A Robust Statistical Set of Features for Amazigh Handwritten Characters[1]

N. Aharrane*, A. Dahmouni, K. El Moutaouakil, and K. Satori

Computer sciences, Imaging and Numerical Analysis Laboratory (LIIAN) USMBA University, Fez, Morocco
*e-mail: *aharranenabil@gmail.com*

Abstract—The main problem in the handwritten character recognition systems (HCR) is to describe each character by a set of features that can distinguish it from the other characters. Thus, in this paper, we propose a robust set of features extracted from isolated Amazigh characters based on decomposing the character image into zones and calculate the density and the total length of the histogram projection in each zone. In the experimental evaluation, we test the proposed set of features, to show its performance, with different classification algorithms on a large database of handwritten Amazigh characters. The obtained results give recognition rates that reach 99.03% which we presume good and satisfactory compared to other approaches and show that our proposed set of features is useful to describe the Amazigh characters.

Keywords: features extraction, pattern recognition, machine learning, HCR, amazigh character recognition, Tifinagh

DOI: 10.1134/S1054661817010011

1. INTRODUCTION

The introduction of new technologies in communication and documents archiving has not prevented the handwritten from continuing also its persistence; this required an automatic processing. To this end, the handwritten characters recognition (HCR) has received much attention in academic areas. In recent years, the HCR remains one of the most popular research subjects due to its diverse applications such as address classification systems, processing of bank check, indexing of archives, documents analysis, etc. Therefore, much work has been achieved for many languages, some of the excellent and recent surveys can be found in [1–4].

Recently, researchers have begun to give attention to the Amazigh language. Section 2 reports the related works to the Amazigh HCR.

The Amazigh language has existed since the earliest antiquity. It has an original writing system, called Tifinagh, used and preserved to this day. In recent decades, all Amazigh groups have reclaimed this ancestral writing. Currently, the Amazigh language is spoken by about 30 million speakers in North Africa (from the oasis of Siwa in Egypt to Morocco passing through Libya, Tunisia, Algeria, Niger, Mali, Burkina Faso and Mauritania). In Morocco, where nearly 50% of people are amazigh, the Amazigh language is divided into three regional varieties with Tarifit in the North, Tamazight in Central Morocco and South-East and Tachelhit in South-West [5].

The official introduction of the Amazigh language's teaching in the Moroccan educational system in 2003 involves the selection of a standard common language to teach. This task was accomplished by "Royal Institute of the Amazigh Culture" (IRCAM) created in July 2001 [6]. Nowadays, the Tifinagh-IRCAM alphabet is based on 33 characters (Fig. 1). In the amazigh HCR field, one works only on 31 characters because ⴳⵯ and ⴽⵯ do not have their corresponding Unicode.

The most important phase in the HCR system presented in Fig. 2 is the features extraction which consists in finding a set of features that can provide a description to each character to distinguish it from the others. Therefore, in this paper, we propose for each character, a new set of features that rests on decomposing the character image to different overlapped

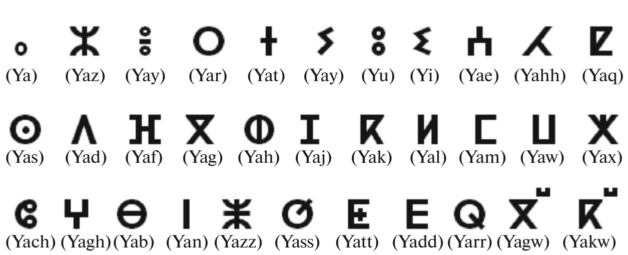

Fig. 1. Tifinagh characters adopted by the IRCAM.

[1] The article is published in the original.

Received March 2, 2016

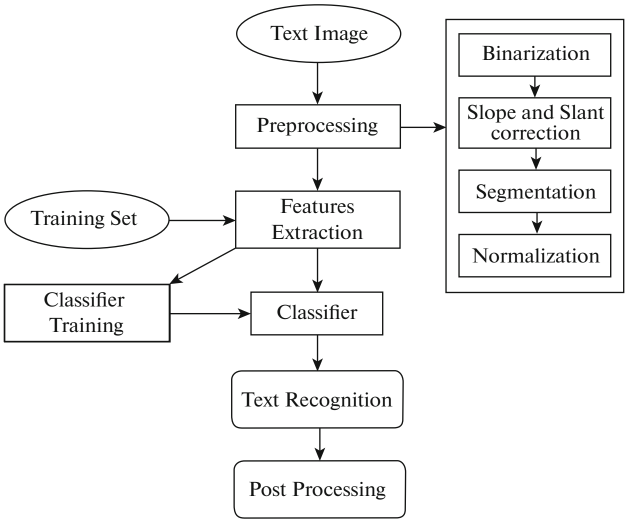

Fig. 2. Handwritten Characters Recognition System.

zones, and then we extract a vector of 79 components by calculating, in each zone, the density and the total length of the histogram projection. A detailed description of the extraction method of the set of features is given in Section 4. In order to test and prove the performance and robustness of the proposed set of features, we use different supervised classification methods for the recognition phase.

The rest of this paper is organized as follows: In Section 2, we present the related works to the Amazigh HCR systems. Section 3 delineates all Necessary preprocessing steps. The set of features proposed is presented in Section 4. Section 5 exposes the supervised classification methods, used in this work, to test the performance of our proposed set of features. In Section 6, we present the obtained experimental results. Section 7 detailed the post processing. Finally, we conclude the paper with Section 8.

2. RELATED WORKS

The Amazigh language has not been studied sufficiently. Among the researches in the literature, few works have been devoted to the HCR systems. Oulamara et al. [7] introduced the problem, in 1988, to the academic areas for both printed and handwritten characters; the authors used the Hough transform to describe characters and the absolute difference function to recognize them. 9 years later, Djematene and al [8] used a geometrical method to extract characters features and a metric distance in the recognition phase. The researches have been times till 2010 where the researchers began to give interest to the Amazigh language after its normalization and its introduction in the Moroccan educational system. Amrouch and al [9, 10] proposed two contributions basing on discrete and continuous HMM. In [11], Saady and al proposed a statistical approach for the features extraction and a neural network for the recognition. Abaynarh and al [12] used the Legendre moments with different classifiers and proposed an improvement [13] based on maximum entropy principle to optimize the features set extracted. Some structural approaches are also employed to describe characters such as Speeded Up Robust Features (SURF) and GIST descriptors [14, 15].

3. PREPROCESSING

In this section, we describe the preprocessing operations used to prepare the image to the next phases.

Binarization

The output of this operation is a binary image where the black pixels represent the text and the white ones indicate the background. In this regard, several algorithms have been proposed in [16]. The Otsu's method [17] still one of the most used binarization methods in character recognition systems.

Slope and Slant Correction

The correction of the line slope consists of searching a rotation transform making the words baseline parallel to the horizontal. Whereas the slant correction searches a vertical rectification of characters by making vertical the strokes which are supposed to be vertical. Several methods are proposed in the literature, Virajitha and al [18] gave a good description to the related works and proposed simple and effective algorithms for the slope and slant corrections.

Segmentation

The characters segmentation is one of the most important steps in a Handwritten Character Recognition system. The objective is to decompose the text image into a sequence of sub-images whereby each sub-image should contain a single character. For this, a segmentation of lines is performed and then each line is segmented into characters. A survey of methods and strategies in character segmentation is presented in [19]. The horizontal projection histogram is often used in order to segment the text image into lines. This method can distinguish between high-density areas characterizing the lines and low-density areas indicating the space between the lines. As the Amazigh writing, handwritten or printed, is never cursive, the character extraction from each line becomes easy by using the vertical projection histogram. Characters correspond to the areas of high density in the histogram.

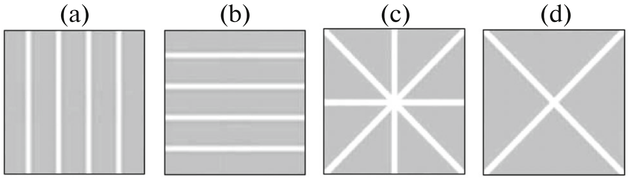

Fig. 3. First decomposition of character image. (a) Decomposition to 5 vertical equal zones; (b) Decomposition to 5 horizontal equal zones; (c) Decomposition to 8 octants; (d) Decomposition to 4 quadrants.

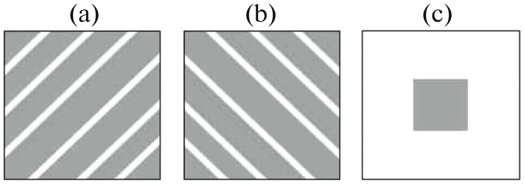

Fig. 4. Second decomposition of character image. (a) Left diagonal decomposition; (b) Right diagonal decomposition; (c) 10 × 10 middle zone.

Normalization

The segmentation process produces images of the isolated characters in different sizes. To solve this problem, a normalization step is needed. This latter consists on resize all the images of characters to a common size. In this work, due to its zooming quality, we use a spline-based algorithm [20] to resize all characters in a size of 30 × 30. This method outperforms the standard interpolation techniques and works for arbitrary scaling factors (image magnification or reduction). This spline-based algorithm achieves a reduction of artifacts such as aliasing and blocking and reaches a significant improvement of the signal-to-noise ratio.

4. FEATURES EXTRACTION

The features extraction step is the most important operation for an OCR system. Its aim is the selection of the most relevant information identifying each character to form a set of features describing this character.

In the literature, many features extraction methods have been applied to OCR systems. Arica.N and al [21] categorize them according to their type as follows:

—Global Transformation and Series Expansion;

—Statistical features;

—Geometrical and topological features.

In this paper, we propose a statistical set of features. This latter rests on decomposing the isolated character image into several overlapped zones according to different directions, then the density of black pixels and the total length of the histogram projection are calculated in each zone. The resulting set of features contains 79 components. In this section, we present the different stages to construct the proposed set of features.

Density Features

To create the first features subset, we carry out different decompositions of the character image and the density of foreground pixels is calculated in each zone to obtain 37 features. We obtained the density of each zone by dividing the number of black pixels by the total number of pixels in this zone. The density features describe the spatial dispersion of characters.

The first decomposition, as shown in Fig. 3, consists of dividing the character image vertically (Fig. 3a) and horizontally (Fig. 3b) to five equal zones, then to 8 octants (Fig. 3c) and in last to 4 quadrants (Fig. 3d).

The second decomposition is obtained by dividing the character image to 7 diagonal zones in both left and right directions (Figs. 4a, 4b), then considering only the middle zone whose size is 10 × 10. Diagonal features increase the recognition accuracy and reduce the misclassification. The middle zone is added to distinguish between some resembling Amazigh characters.

Following these decompositions, we obtain 37 zones and we calculate the density in each zone.

Total Length of the Histogram Projection Features

The second features subset is obtained from the first decomposition of the character image (Fig. 3).

In each zone, we calculate the total length of the histogram projection following the direction of projection (Fig. 5). This consists of summing the width of the histogram bins as shown in Fig. 6.

Each calculated value must be normalized by dividing it on the maximum possible length of the corresponding side.

5. CLASSIFICATION METHODS

After the features extraction, we proceed to the recognition phase where we use the most known algorithms of the supervised classification to test the robustness of our proposed set of features. The supervised classification algorithms aims to create a system that can predict the correct class of the domain instances. It consists of building a procedure that will be applied to a sequence of instances, where each new instance must be assigned to one of the predefined classes on the basis of the observed attributes or features. This kind of algorithms can be based on probabilistic models (Bayesian classifier); on notions of

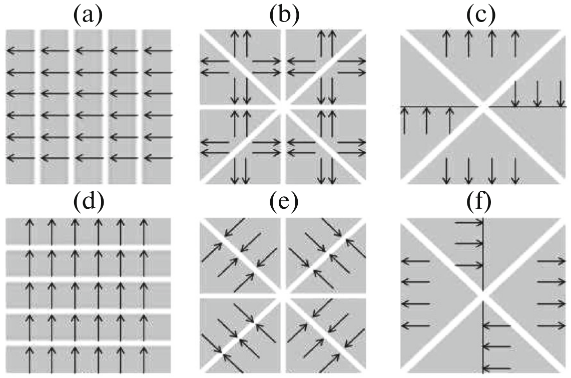

Fig. 5. Decomposition for total length of the histogram projection features. (a) 5 vertical zones; (b) 5 horizontal zones; (c) Horizontal and vertical features for each octant; (d) Diagonal features for each octant; (e) Vertical features for each quadrant; (f) Horizontal features for each quadrant.

proximity or similarity (nearest neighbors) or on research in areas of hypotheses (decision trees, neural networks, rule-based algorithms, Support Vector Machine...).

In this paper, we use the four most used methods in the context of supervised classification which are the multilayer perceptron, the Bayesian network, the k-nearest neighbors and support vector machine (SVM).

Multilayer Perceptron

The multilayer perceptron is one of the most popular neural networks used in machine learning. It is an oriented network, with supervised learning, organized in layers where the information is propagated in one direction, from the input layer to the output layer.

The input layer, the first layer, is always a virtual layer associated to the system inputs. It contains no neuron. The following layers are hidden layers. Neurons of the output layer, i.e., last layer, always correspond to the system outputs. In general, a multilayer perceptron can have any number of hidden layers and any number of neurons per hidden layer, but it should be noted that using one hidden layer is sufficient to solve a non-linear complex problem and the choice of the number of hidden layer is still a challenging issue. The neurons are connected together by weighted connections. The weights of these connections govern the operation of the network and program an application from input space to the output space through a non-linear transformation. The creation of a multilayer perceptron to solve a given problem requires the inference of the best possible application as defined by a set of training data consisting of pairs of input vectors and desired outputs. This inference can be realized, among others, by the algorithm called back-propagation.

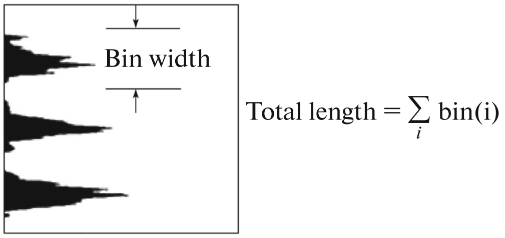

Fig. 6. Total Length of the histogram projection.

Bayesian Network

Bayesian networks rest on a formalism based on the theories of probabilities and graphs. It represents a set of random variables and their conditional dependencies. Introduced by Pearl in 1985, $B = (G, \theta)$ is a Bayesian network if $G = (X, E)$ is a directed acyclic graph (DAG) where the vertices represent a set of random variables $X = \{X_1, \ldots, X_n\}$ and if $i = [P(X_i/Pa(X_i))]$ is the conditional probabilities matrix of node i knowing the state of its parents.

A Bayesian network B represents a probability on X which admits the following joint distribution:

$$P(X) = P(X_1, X_2, \ldots, X_n) = \pi P(X_i/Pa(X_i)).$$

This decomposition of the joint distribution allows us to have powerful inference algorithms that make the Bayesian networks very useful tools for modeling and reasoning when the situations are uncertain or the data are incomplete. They are so useful also for classification problems where the interactions between the different criteria can be modeled by conditional probability relations which introduce the problem of choosing the network structure.

K-nearest Neighbors

This algorithm, unlike other algorithms, does not seek a function of classification to classify a new instance but it compares it to a database of pre-classified data examples. This comparison is based on a distance function that determines how close a new input instance is to each training set instance, and then the algorithm predicts the output class using the class of the nearest instance or instances. Many distance functions are proposed in the literature. The k-nearest neighbors showed its power in a large number of domains.

Support Vector Machine

The support vector machine (SVM) is a universal constructive learning procedure based on the statistical learning theory. Originally it was worked out for the binary classification. Thus, there are two approaches to solve a M-classes classification problem. In the first one, we should construct and combine a set of M binary classifiers $f_1, f_1, ..., f_M$, each trained to separate one class from the rest. The second approach considers all data in one optimization formulation.

This algorithm searches the optimal hyperplanes separating linearly the instances and can be extended to separate instances that are non-linearly separable by transforming the original data, using the kernel functions, in a space of a larger dimension where the data can be separated linearly. It uses a flexible representation of the class boundaries and maximizes the margin around the separating hyperplanes.

Multiple Classifiers Combination

The combination of multiple classifiers using the majority voting in the recognition phase is now considered as a very powerful tool to reach best performances. It had a great interest in the characters recognition research area. Its simple principle consists in giving to each instance the class that receives the largest number of votes by the classifiers.

The use of the output probabilities of each classifier allows to this technique to have different variants to predict the output class of the combined classifier. These output probabilities are combined by a combiner function such as the average, the product, or the maximum, and then the class with the highest probability after the combination is chosen as output. An overview of the use of Majority voting techniques for characters recognition is given in [22].

In order to improve our system performance, we combine all of these classifiers using some variants of majority voting systems by varying the combiner function.

6. EXPERIMENTATIONS AND RESULTS

Database

In order to evaluate the performance of the proposed set of features, the AMHCD database [23] is used as a source of training and test. This database consists of 25740 isolated Amazigh handwritten characters produced by 60 writers who wrote 13 samples of each of the 33 characters.

The writers were selected from various age, gender, and educational background groups. The samples were gathered by asking the writers to write on a form of 13 examples for each Amazigh character. An example of a filled form used in the data collection is given in Fig. 7.

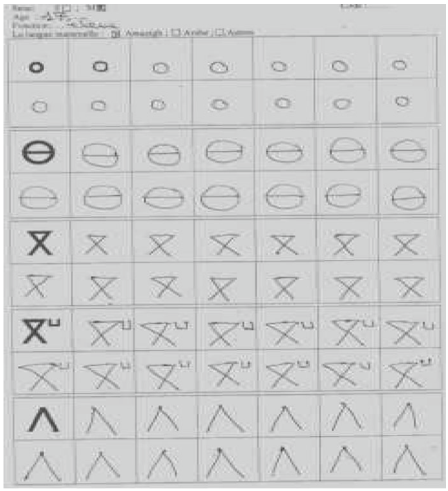

Fig. 7. Example of filled form.

As mentioned in section 2, researchers work only on 31 characters excluding characters ⴳⵯ and ⴽⵯ. So, experimentations were carried out on 24180 characters where 70% (16926) were for the training and the rest 30% (7254) for the test. Noted that the training/test set is composed of 546/234 characters of each class.

Classifiers Evaluation

A way for evaluating the performance of a classification model is the contingency matrix, called also confusion matrix, which records the observed and the predicted classes of data. Table 1 reports the contingency matrix for a binary classification and the commonly-accepted performance evaluation measures [24].

Table 1. The contingency matrix for the binary classification and some performance measures

		Observed Classes		
		Positive	Negative	Measures
Predicted Classes	Positive	True Positive (TP)	False Positive (FP)	Positive Predictive Value (PPV) or Precision
	Negative	False Negative (FN)	True Negative (TN)	Negative Predictive Value (NPV)
	Measures	Sensitivity or Recall	Specificity	Accuracy

Table 2. Comparison of different used classifiers

Classifiers	Accuracy	Recall	F-Score	Test Time (ms)
MLP	98.86	98.9	98.9	0.36
Bayesian Network	96.84	96.8	96.9	0.72
KNN	96.60	96.6	96.6	29.7
SVM	98.71	98.7	98.7	6.69
Classifiers Combination	**99.03**	**99.0**	**99.0**	**47.5**

Where:

$$Precision = PPV = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}, \quad (1)$$

$$Recall = Sensitivity = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \quad (2)$$

$$Specificity = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}}, \quad (3)$$

$$NPV = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FN}}, \quad (4)$$

$$Accuracy = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{TN} + \mathrm{FN}}. \quad (5)$$

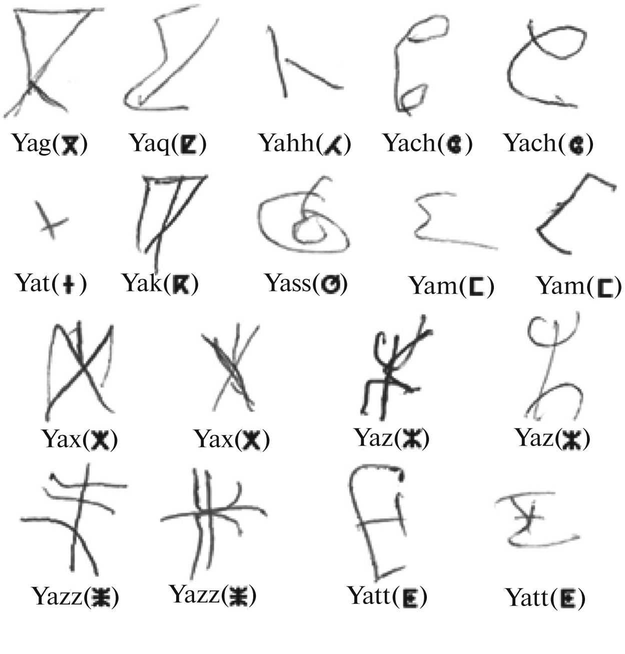

Fig. 8. Some of the characters badly written in AMHCD database.

Authors in [23] have noted also other useful measures, such as F-Score, ROC and AUC where:

$$F - Score = \frac{2 * \mathrm{Precision} * \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}}. \quad (6)$$

$$ROC = \frac{\mathrm{P(x|positive)}}{\mathrm{P(x|negative)}}. \quad (7)$$

$$AUC = \frac{\mathrm{Sencitivity} + \mathrm{Sensibility}}{2}. \quad (8)$$

The F-Score combines both precision and recall by a harmonic mean. The ROC (Receiver Operating Characteristic) And AUC (Area under ROC Curve) are used in applications that need a cost/benefits decision analysis.

Experimentations Setup

Several experimentations have been conducted for all algorithms with different configurations under a compatible HP ProBook, Intel (R) Core (TM) i5-2520M CPU 2.50 GHz, and 4 GB of RAM through Java language.

For the MLP, we carried out a series of runs by varying the number of neurons in the hidden layer. This number must be selected to be high enough to model the problem but not very high to avoid the overfitting. For the Bayesian approach, one of the main problems is to find the optimal network structure, thus, we tested different algorithms for learning the network structure (Hill Climber, Tabu Search, K2 and Tree Augmented Naive). We tested the KNN algorithms with different distance measures (Euclidean, Manhattan, Chebyshev and Minkowski). As for SVM, the experimentations were carried out with different kernel functions (Linear, Gaussian RBF and polynomial).

In order to improve the system performance, we combine all of these classifiers using the majority vote system with different combiner functions such as Average, Product, Max, and majority.

7. RESULTS AND DISCUSSIONS

Table 2 shows a global comparison of different used classifiers, combined with the proposed set of features, retaining only the best performance for each one. The detailed results for all classifiers and their different configurations are presented in Appendix A. Noted that the test time, the last column, represents the mean time taken (in milliseconds) by the classifier to recognize one character.

The misrecognition of the most characters is due to two factors. The first one is the structural similarity of some characters; The Appendix B shows the confusion matrix corresponding to the best performance recorded by the classifiers combination using the average of probabilities method. The second factor is bad writing of some characters in the database whose classification is difficult even for a human operator; Fig. 8 illustrates some of the badly written letters in the AMHCD database.

To showcase our method, in Table 3, we compare our obtained result with the other best approaches in the Amazigh HCR field. We point that the two approaches [10] and [11] used as source of training and testing sets the same used AMHCD database.

We signal that our obtained accuracy is the best one till now, in our knowledge, compared to other approaches in Amazigh HCR systems.

8. POST PROCESSING

A way to decrease the classification errors is to use the rejection option as a post-processing. Its aim is to reject the patterns that are the most likely to be misclassified. However, correct classifications can also be rejected.

Chow's rule [25] is commonly used to get a trade-off between errors and reject rates. According to Chow, a pattern is accepted and classified as class Q if the maximum of the a posteriori probabilities is higher than a given threshold value t ∈ [0, 1]:

$$\max_{k=1,\ldots,N} P(C_k|x) = P(C_i|x) \geq t.$$

Otherwise, the pattern is rejected.

The ROC curve is often used to determine the optimal threshold in classification problems. This curve represents the evolution of sensitivity depending on (1-specificity) when we vary the threshold t. The area under the ROC curve (AUC) gives a good estimate of the system's rejection capability, i.e., its ability to reject misclassified patterns without rejecting the well-classified characters. We can also use the trade-off Precision/Rejection rate (resp. Accuracy/Rejection rate) to present the results of rejection, in this case, it comes to minimize the rejection rate and maximize the precision (resp. the accuracy). Appendix C: Figure 9 illustrates the three curves for each of the classifiers using only their best configurations. The choice of the optimal threshold from the curves depends always to the type of the target application.

In this paper, we are interested in the system's accuracy, as we see in the Accuracy/Rejection rate curve, the accuracy decreases when using the rejection option for all the classifiers. This is due to the use of a single threshold for all the classes.

Femara et al. [26] have proved that the use of multiple reject thresholds related to the data classes in multi-classes classification problems can be very beneficial. Cecotti and al [27] choose the thresholds related to each class using some rejection methods aiming to maximize the accuracy, the precision, and the AUC. As mentioned previously, we are interested in the system's accuracy, i.e., maximize the rate of accepted and well-classified characters and the rate of rejected and misclassified ones, so we chose for each class C_i the threshold t that maximize the accuracy Acc_i.

Table 3. Comparison with other approaches

Approaches	Accuracy	Training set size	Test set size
Our Approach	**99.03**	**16926**	**7254**
Continuous HMM and Directional features [10]	97.89	16120	8060
Horizontal and vertical centerline of character [11]	96.32	18135	2015
Geometrical Method [8]	92.30	1000	700
Fuzzy KNN and Bi-gram language Model [14]	91.05	1940	Not specified

Table 4. Results for multi-thresholds for different classifiers

Classifier	Accuracy	Rejection Rate
MLP	98.91	0.17
SVM	98.92	0.35
Bayesian Network	97.04	1.80
KNN	96.71	0.37
Classifiers combination (Average of probabilities)	99.10	0.37

Table 4 reports the obtained results when using multi-thresholds for different classifiers. The results show that the accuracy of all the classifiers has been improved while keeping low rejection rates. This amounts to saying that the most of the rejected characters are those that are supposed to be misclassified.

These results prove also the beneficial use of multiple thresholds against one in a multi-classes classification problem.

CONCLUSION

We proposed a robust statistical set of features to describe the Amazigh characters. This one consists on decomposing the image character to different overlapped zones and then we calculate the density and the total length of the histogram projection of each zone. In result, we extracted a vector of 79 components to represent each character which serves as input for the different supervised classification algorithms used in this work.

According to the experimental analyzes, the proposed statistical set of features gives good results with very satisfactory recognition rates when combined with different classifiers. And we assert that the statistical features are useful features to describe Amazigh characters.

APPENDIX A

Table 5. Results for different Number of hidden nodes

Number of hidden nodes	Accuracy on test set (%)	Recall (%)	F-Measure (%)
70	98.41	98.4	98.4
75	98.32	98.3	98.3
80	98.69	98.7	98.7
85	**98.86**	**98.9**	**98.9**
90	98.75	98.7	98.7
95	98.68	98.7	98.7

Table 6. Results for different Algorithms for learning the network structure.

Structure of Bayesian Network	Accuracy on test set (%)	Recall (%)	F-Measure (%)
Tree Augmented Naive	96.41	97.2	97.2
Tabu search	**96.84**	**96.8**	**96.9**
K2 Algorithm	96.56	96.6	96.6
Hill Climber Algorithm	96.85	96.9	96.9

Table 7. Results for SVM with different kernel functions

Kernel Function	Accuracy on test set (%)	Recall (%)	F-Measure (%)
Linear	97.61	97.6	97.6
Gaussian RBF	98.56	98.6	98.6
Polynomial	**98.71**	**98.7**	**98.7**

Table 8. Results for different distance functions used with KNN

Distance Function	Accuracy on test set (%)	Recall (%)	F-Measure (%)
Euclidean Distance	96.18	96.2	96.2
Manhattan Distance	**96.60**	**96.6**	**96.6**
Chebyshev Distance	94.11	94.1	94.1
Minkowski Distance	96.18	96.2	96.2

Table 9. Results for different classifiers combination methods

Voting Method	Accuracy on test set (%)	Recall (%)	F-Measure (%)
Majority Vote	98.76	98.8	98.8
Maximum of probabilities	98.56	98.6	98.6
Product of probabilities	98.87	98.9	98.9
Average of probabilities	99.03	99.0	99.0

APPENDIX B

Table 10. Confusion matrix for the classifiers combination with the average of probabilities method

	ⴰ	ⵣ	ⴼ	ⵔ	ⵜ	ⵢ	ⵓ	ⵉ	ⵀ	ⵄ	ⵇ	ⴱ	ⴷ	ⵃ	ⴳ	ⵟ	ⵊ	ⴽ	ⵍ	ⵎ	ⵡ	ⵅ	ⵛ	ⵖ	ⵙ	ⵏ	ⵥ	ⵚ	ⴹ	ⴻ	ⵕ
ⴰ	234																														
ⵣ		230																				1					3				
ⴼ			232				2																								
ⵔ				228								5													1						
ⵜ					231																					3					
ⵢ						233									1																
ⵓ			1				233																								
ⵉ			1			1		230			2																				
ⵀ					1				232	1																					
ⵄ										233			1																		
ⵇ											234																				
ⴱ				4								228																			2
ⴷ													234																		
ⵃ														233					1												
ⴳ															234																
ⵟ																233												1			
ⵊ																	233			1											
ⴽ											1							232												1	
ⵍ																			234												
ⵎ																	1			233											
ⵡ																					234										
ⵅ		1			1					1					3							228									
ⵛ																							234								
ⵖ																			1					233							
ⵙ																									234						
ⵏ																										234					
ⵥ		12												1								1					220				
ⵚ				1																								233			
ⴹ																		1											228	5	
ⴻ																				1									2	231	
ⵕ												1													1		1				231

APPENDIX C

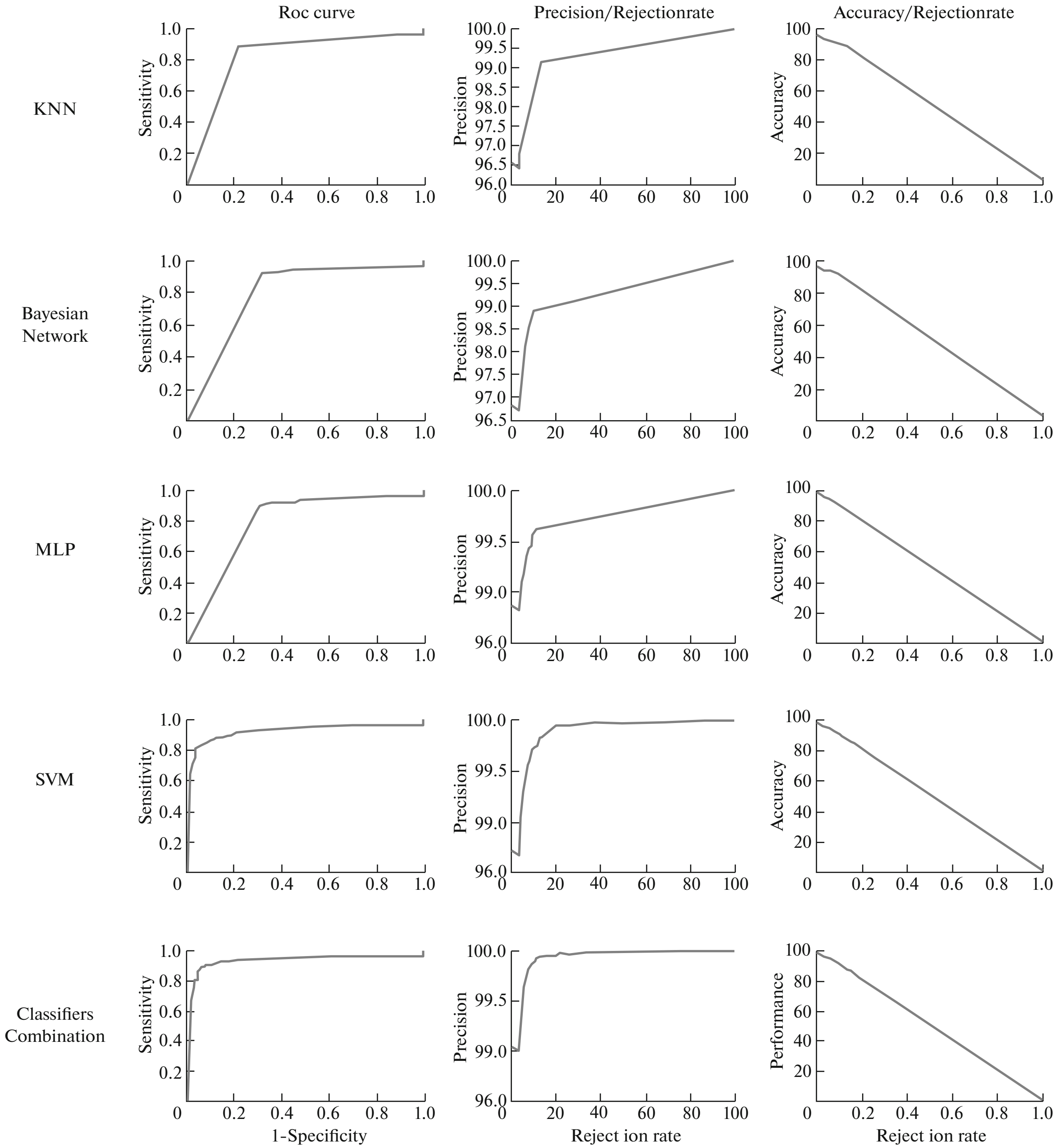

Fig. 9. ROC curve, Precision/Rejection Rate and Accuracy/Rejection Rate curves for different classifiers

REFERENCES

1. H. Mirvaziri, M. Masood Javidi, and N. Mansouri, “Handwriting recognition algorithm in different languages: survey,” Int. Lecture Notes Comput. Sci. **5857**, 487–497 (2009).

2. Y. Elarian, R. Abdel-Aal, I. Ahmad, M. Parvez, and A. Zidouri, “Handwriting synthesis: classifications and techniques,” Int. J. Document Anal. Recogn. **17** (4), 455–469 (2014).

3. N. Tagougui, M. Kherallah, and M. Alimi, “Online arabic handwriting recognition: a survey,” Int. J. Document Anal. Recogn. **16** (3), 209–226 (2013).

4. S. Bag and G. Harit, “A survey on optical character recognition for Bangla and Devanagari script,” Sadhana **38** (1), 133–168 (2013).

5. M. Ameur, A. Bouhjar, F. Boukhris, A. Boukouss, A. Boumalk, M. Elmedlaoui, E. Iazzi, and H. Souifi, “Initiation à la langue amazighe,” Publications de l’Institut Royal de la Culture Amazighe, Manuels, No. 1, 9 (2004).
6. Presentation of The Royal Institute of the Amazigh Culture (2014). http://www.ircam.ma/doc/divers/presentation_of_ircam.pdf.
7. A. Oulamara and J. Duvernoy, “An application of the hough transform to automatic recognition of Berber characters,” Signal Processing **14** (1), 79–90 (1988).
8. A. Djematene, B.Taconet, and A. Zahour, “A geometrical method for printing and handwritten Berber characters recognition,” in *Proc. 4th Int. Conf. on Document Analysis and Recognition* (Ulm, 1997), Vol. 2, pp. 564–567.
9. M. Amrouch, A. Rachidi, M. El Yassa, and D. Mammass, “Handwritten amazigh character recognition based on hidden Markov models,” J. Graph., Vision Image Processing **10** (5), 11–18 (2010).
10. M. Amrouch, Y. Es-saady, A. Rachidi, M. El Yassa, and D. Mammass, “Handwritten amazigh character recognition system based on continuous HMMs and directional features,” Int. J. Modern Eng. Res. **2** (2), 436–441 (2012).
11. Y. Es Saady, A. Rachidi, M. El Yassa, and D. Mammass, “Amazigh handwritten character recognition based on horizontal and vertical centerline of character,” Int. J. Adv. Sci. Technol. **33**, 33–50 (2011).
12. M. Abaynarh, H. Elfadili, and L. Zenkouar, “Recognition of Tifinaghe handwritten characters using moments for feature extraction,” in *Proc. 4th atelier international sur l’amazighe et les TICs : Les resources langagieres : construction et exploitation* (2011), pp. 345–356.
13. M. Abaynarh, H. Elfadili, and L. Zenkouar, “Enhanced feature extraction of handwritten characters and recognition using artificial neural networks,” J. Theor. Appl. Inf. Technol. **72** (3), 355–365 (2015).
14. S. Gounane, M. Fakir, and B. Bouikhalene, “Handwritten Tifinagh text recognition using fuzzy K-NN and Bi-gram language model,” in *Selected Papers from 3rd Int. Symp. on Automatic Amazigh Processing (SITACAM’13)* (Beni Mellal, 2013), Special Issue, pp. 29–32.
15. H. Moudni, M. Er-rouidi, M. Oujaoura, and O. Bencharef, “Recognition of amazigh characters using SURF & GIST descriptors,” Int. J. Adv. Comput. Sci. Appl. Special Issue on Selected Papers from 3rd Int. Symp. on Automatic Amazigh Processing (2013), pp. 41–44.
16. N. Chaki, S. Shaikh, and K. Saeed, “A comprehensive survey on image binarization techniques,” Studies Comput. Intellig. **560**, 5–15 (2014).
17. N. Otsu, “A threshold selection method from gray-level histograms,” IEEE Trans. Syst. Man Cybern. **9**, 62–66 (1979).
18. B. Jain and M. Borah, “Simple and effective techniques for skew correction, slant correction and core-region detection for cursive word recognition,” Adv. Intellig. Soft Comput. **132**, 353–361 (2012).
19. T. Saba, A. Rehman, and M. Elarbi-Boudihir, “Methods and strategies on off-line cursive touched characters segmentation: a directional review,” Artif. Intellig. Rev. **42** (4), 1047–1066 (2014).
20. A. Muñoz Barrutia, T. Blu, and M. Unser, “Least-squares image resizing using finite differences,” IEEE Trans. Image Processing **10** (9), 1365–1378 (2001).
21. N. Arica and F. T. Yarman-Vural, “An overview of character recognition focused on off-line handwriting,” IEEE Trans. Syst. Man Cybern. C Appl. **31** (2), 216–232 (2001).
22. A. F. R. Rahman, H. Alam, and M. C. Fairhurst, “Multiple classifier combination for character recognition: revisiting the majority voting system and its variations,” Lecture Notes Comput. Sci. **2423**, 167–178 (2002).
23. Y. Es Saady, A. Rachidi, M. El Yassa, and D. Mammass, “AMHCD: a database for amazigh handwritten character recognition research,” Int. J. Comput. Appl. **27** (4), 44–48 (2011).
24. M. Sokolova, N. Japkowicz, and S. Szpakowicz, “Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation,” Lecture Notes Comput. Sci. **4304**, 1015–1021 (2006).
25. C. Chow, “On optimum recognition error and reject tradeoff,” IEEE Trans. Inf. Theory **16** (1), 41–46 (1970).
26. G. Fumera, F. Roli, and G. Giacinto, “Multiple reject thresholds for improving classification reliability,” Lecture Notes Comput. Sci. **1876**, 863–871 (2000).
27. H. Cecotti and S. Vajda, “Rejection schemes in multiclass classification – application to handwritten character recognition,” in *Proc. 12th Int. Conf. on Document Analysis and Recognition* (Washington, 2013), pp. 445–449.

Nabil AHARRANE received his MSc degree in computer science and imaging in 2008 from USMBA University in Morocco. Now he is a PhD student at the Department of computer science at the same University on the subject of Pattern recognition. His interests include OCR Systems problems, machine learning and features extraction from images.

Karim EL MOUTAOUAKIL received the PhD degree from the Faculty of sciences and technologies in Fez-Morocco in 2011. He is currently an Assistant professor of computer science at the National school of applied sciences in Al-Hoceima-Morocco. His research interests include Articiciel intelligence, machine larnining and pattern recognition.

Abdellatif DAHMOUNI received his MSc degree in computer science and imaging in 2013 from USMBA University in Morocco. Now he is a PhD student at the Department of computer science at the same University on the subject of Pattern recognition. His interests include face recognition problems, machine learning and Local Binary Pattern.

Khalid SATORI received the PhD degree from the National Institute for the Applied Sciences INSA at Lyon in 1993. He is currently a full professor of computer science at USMBA University in Morocco. His is the director of the LIIAN Laboratory. His research interests include real-time rendering, Image-based rendering, virtual reality, biomedical signal, camera self calibration and 3D reconstruction and pattern recognition.