT and SURF	Isolated Arabic handwritten characters which contain about 8988 letters collected from 107 writers	94.28%
[12]	Support Vector Machine, K Nearest Neighbor, Neural Networks	Zoning and crossing counts	IFHCDB dataset	94.82%
[13]	Support Vector Machine	Under-sampled bitmaps and chain code directional frequencies	IFHCDB dataset	96.68%
[14]	Support Vector Machine, K Nearest Neighbor, Neural Networks	Gradient, shadow, directional, line-fitting, intersection and undersampled features	IFHCDB dataset	96.91%
[15]	Hierarchy of multidimensional recurrent neural networks	Raw data	CENPARMI and IFHCDB datasets	91.85%
[16]	Artificial neural network	Cosine Transform and Discrete Wavelet Transform	A dataset of 5600 characters	79.87%
[17]	Artificial neural network	Number of dots and angles between points in the contour	A Datasets of 39200 of Arabic isolated characters	96.91%
[18]	K Nearest Neighbor, Random forest, Linear discriminant analysis	Feature selection based on Moth-flame optimization including : gradient, right, left, vertical and horizontal projections, number of holes, aspect ratio, number and position of secondaries	CENPARMI dataset	99.25%

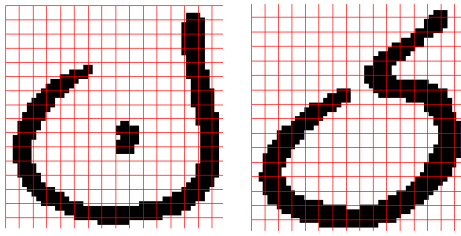


Fig. 1. Example of decomposition of the bounding box of a character (from the IFHCDB dataset) in a grid of 16x16 cells.

From this grid called G, a binary matrix called M with the same size, is generated as in (1):

$$M(i, j) = \begin{cases} 1 & \text{if } G(i, j) \text{ not empty} \\ 0 & \text{else} \end{cases} \quad (1)$$

From the training set and for all the samples of a class, the Matrices of all characters are superposed. The result is called “The Mask of the class”. This Mask contains for each cell, the number of times this cell is filled (Fig. 2).

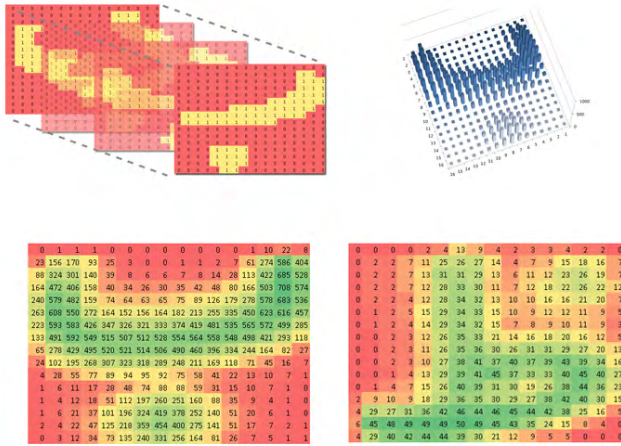


Fig. 2. Example of a 16x16 Mask reconstruction. (a) The superposition of all Matrices of the character (Baa, ب). (b) a 3D histogram of the Mask. (c) and (d) are respectively the resulted Masks of the classes (Baa ب, and thad, ا).

Since the number of samples in the dataset is not the same for all characters, the resulted Masks will contain values of different scales which will influence the result of the comparison. For example, the maximum value in the Mask in Fig. 2.c is 708, while the maximum value in the Mask in Fig. 2.d is 50.

To cope with this issue, the idea is to normalize all the Masks using one scale. Therefore, the algorithm below is used for this purpose.

The normalized versions of the Masks in Fig. 2.c and 2.d, are shown in the Fig. 3. As can be noticed the maximum value in both Masks after applying the normalization algorithm is equal to 256.

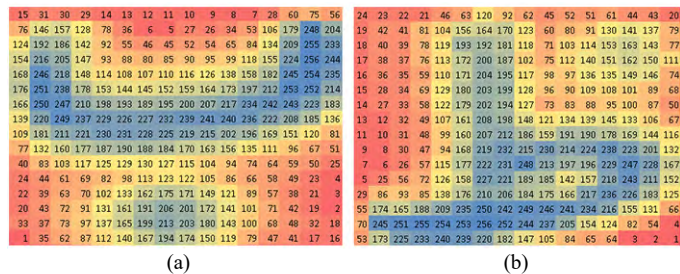


Fig. 3. Normalized Masks for the two characters (Baa ب, and thad, ا).

**Algorithm 1** Normalization of a Mask

```

for row = 1 to WS do
  for col = 1 to WS do
    M[k, 1] = row
    M[k, 2] = col
    M[k, 1] = Mask[row,col]
  end for
end for
OM = SORT(M)
NbrCells = WS * WS
MaxVal = NbrCells
for k = 1 to NbrCells do
  NMask(OM(K, 1), OM(K, 2)) = MaxVal
  MaxVal = MaxVal - 1
end for
    
```

For a given character to be recognized, we extract its binary Matrix and compute the Matching Rate (MR) from all the Masks as in (2):

$$MR_k(B) = \frac{1}{1 + \sum_{i=1}^W \sum_{j=1}^W M_k(i, j) * B(i, j)} / k=1...28 \quad (2)$$

Where  $M_k$  is the Mask of the class  $k$ , and  $B$  is the binary Matrix of the character to be recognized,  $W$  is the number of rows and columns.  $MR_k(B)$  is the amount of participation which measures how the input is close to the Mask of the class  $k$ . Therefore, computed over all the Masks, the corresponding class of an input character is the one with the smallest distance.

Fig. 4, illustrates the TOP3 measures (the percentage of samples that the true class is among the TOP3 positions in the list of candidates) recognition rates according to different configuration of the grid.

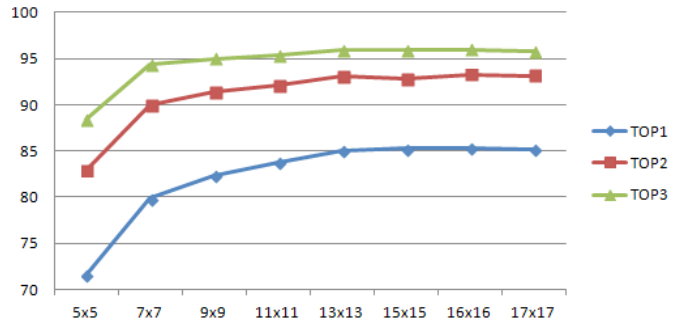


Fig. 4. TOP3 measures for different configurations of the size of the grid.

The higher recognition rate corresponds to the 16x16 configuration, which gives respectively for TOP1, TOP2 and TOP3 the rates 85.29%, 93.29% and 95.98%. This configuration allows certainly to resolve the inter-class problem since a given character will be certainly included in the Mask of some class, but to the detriment of the intra-class problem, since the variability of the shapes of characters allow some masks to be included in others, such as the case of (sin, س and yaa, ي) in Fig. 5.

As alternative, using the Binary Matrices of the characters (Fig. 2.a) as features with a feed-forward neural network with Scaled Conjugate Gradient as training algorithm allows us to reach respectively for TOP1, TOP2 and TOP3 the rates 94.56%, 97.62% and 98.51%.

Based on the result of the confusion matrix we have noticed that this method produces errors between some characters such as (ج, ا), (ل, ن), (ت, ن).

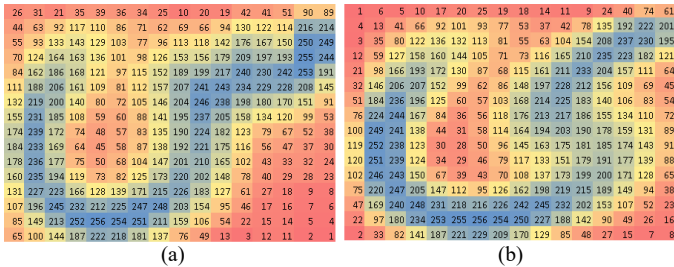


Fig. 5. Example of inclusion of the two normalized Masks of the characters (a) (sin, س) and (b) (ya, ي).

When writing a character, a writer does not put the same pressure at the beginning as the end. Certain characters such as (ح, ط, ض, ش, س) require more vigilance from the writer as other character like (ن, ر, د, ل) because they contain more details which result in changing the texture of their shape. This information is not captured by the Modified Bitmap Sampling method, hence the need to use another method that integrates this aspect.

The next section gives a brief overview of Local Binary Patterns method which is used to capture the textural aspect.

### C. Local Binary Patterns (LBP)

Developed by Ojala et al [20], LBP for Local Binary Pattern, is a descriptor designed to analyze textures in terms of local spatial patterns and gray level contrast. In its basic form, a window is used to produce labels for each pixel by thresholding to neighboring pixels with the center pixel in the window and considers the result as a binary number as illustrated in Fig. 6.

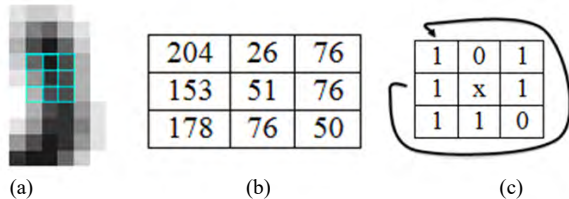


Fig. 6. Example of LBP computation: (a) a 3x3 window centered on a pixel, (b) the gray levels in the window, (c) thresholding the neighborhood compared to pixel x.

As in [1], the use of LBP histogram extracted from the smallest rectangle containing the character and its diacritical points with a feed-forward neural network and a Scaled conjugate Gradient as training algorithm gives a recognition rate of 88.47% which is reasonable, since in the computation of LBP Histogram the spatial information is lost.

So in order to integrate the spatial aspects, as in [1], we simply split the bounding box of the character into four regions from the center point between the centroid of the character and the centroid of diacritics and extract the histogram from each region and then concatenates them to construct the feature vector (Fig. 7). This way allows to discriminate between certain characters that have the same termination.

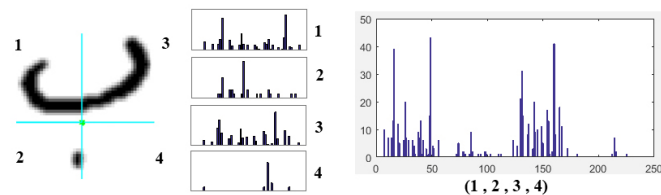


Fig. 7. Concatenation of the histograms extracted from the four regions from the centroid of the character and its diacritics (taken from [1]).

Using these histograms feed-forward neural network with Scaled Conjugate Gradient as training algorithm allows to increase the

recognition rate up to 96.31%, 98.91% and 99.44% respectively for TOP1, TOP2 and TOP3 which proves that the integration of both spatial and textural aspect is beneficial. Table II shows the recognition rates for some characters for the two feature extraction methods.

TABLE II  
RECOGNITION RATES OF SOME CHARACTERS ACCORDING TO MBS AND LBP

Recognition rate (%)	ر	ا	ل	ن	ت	س
MBS	91.67	98.60	92.00	94.08	92.02	96.50
LBP	95.23	99.57	95.33	97.21	90.85	96.50

For the case of ت and س the spatial aspect discriminates better than LBP. This is related to the way the characters are written and to the speed of writing. Some details of the character such as texture may be ignored and therefore the spatial shape can be effective in these cases.

From these results, we can say that the spatial aspect deals with the global shape of the character, but ignores the details which could be captured by the textural aspect. While for LBP the calculation of the histogram loses the spatial information which we compensate by considering four regions of the character.

In what follows we investigate another way of combination that combines the results of the classifiers based on these features.

### D. Combination of Classifiers

The goal of combining classifiers is to conceive a more efficient one which operates on the same data as the base classifiers and separates the same type of classes [21]. As stated in [21] if we denote the score assigned to the class  $i$ , by the classifier  $j$ :  $S_{ij}$ , Therefore, a combination rule is a function  $f$ , and the final combination of the scores of the class  $i$  is in (3):

$$S_i = f(\{S_{ij}^j, j=1 \dots M\}) \quad (3)$$

Then the sample is classified as belonging to the class with the maximum score. In this work we have two classifiers of the same type (neural networks) that are trained with different features. So we are interested in non-ensemble combinations [21].

Since we have the scores of the classifications for each classifier, we can use basic combination schemes at the score level, by applying simple rules such as: the maximum, average, multiplication. For an input image, respectively the maximum, the average and the multiplication rule simply calculates the maximum, the average, the product of the scores of each classifier for each class, and assigns the label of the class having the maximum score.

Table III shows the recognition rates according to the combination of MBS and LBP.

TABLE III  
RECOGNITION RATES ACCORDING TO DIFFERENT COMBINATION RULES

Combination (%)	Maximum	Average	Multiplication
MBS + LBP (entire image)	94.32%	94.47%	95.75%
MBS + LBP (split image)	96.84%	96.95%	97.18%

The multiplication rule applied to the scores of the MBS classifier and the classifier based on LBP after splitting the image gives the best recognition rates which is about 97.18 %, 99.19% and 99.54% respectively for TOP1, TOP2 and TOP3.

E. Comparative Analysis

Is difficult to compare the performance of the proposed approach with most of related works since most existing datasets are not public or not representative (small number of characters). Therefore a comparative analysis (Table IV) is conducted in order to compare the proposed approach with the existing systems working on the same dataset (IFHCDB) when considering 28 (Arabic set), 32 and 33 (Arabic and Farsi sets) class problems.

TABLE IV  
COMPARISON OF THE PROPOSED SYSTEM WITH OTHER METHODS ON IFHCDB DATASET

Method	Training size	Testing size	Number of classes	Reco rate (%)
[12]	36000	13320	33	94.82
[13]	36682	15338	32	96.68
[22]	36682	15338	32	96.91
[1]	36437	15233	32	95.87
[1]	36437	15233	33	96.04
[1]	35989	15041	28	96.31
Proposed method	36437	15233	32	96.51
Proposed method	36437	15233	33	96.50
Proposed method	35989	15041	28	97.18

F. Discussion

As can be seen from the result of the combination rules, taking into consideration both spatial and textural aspect improves more the recognition rate. This confirms that the two aspects are complementary since both classifiers participate equally without favoring one over the other as in the case of the maximum and the average rules.

As can be seen from the comparison the proposed method is competitive. The analysis of the confusion matrix (Fig. 8) shows that most errors are due mainly to :

- The confusion between different characters drawn in a similar way such as the case of (ر , ل), (ع , ح) and (ص , س).
- The presence of illegible characters due to the noise.
- The presence of misclassified characters due to human errors of classification during the assembly of the database.

Fig. 9 shows some of unsuccessful recognition of the proposed method which proves that even humans may find difficulty to recognize some of them.

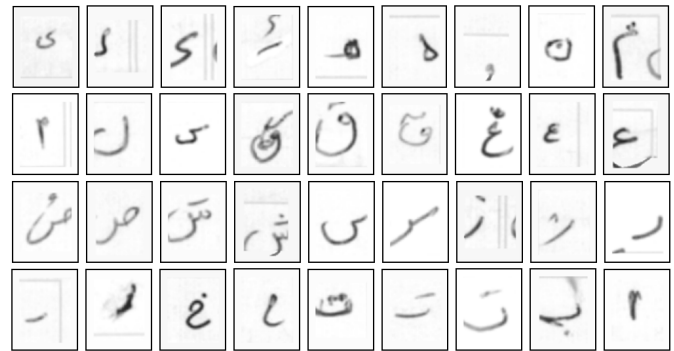


Fig. 9. Example of some misclassified characters.

	ا	ب	ت	ث	ج	ح	خ	د	ذ	ر	ز	س	ش	ص	ض	ظ	ط	ع	!	"	#	\$	%	&	'	( )
ا	2992						4		2													1			1	
ب		39						2	1					1								1			1	
ت			46	1															4	3	5			6	1	
ث			4																							
ج					149	16	1											2								
ح						364											5									
خ					2		12										1	3								
د	2							791		5																
ذ									7		2								1	2						
ر	37	4					2	3	1245	2	1													6	2	
ز			1				1	7	3	322		1														
س									1	7			5								1				7	
ش									1		3	16	1	1												
ص									1	1	23		114	1								1				
ض												6		12												
ظ															59	1										
ط															2	13										
ع	1				1	12												241	3					1	2	
!				1			4											1	36							
"																			268		5				2	
#				2								3							5	126	1		1		3	
\$										1		1									169	1			9	
%	3	1					2	1													1	858		34		
&	3				1			1						1									1483	1	1	
'			7	1				3										1	4	2	1	4		112	5	
( )								5	1	3														397		
*		1	1													1			1		1				753	
)		1	1	1	1				1	1	5		1						1		4	1	3	2	5	
)																									27	
)																									13	

Fig. 8. Confusion matrix of the proposed method.

#### IV. CONCLUSION

The texture of a character changes from beginning to the end. Using LBP to take into consideration this information allows to tolerate the inter-class variability. And in order to recognize characters that differ only in certain details, we divided the image in four regions so as to differentiate those in interclass.

This technique is limited to treat different characters having similar shape but differ only in diacritical points. This motivated us to propose a method integrating the spatial aspect.

These two methods operate on the same object from different but complementary viewpoints. By combining them at the scores level using multiplication rule, the recognition rate is improved to 97.18% in TOP1 and 99.19% in TOP2.

#### REFERENCES

- [1] Y. Boulid, A. Souhar, and M. Y. Elkettani, "Handwritten character recognition based on the specificity and the singularity of the arabic language," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 4, no.4, pp. 45-53 Regular Issue, 2017.
- [2] J. Cai and Z.-Q. Liu, "Integration of structural and statistical information for unconstrained handwritten numeral recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 3, pp. 263–270, 1999.
- [3] L. Heutte, T. Paquet, J.-V. Moreau, Y. Lecourtier, and C. Olivier, "A structural/statistical feature based vector for handwritten character recognition," *Pattern recognition letters*, vol. 19, no. 7, pp. 629–641, 1998.
- [4] M. T. Parvez and S. A. Mahmoud, "Arabic handwriting recognition using structural and syntactic pattern attributes," *Pattern Recognition*, vol. 46, no. 1, pp. 141–154, 2013.
- [5] A. Amin, "Recognition of hand-printed characters based on structural description and inductive logic programming," *Pattern recognition letters*, vol. 24, no. 16, pp. 3187–3196, 2003.
- [6] A. Sahlol and C. Suen, "A novel method for the recognition of isolated handwritten arabic characters," *arXiv preprint arXiv:1402.6650*, 2014.
- [7] G. Abandah and N. Anssari, "Novel moment features extraction for recognizing handwritten arabic letters," *Journal of Computer Science*, vol. 5, no. 3, p. 226, 2009.
- [8] A. Asiri and M. S. Khorsheed, "Automatic processing of handwritten arabic forms using neural networks.," in *IEC (Prague)*, pp. 313–317, 2005.
- [9] J. Shanbehzadeh, H. Pezashki, and A. Sarrafzadeh, "Features extraction from farsi hand written letters," *Proceedings of Image and Vision Computing*, pp. 35–40, 2007.
- [10] F. Bouchareb, R. Hamdi, and M. Bedda, "Handwritten arabic character recognition based on svm classifier," in *Information and Communication Technologies: From Theory to Applications, 2008. ICTTA 2008. 3rd International Conference on*, pp. 1–4, IEEE, 2008.
- [11] M. Toriki, M. E. Hussein, A. Elsallamy, M. Fayyaz, and S. Yaser, "Window-based descriptors for arabic handwritten alphabet recognition: A comparative study on a novel dataset," *arXiv preprint arXiv:1411.3519*, 2014.
- [12] M. Rajabi, N. Nematbakhsh, and S. A. Monadjemi, "A new decision tree for recognition of persian handwritten characters," *International Journal of Computer Applications*, vol. 44, no. 6, pp. 52–58, 2012.
- [13] A. Alaci, P. Nagabhushan, and U. Pal, "A new two-stage scheme for the recognition of persian handwritten characters," in *Frontiers in Handwriting Recognition (ICFHR), 2010 International Conference on*, pp. 130–135, IEEE, 2010.
- [14] A. Alaci, U. Pal, and P. Nagabhushan, "A comparative study of persian/arabic handwritten character recognition," in *Frontiers in Handwriting Recognition (ICFHR), 2012 International Conference on*, pp. 123–128, IEEE, 2012.
- [15] S. Mozaffari and H. Soltanizadeh, "Icdar 2009 handwritten farsi/arabic character recognition competition," in *Document Analysis and Recognition, 2009. ICDAR'09. 10th International Conference on*, pp. 1413–1417, IEEE, 2009.
- [16] A. Lawgali, A. Bouridane, M. Angelova, and Z. Ghassemlooy, "Handwritten arabic character recognition: Which feature extraction method?," *International Journal of Advanced Science and Technology*, vol. 34, pp. 1–8, 2011.
- [17] Hammad, N. H., & Musa, M. E. (2016). The Impact of Dots Representation in Recognition of Isolated Arabic Characters. *International Journal of Information Engineering and Electronic Business*, vol. 8, no 6, p. 37.
- [18] Ewees, A. A., Sahlol, A. T., & Amasha, M. A (2017). A Bio-inspired Moth-Flame Optimization Algorithm for Arabic Handwritten Letter Recognition. *International Conference on Control, Artificial Intelligence, Robotics and Optimization (ICCAIRO 2017)*.
- [19] S. Mozaffari, K. Faez, F. Faradji, M. Ziaratban, and S. M. Golzan, "A comprehensive isolated farsi/arabic character database for handwritten ocr research," in *Tenth International Workshop on Frontiers in Handwriting Recognition*, Suvisoft, 2006.
- [20] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern recognition*, vol. 29, no. 1, pp. 51–59, 1996.
- [21] S. Tulyakov, S. Jaeger, V. Govindaraju, and D. Doermann, "Review of classifier combination methods," in *Machine Learning in Document Analysis and Recognition*, pp. 361–386, Springer, 2008.
- [22] M. Askari, M. Asadi, A. Asilian Bidgoli, and H. Ebrahimpour, "Isolated persian/arabic handwriting characters: Derivative projection profile features, implemented on GPUs," *Journal of AI and Data Mining*, vol. 4, no. 1, pp. 9–17, 2016.



Youssef Boulid

artificial intelligence.



Abdelghani Souhar

A. Souhar received M.S. degree in applied Mathematics in 1992, PhD degree in computer science in 1997 from University Mohammed 5 in Rabat - Morocco. Now he works as a Professor at university Ibn Tofail in Kenitra – Morocco. His research interests include CAE, CAD and Artificial Intelligence.



Mly Moustafa Ouagague

Mly.M Ouagaguereceived his M.S. degree in Decision Support Systems in 2001 from Blaise Pascal University. Now he works as a researcher at Maxware Technology Kénitra - Morocco. His research interests include artificial intelligence, complex system, image processing, serious games.