

Research Article

Online Handwritten Arabic Scripts Recognition Using Stroke-Based Class Labeling Scheme

Rabiaa Zitouni^{1,*}, Hala Bezzine², Najet Arous¹

¹Laboratory LR-SITI ENIT University Tunis El Manar, B.P.37, Tunis, 1002, Tunisia

²Laboratory REGIM ENIS University sfax, B.P.1173, Sfax, 3038, Tunisia

ARTICLE INFO

Article History

Received 15 Jul 2020
 Accepted 15 Oct 2020

Keywords

Beta-elliptic model
 Fuzzy perceptual codes
 Strokes classification
 Spatial relations
 Online handwriting recognition
 Two-stage SVMs

ABSTRACT

With the increasing availability of pen-based user interfaces, we often come upon multiple data sets of online handwritten scripts such as letters, words, etc., that are collected based on a viable interface. In this paper, we set forward a new method for online handwritten Arabic scripts recognition. Departing from the assumption that handwritten scripts are encoded as a set of strokes, the proposed approach relies first upon classifying strokes contained on the script and then recognizes the whole script. For stroke classification, an support vector machine (SVM) is trained with stroke features vectors obtained from the Beta-elliptic model and fuzzy elementary perceptual codes to obtain class stroke probabilities. The output of this SVM is combined with spatial relation vectors feeding to a second SVM to provide scripts level recognition. The proposed model has been tested on MAYASTROUN dataset. In order to obtain additional insight into the efficiency of the proposed approach, we performed further experiments on ADAB data set. The experimental results highlight its relevance by comfortably outperforming state-of-art systems.

© 2021 The Authors. Published by Atlantis Press B.V.

This is an open access article distributed under the CC BY-NC 4.0 license (<http://creativecommons.org/licenses/by-nc/4.0/>).

1. INTRODUCTION

It is well established that handwriting recognition (HR) has been still a complicated and a challenging task for more than 30 years despite the rapid progress in terms of techniques used in the recognition [1,2]. Resting on input data, it is customary to distinguish between online and offline HR systems. The offline HR system obtains data in the form of image and applies image processing technique for its analysis [3,4]. On the contrary, online HR system obtains data as an ink composed of sequence of (x, y) coordinates [5,6]. For this reason, in the last decade, research on online HR has gained more interest owing to increasing pen-based user interfaces. Within this framework, several methods have been set forward for online HR in different languages such as Latin [7,8], Chinese-Japanese [9,10], Indian [11,12], Arabic scripts [13,14], etc. Compared to Indo-European scripts, Arabic ones are linguistically more different and richer. They allow only cursive style implying four different forms: isolated, initial, medial and final according to their appearance in the script. Diacritic signs such as duma (´), fatha (-) or chadda (ω) as well as dots have a heavy presence in the Arabic language. Moreover, most Arabic scripts look very similar except for some minor changes in the scripts dots. This raises the need to carry out more research in the field of design and development of Arabic HR system as presented in survey [2,15] as well as competition papers [16,17]. The main difficulty to obtain an efficient and scalable system for online HR resides in the variation

of handwritten scripts coming from different writers and writing instruments. To resolve this problem, researchers resort basically to the use of pattern recognition (PR) families, i.e., structural and statistical methods. On the one hand, statistical methods are entirely based on feature representation as a formalism of representation of script. These representations provide us with a flexible and controllable vector space allowing us to define basic operators [18]. On the other hand, structural methods provide descriptions of scripts as well as relationships that might exist within their strokes [19]. This leads to being invariant to scattering operators such as rotation, scale, reflection, occlusion, deformation and translation. Obviously, both approaches display certain advantages and the latest research attempts to take benefit from both. Combining them in HR raises the need to model components of scripts as well as their spatial relation so as to benefit from their main advantages: (i) Modeling component shape as well as relationships within their stroke rather than whole script reduces modeling complexity. (ii) Relationships are usually fairly robust to geometric variations and important for differentiating between scripts of similar shapes. To the best of our knowledge, despite its attractive merits, the fusion of both approaches has not yet been explored in online HR. From this perspective, we propose a new methodology for Arabic handwritten script recognition that employs two support vector machines (SVMs) classifiers. In fact, SVM seeks to resolve the main problems of the statistical learning theory, i.e., over fitting and control the classification complexity using structural risk minimization. On the one hand, the proposed methodology depends on following the stroke-based class labeling scheme. Generally, an

*Corresponding author. Email: zitounirabaa@yahoo.fr

input script is arranged based on a collection of strokes. The latter refers to a hand-drawn shape comprised between pen up and pen down. Therefore, this scheme of class labeling basic stroke enables recognition of those scripts. Thus, script recognition accuracy of the system increases. To shed light on the efficiency of this methodology, several experiments were carried out. The results reveal promising findings that might serve for future investigations. The detailed block diagram of our framework is demonstrated in Figure 1. In the proposed system, initially, for the given input handwritten script, the script was subdivided into its constituent basic strokes. Basic stroke wise features were then characterized from the Perceptual Theory for On-line Handwriting Segmentation (PerTOHS) theory [20]. Grounded on some theories of visual perception [21], scientists report that handwriting is as an ordered movement expressing forms and can be encoded by a set of primitives grouped together to get suitable visual representation. Departing from this hypothesis, Sourour [20] presented a new PerTOHS relying on identifying relevant perceptual codes which are concatenated to obtain a suitable shape of writing permitting its easy representation. The PerTOHS theory was performed according two stages: (i) first a segmentation stage using Beta-elliptic model (BEM) is constructed in order to collect a set of elliptic strokes and parameters and (ii) second, using parameter θ for each ellipsis by the horizontal axis, each elliptic stroke codifies with one of the four types of elementary perceptual codes (EPCs). Nevertheless, the obtained EPCs suffers from uncertainty and imprecision owing to different constraints such as hand disorder or false visual feedback of handwritten shape. Although there are many methods that can be used to solve the problems of uncertain data, fuzzy Logic still remains the best one. In fact, fuzzy logic stands for an important tool providing a consistent framework for information fusion as well as granting mathematical tools for reasoning and decision-making. Thus, a fuzzy logic detector is employed, giving fuzzy elementary perceptual codes (F-EPCs) with a predetermined decision to form an appropriate stroke bounded with pen up and pen down [22].

Over the last decade, PerTOHS theory was performed on many tasks like writer identification and verification [23,24], handwriting generation [25,26]. It achieved good results compared to state of art works. Recently, PerTOHS theory has also fulfilled significant progress in HR. Among them, we state the system of [27]. In this system, the authors start with encoding handwriting strokes

extracted from the Beta-elliptic step with EPCs. Thereafter, the basic operations of genetic algorithm are applied to obtain global perceptual codes (GPCs). Those CPGs are evaluated with decision tree classifier and achieved a recognition rate of 80% on a set of 120 examples of digits and Arabic letters collected on mobile application. In more recent systems, the same authors proposed a novel model based on F-EPCs and long short-term memory (LSTM) network for online HR. These are able to support both languages (Arabic and Latin) [28]. Experiments were performed on a 5 data sets and displayed very promising results. In 2019, Akouadi *et al.* combined BEM parameters with F-EPCs using convolution neural network (CNN) classifier. The test results on MAYASTROUN and ADAB data sets proved the effectiveness of this architecture [29]. The obtained results and findings encourage to design more online HR based on PerTOHS theory. Therefore, the obtained BEM parameters and F-EPCs were studied using SVM to specify the class of that particular basic stroke. To this extent, additional features on spatial relationships between classified strokes together with an SVM output generate the features of the entire script that would be processed using second SVM to recognize the script. In order to model the spatial relation between strokes, we suggested the strategy extended from our previous work to portray them. This strategy produces directional and topological relations thanks to the use of geometric features and fuzzy logic. Spatial relationships exhibited among handwriting strokes play a significant role in the interpretation of a handwritten script. They permit to describe the spatial arrangement of two strokes in relation to each other. At this stage of analysis, we would assert that exploring the fuzzy logic concept is meant to evaluate the possible interpretations and precisions through a fuzzy set. Fuzzy sets need to introduce a linguistic designation to each spatial relation. Furthermore, since fuzzy rule-based systems simulate human reasoning, it is natural to use them in handwriting spatial relation description. The rest of this paper is organized as follows: Section 2 introduces the suggested method for modeling and classification of strokes within online Arabic scripts. Section 3 exhibits the details for developing the recognition system for Arabic scripts. Section 4 highlights experimental results along with the experimental setup as well as discussions regarding the performance of the proposed approach as compared to the state-of-the-art ones. Section 5 wraps up the conclusion and provides new perspectives for future works.

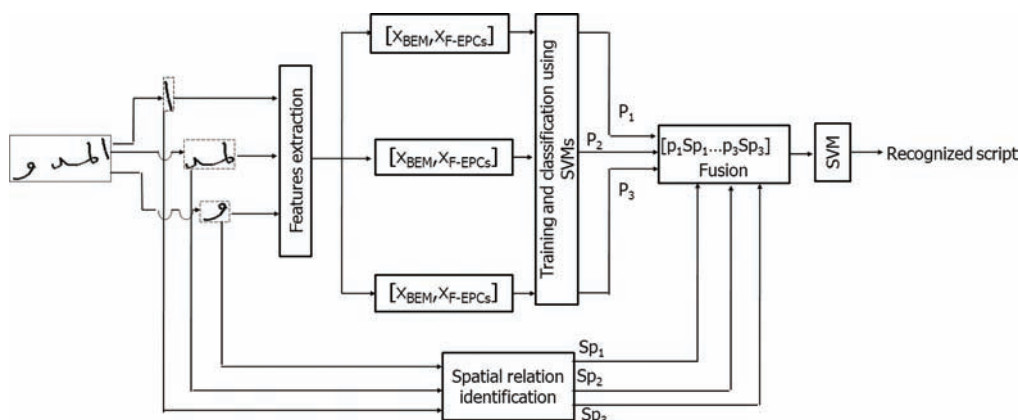


Figure 1 | Detailed architecture of the proposed recognition system.

2. HANDWRITTEN STROKES MODELING AND CLASSIFICATION

Online handwritten scripts consist of multiple sequential units called strokes and the relationships between them. A stroke forms the building block that occurs between pen ups and pen downs and is composed of an ordered sequence of points. Each point is formulated as (x, y, p) where (x, y) is the initial coordinates and, p indicates the pressure with value 0 or 1. The binary indicator p is determined by judging that the pen is persistently touching the pad (value 0) or will be lifted from the pad after the point (value 1). As illustrated in Figure 2, the lowest hierarchy is the handwritten strokes, and the script is formed using one or more strokes.

Character = Stroke. Stroke. $\{Stroke\}^+$

Stroke = point. point. $\{point\}^+$

where “+” denotes repetition and “.” indicates concatenation. In our approach, using MAYASTROUN and ADAB data sets, we construct a dictionary of forty seven strokes as illustrated in Figure 3.

Noting that, strokes numbered from S_1 to S_{28} are fundamental strokes, whereas strokes numbered from S_{29} to S_{47} result from frequent merging or warping of fundamental strokes that occur in handwriting.

2.1. Stroke Classification Using Static, Dynamic and Perceptual Features with a Gaussian Kernel

In order to classify strokes, a feature vector representation is needed. Thus, two types of features are distinguished: Beta-elliptic features and perceptual ones. The full obtained feature vector feeds an SVM classifier to obtain the class probability of stroke.

2.1.1. Beta-elliptic features

Beta-elliptic model is designed essentially for understanding and generating simple and complex handwriting movements. Recently, it has been used for many online HR tasks and permits to obtain efficient results. In those approaches, Beta-elliptic parameters are extracted and transformed into feature representation for input script. From this perspective, we tend to explore Beta-elliptic feature representation for our stroke classification. Beta-elliptic approach start by segmenting each input signal into a smaller unit called Beta strokes which are the output of a superimposition of time-overlapped of velocity [30,31]. The beta function defining a velocity

profile is expressed as

$$\beta(t, p, q, t_0, t_1) = \left[\frac{(t - t_0)}{(t_c - t_0)} \right]^p \left[\frac{(t_1 - t)}{(t_1 - t_c)} \right]^q \text{ if } t \in [t_0, t_1] \quad (1)$$

$$= 0 \text{ if not } p, q, t_0 < t_1 \in \mathbb{R}$$

$$t_c = \frac{p * t_1 + q * t_0}{p + q} \quad (2)$$

where t_0, t_1 are the starting and the ending time respectively, q and p are intermediate parameters values and t_c is the time when the curvilinear velocity attains its maximum value. Thus, in the static domain, each beta stroke is modeled by an elliptic arc scriptized with five parameters (a, b, x_0, y_0, θ) where a and b represent respectively the dimensions of the large and the small axes of the elliptic arc. x_0 and y_0 are the coordinates of the elliptic center with respect to the orthogonal reference as depicted in Figure 4.

The deviation angle θ formed by the ellipsis and the horizontal axe, as presented in [3].

$$\theta = \arctan \left(\frac{(y_1 - y_0)}{(x_1 - x_0)} \right) \quad (3)$$

Basically, using BEM each script is modeled in the static and dynamic domain by a series of elliptic arcs and Beta profiles respectively. The latter denote the set of strokes constituting the script. As a matter of fact, for each stroke we have ten parameters $(t_c, p, q, t_0, t_1, a, b, x_0, y_0, \theta)$.

2.1.2. The EPCs extractor

By means of the BEM for the segmentation of handwriting movements, each handwritten script is modeled by a series of elliptic arcs in the trajectory domain.

Each elliptic arc will be coded by a set of five parameters (respectively: a, b, x_0, y_0, θ) and for each elliptic stroke, an EPC is assigned. However, it is obvious that when an elliptic stroke has such a representation, it exhibits certain problems such as vagueness and uncertainty. Departing from this assumption, we have applied the fuzzy logic theory to attribute a membership degree for each EPC (Figure 5). In conclusion, each stroke is defined in terms of the combination of multiple F-EPCs [20].

Stroke = $\{F - EPC_{i1}, F - EPC_{i2}, \dots, F - EPC_{in}\}$ where $1 \leq n \leq 4, i$ corresponds to the total number of strokes in handwriting script.

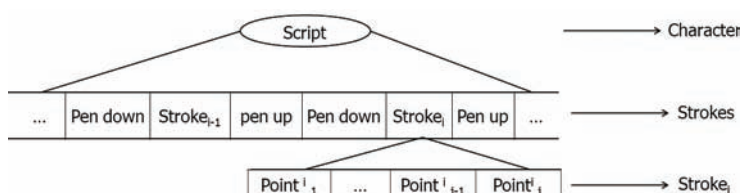


Figure 2 | Online character formation hierarchy.

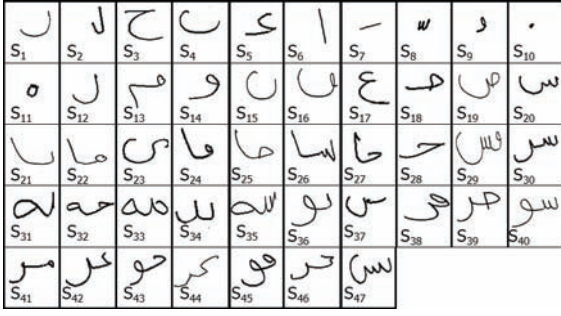


Figure 3 | Dictionary of forty seven strokes from MAYASTROUN and ADAB data sets.

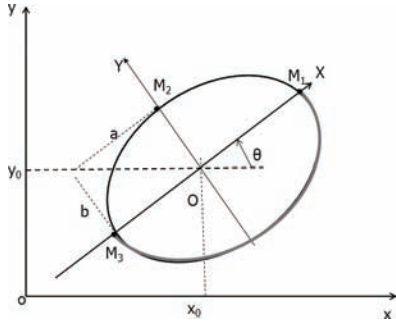


Figure 4 | Different elliptic parameters.

2.2. Stroke Classification

Ultimately, the plan was to classify an input stroke for a script based on the feature vector computed as indicated in the previous section. More formally, each stroke S_i can be described by a unique expression denoted in term of as

$$x_i = [x_{BEM}, x_{F-EPC}] \quad (4)$$

In this expression, the feature vector x_{BEM} is provided as $x_{BEM} = [x_{BEM1} x_{BEM2} \dots x_{BEMd}]$ with d is equal to the ten features extracted from the Beta- elliptic feature; five of them refer to dynamic parameters and five refer to static parameters. Similarity, x_{F-EPC} is represented as a vector $x_{F-EPC} = [x_{F-EPC1} \dots x_{F-EPCz}]$ with z being equal to four value associated to F-EPCs. Indeed, the classification step now is established using SVMs which are considered as the best binary classifiers to explore the optimal separating hyperplane between two classes [32]. Specifically, let $(k_n, c_n) \in R^M \times (+1, -1)$ be a set of training samples so that M corresponds to data dimension $n = 1 \dots N_c$ and N_c stand for the number of samples per class. SVM training chooses the function f , which maximizes the margin between the two classes by minimizing an upper bound on the generalization error [33]. Therefore, data are classified in accordance with

$$f(k) = \text{sign} \left(\sum_{i=1}^{sv} \beta_i q_i K(k_i, k) + b \right) \quad (5)$$

where b is the bias, q_i is the class label, β_i is the Lagrange multipliers and SV is the number of support vectors. The performance of SVM depends on the used kernel corresponding to: linear, polynomial, RBF and sigmoid kernels. We opted to use sigmoid kernel since it proved to be the most effective and suitable for many applications,

especially for HR. This kernel is expressed as follows:

$$k(k_i, k_n) = \exp \left(\frac{-1}{2\sigma^2} \|k_i, k\|^2 \right) \quad (6)$$

σ is the user-defined parameter but SVM has another tuning parameter which is c needed in case the training set is an inseparable problem. The value of both parameters will be defined in experimentation. It's worth noting that SVMs were originally designed to solve binary classification problems.

In order to address the multi-classification problem, binary SVM classifiers opt to use a one-against-one technique that constructs $N(N-1)/2$ classifiers, where N is the number of classes. When an input stroke S is given, the output values in Equation (5) are computed from the N SVMs and the stroke is classified as one of the N classes providing the maximum output value. Overall, an input stroke S_i is classified as one of N classes. It is first designed as a feature vector x_i in (4). Then, the feature vector is applied to N SVMs and the output of this stroke classification is indicated as the following:

$$p = [p_1, p_2, \dots, p_N] \quad (7)$$

$$p_i = \frac{\exp(\gamma_i)}{\sum_{i=1}^N \exp(\gamma_i)} \quad (8)$$

where γ_i is the output of the i -th SVM in Equation (5) and p_i denotes the probability that the feature vector x_i belongs to the i -th class.

3. STROKE CLASSIFICATION EXTENSION TO SVM-BASED SCRIPT RECOGNITION

In this section, the proposed approach for recognizing handwritten script is handled. The approach begins with grouping an input script to the number of strokes it involves. Let's consider for example, the scripts illustrated in Figure 6 composed of one, two, three and four strokes respectively, where each stroke is denoted as a dashed-colored rectangle.

Subsequently, the system classifies the handwritten scripts inside each of the groups through the use of appropriate method. While in previous methods only the classification stroke result is considered in script recognition steps in our method the spatial relation among the classified strokes is additionally considered. This helps to capture spatial structure, order and script shape.

3.1. Spatial Relationships Description via Fuzzy Rule-Based Approach

Spatial relationship identification is a very basic problem, which makes it widely applicable to online [34] as well as other offline tasks [35]. Within handwriting processing domain, spatial relations between handwritten strokes can play an intrinsic role to ease handwriting description and understanding. For example, same strokes can appear differently in different scripts and can be placed in different positions, which may change the scripts meaning. Figure 7 highlights an example of two Arabic scripts that share the same number of strokes, diacritic (number and type) and shape.

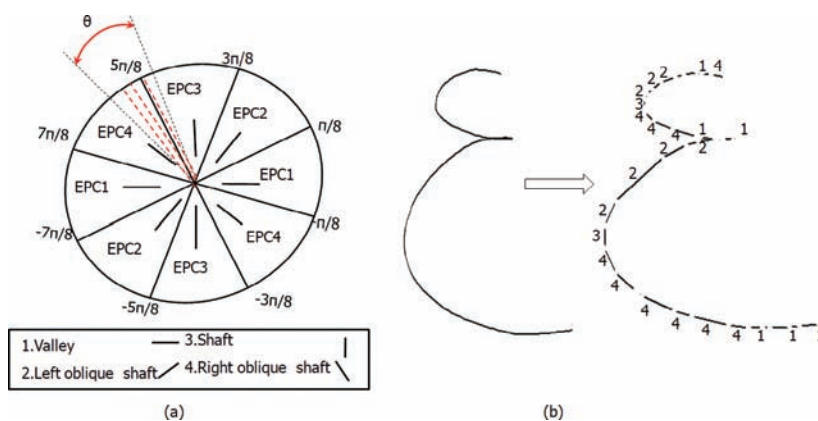


Figure 5 | (a) Different elementary perceptual codes (EPCs) on the trigonometric circle and (b) stroke and produced sequence of EPCs.

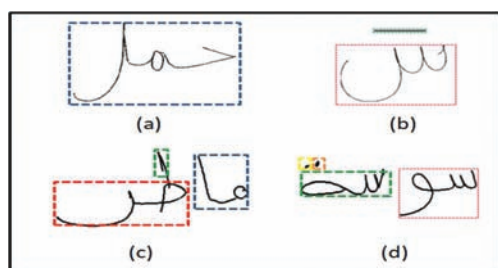


Figure 6 | Samples of handwritten scripts.



Figure 7 | The same Arabic script with different diacritic signs position (colored in red).

Yet, these scripts differ in meaning (stroke number 3 has the position number 1 in scripts to scoop up and position 4 in script to play). Basically, spatial relations can be subdivided into three major categories: topological relations, directional relations and metric relations. Nevertheless, there are many works that only use directional relations or topological ones and others that combine them together. Their choice depends on the kind of application used. In previous works, spatial relation is often defined by computing some geometric features such as, angle computation, centroids point, minimum bounding box of hand-drawn primitives, etc. For instance, an attempt to express complex spatial relations by means of geometric features and bounding box is recorded by Simistira [36]. This information is used to check the structure within mathematical symbols. The relationships are then expressed in a quantitative way and it might be easy to compute similarities between them. Nevertheless, these spatial relations come often without capturing their imprecision or their expression. In this context, some methods based upon fuzzy logic are proposed for the treatment of the relations where a fuzzy membership value reflects the measure of satisfaction in a fuzzy region [34,37].

Inspired from these works, we propose a fuzzy logic approach where a set of feature inputs that affect specific outputs are identified for given strokes. In this work, we deliberately choose to exclude a set of 6 geometric features $\{F_g | 1 \leq g \leq 6\}$ based on bounding boxes and centroid to exploit spatial relations. Considering two successive strokes S and S_1 illustrated in Figure 8, the features are computed as follows:

- F1: angle computed between two bounding boxes' centroid denoted α
- F2: distance between the y-coordinates of two centroids denoted d_4
- F3: distance between the x-coordinates of two centroids denoted d_3
- F4: distances between bottom right boundaries of bounding boxes denoted d_2
- F5: distance between top left boundaries of bounding boxes denoted d_1
- F6: The intersection between two strokes denoted by 0 or 1

Noting that a bounding box coordinates of stroke are computed as $x_1^S = \min(x_i^S), x_2^S = \max(x_i^S), y_1^S = \min(y_i^S)$ and $y_2^S = \max(y_i^S)$, the centroid c is defined as $x_c = (x_2^S - x_1^S)/2$ and $y_c = (y_2^S - y_1^S)/2$.

Basically, we developed a set of linguistic variables for the six features inputs and their fuzzy sets (e.g., Negative Big, Negative Small, Zero, Positive Small and Positive Big for angle feature "F1," Far or Near for "F5," etc.) as well membership functions to initiate the fuzzification process. Therefore, the rule blocs is done by combining these inputs. In our case, we identified 10 rules for the designed system, whereby different combinations of inputs features asses the degree of satisfaction of a spatial output. As a result, each spatial relation will be codified with a vector of 6 values. As a matter of fact, an online handwriting will be scripted by a matrix of different membership degrees of all spatial relations existing within strokes.

These relations are portrayed in Figure 9. For more details about calculation of spatial relations, we refer the reader to our previous work [38].

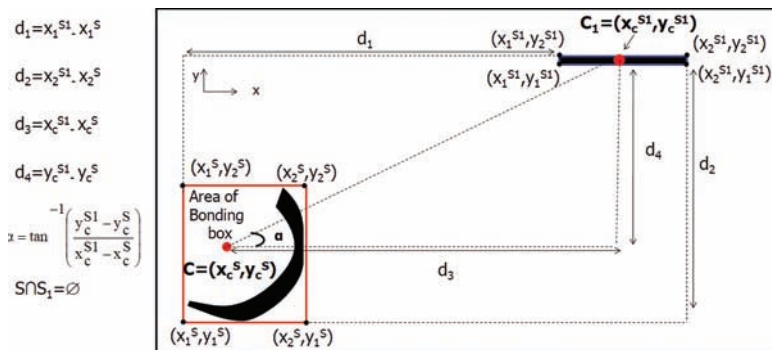


Figure 8 Geometrical features between two handwritten strokes S and of Arabic character with diacritical marks.

3.2. Script Recognition

Based on the stroke classification result and spatial relation modeling, the *i*-th stroke will be represented as (p_i, sp_i) where p_i and sp_i stand for the stroke classification result and spatial relation respectively. Finally, each *S* input script with *k* stroke can be expressed in terms of a feature vector as follows:

$$S = [(p_1, sp_1) \dots (p_k, sp_k)] \tag{9}$$

Once all scripts have been embedded into a vector, the corresponding vector can be used as input to an appropriate SVM. For this reason, we start with classifying automatically all the input scripts into a sub set based on the number of strokes *N*. Eventually, for each value of *N*, multiple SVMs are trained as depicted in Section 2.2 and the details shall be described in Section 4.

4. EXPERIMENTAL RESULTS AND EVALUATION

In this section, the experiments that were carried out in order to assess the designed system are portrayed. Both stroke classification and script recognition are performed on ADAB and MAYASTROUN data sets. It is worth noting that, all experiments in this paper were performed using python running on a 2.80 GHz with Intel(R) Core i7-7500, 64 bits laptop with 8 GB running on windows 10. In what follows, the details of datasets are described; experimental setups and the results are exhibited. Finally, a brief comparison with the state art models is established.

4.1. Data Set

For the purpose of experimental evaluation, two state of arts MAYASTROUN and ADAB datasets were used and will be described later in detail.

4.1.1. MAYASTROUN data set

MAYASTROUN is a large data set of online handwritten scripts, digits, western scripts in lower and upper case, Arabic texts and scripts, mathematical expressions, symbols and scripts in different languages and signatures. Some instances of this data set are illustrated in Figure 10. This data set was developed in the Research

S above_left S_1 S_1 below_right S	S above_right S_1 S_1 below_left S	S intersect S_1 S_1 intersect S
S parallel S_1 S_1 parallel S	S contains S_1 S_1 inside S	S left S_1 S_1 right S

Figure 9 Ten kinds of spatial relationships between two strokes.

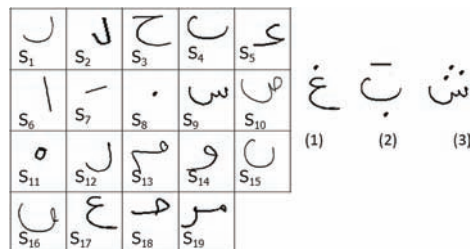


Figure 10 (a) Example of nineteen strokes and (b) example of the corresponding class of three scripts (for each class an example is shown).

Groups in Intelligent Machines laboratory (REGIM-Lab). Further details are provided in [39]. It contains up to 1000 scripts written by more than 350 different writers mainly of Tunisian nationality. A subset was selected to perform stroke classification as well as script recognition. This subset is totally composed of 1500 scripts. We derived 3500 strokes from these scripts and chose nineteen basic strokes as displayed in Figure 10.

The obtained stroke set is uniformly distributed over 19 classes, i.e., basic strokes and the resulting set is split into a testing set of size 1000 and a training set of size 2500. The obtained scripts sets are uniformly distributed over 3 classes. This implies that all scripts are divided into 3 groups varying from 2 to 4 according to the number of stroke. Finally, the resulting script set is split into a training sets of size 1000 and a testing set of size 500. Table 1 summarizes the main properties of these different data sets.

4.1.2. ADAB Data Set

The second data set, called ADAB, was developed by research Groups in Intelligent Machines, Tunisia and Institut fuer Nachrichtentechnik (IfN). It was collected through the use of digital tablets connected to computers by more than 130 different writers, most of them of Tunisian nationality. This data set contains 21575 online Tunisian town and village names in Arabic languages [40] which are distributed in 6 subsets. It is noteworthy, each sample is registered in "UPX" file format containing its sequence of x-y trajectory and details about the writer, number of files, characters and word label (see Table 2).

Departing from this data set, two subsets are collected i.e. strokes and scripts as presented in Figure 11.

The strokes data set is divided into 28 classes (Table 3). A testing set of size 2000 and a training size 3000 are used. Thus, there are totally 5000 strokes. Hence, these data sets contain 2 subsets training with 4000 scripts and testing with 3000 scripts collected randomly from

Table 1 | MAYASTROUN subsets details.

Subsets	Classes	Size	
		Training	Testing
Strokes	19	2500	1000
Scripts	3	1000	500

Table 2 | ADAB data set details.

	Number of Words	Number of Pseudo Words	Writers
Set 1	5037	40269	56
Set 2	5090	25450	37
Set 3	5031	15039	39
Set 4	4417	22085	25
Set 5	1000	4000	6
Set 6	1000	8000	3
Total	21575	114924	166

Table 3 | ADAB subsets details.

Subsets	Classes	Size	
		Training	Testing
Strokes	28	3000	2000
Scripts	8	4000	3000



Figure 11 | (a) Example of twenty eight strokes and (b) example of the corresponding class of eight scripts (for each class an example is shown).

different sets. Finally, the scripts set is uniformly distributed over the 8 classes, i.e., stroke number varying from 3 to 10 strokes.

4.2. Stroke Classification Experiments

Experiments were carried out according to the previously introduced approach and are introduced in this section.

- SVM parameters tuning

It's noteworthy that we opted to use Gaussian kernel so it is interesting to observe how important the careful choice of the cost parameters C and the Gaussian parameter σ is. Indeed 10-fold cross validation basis were performed for adjusting C and σ parameters. Thus tests were carried out on both MAYASTROUN and ADAB data sets with $C \in (0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1)$ and $\sigma \in (0.001, 0.01, 0.025, 0.05, 0.1)$. In each test, the couple giving the best accuracy was selected.

As reported in Figure 12, it can be inferred that $\sigma = 0.05$ with a high C value led to the best results, i.e., a high classification accuracy around 98% and 97% for MAYASTROUN and ADAB respectively. In order to further assess our findings, Tables 4 and 5 demonstrate the results of different parameters C and σ for the training and the test data of MAYASTROUN and ADAB data sets. In our analysis, the more interesting results are those recorded where SVM was trained and tested with $\sigma = 0.05$ and $C = 1$ which clearly outperforms other values in the experimentation.

- Experimentation on the feature involved in the stroke classification

Our classification method was also controlled by two different features: (1) BEM feature and (2) EPCs features that describe an input stroke. In order to clarify how these features control the performance of the system, we plotted the classification accuracy for each stroke class by using different combinations of these features. Departing from Figure 13, we notice that for both data sets, the proposed hybrid features improved the classification accuracy. Using BEM, the classification accuracy reaches 96.87% and 96.04%

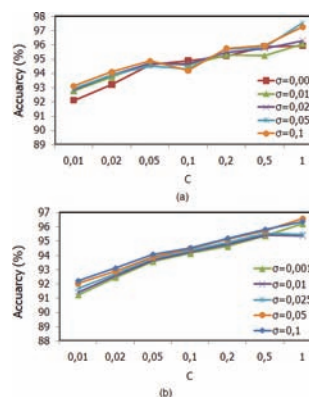


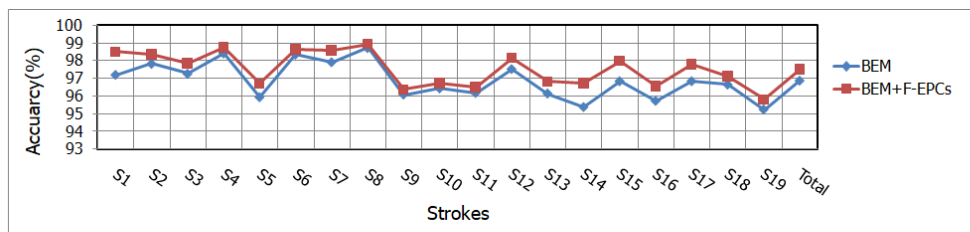
Figure 12 | Classification accuracy: (a) MAYASTROUN data sets and (b) ADAB data set.

Table 4 | Error rates on MAYASTROUN data set.

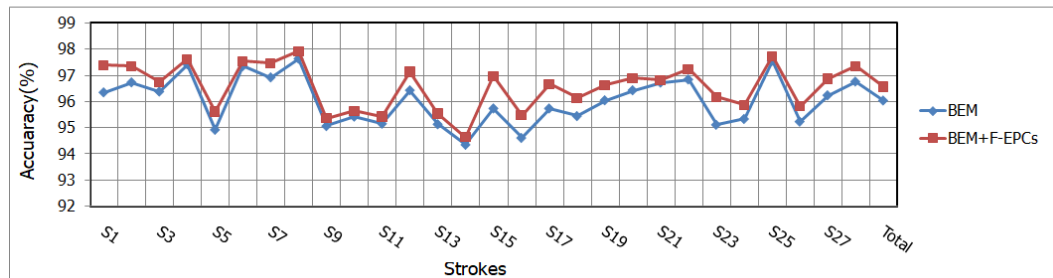
C	$\sigma = 0.001$		$\sigma = 0.01$		$\sigma = 0.025$		$\sigma = 0.05$		$\sigma = 0.1$	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
0.01	10.1	8.77	9.62	8.57	8.56	8.33	8.23	7.96	7.91	7.77
0.02	7.86	7.54	8.23	7.42	7.83	7.29	7.84	7.11	6.93	6.88
0.05	6.8	6.44	6.71	6.32	6.75	6.19	6.73	6.1	6.2	5.92
0.1	6.23	5.87	5.9	5.73	6.2	5.65	5.81	5.52	5.73	5.48
0.2	6.12	5.38	6.03	5.22	5.82	5.09	5.21	4.86	5.2	4.79
0.5	4.87	4.62	5.71	4.58	4.73	4.44	4.42	4.27	4.68	4.18
1	4.4	3.79	4.83	4.64	4.8	4.52	4.12	3.43	5.19	3.62

Table 5 | Error rates on ADAB data set.

C	$\sigma = 0.001$		$\sigma = 0.01$		$\sigma = 0.025$		$\sigma = 0.05$		$\sigma = 0.1$	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
0.01	8.1	7.9	7.4	7.25	7.31	7.17	7.19	7.04	7.23	6.9
0.02	7.32	6.8	6.6	6.28	6.42	6.14	6.52	6.1	6.34	5.93
0.05	6.3	5.4	5.81	5.27	5.49	5.33	5.7	5.47	5.6	5.14
0.1	5.7	5.14	5.7	5.43	5.7	5.35	5.93	5.71	6.1	5.81
0.2	5.1	4.8	5.01	4.73	4.8	4.54	4.7	4.32	4.73	4.27
0.5	4.81	4.12	4.9	4.77	4.62	4.26	4.62	4.17	4.61	4.08
1	4.3	4.09	4.32	3.96	4.28	3.73	2.93	2.48	3.31	2.76



(a)



(b)

Figure 13 | Performances of stroke classification for different combinations of the features: (a) MAYASTROUN data set and (b) ADAB data set.

in MAYASTROUN and ADAB, respectively; whereas hybrid feature raised the classification up to 1%. Although the performance improvements were not as large in such classes as S_5 which dropped slowly from 95.93% with BEM vs. 96.71%, with BEM + F-EPCs in MAYASTROUN. Moreover, we detected many trends of the improvement in many other classes such as S_{11} accuracy with BEM+F-EPCs 96.97% which is proved to be greater than that of BEM 95.74% in ADAB. This indicates that the F-EPCs features can be complementary to the BEM ones in classifying strokes for online HR. To this extent, it's worth noting that the obtained results go in good agreement with those reported in literature as provided by previous works [28].

4.3. Script Recognition

For the recognition stage of handwritten scripts, experiments were performed on scripts collected from both data sets. The concept was to gather the output probability of stroke of the same scripts jointly with spatial relations between them in order to recognize the script label. Experiments were carried out on SVM with Gaussian basis function kernel. The details analysis of the script recognition performance using SVM is presented in Table 6. It can be inferred that both data sets highlight an impressive recognition rate which exceeds 90%.

- Analysis under noisy condition

Performance evaluation of the designed recognizer system is evaluated by applying randomly noise signal. To address this issue, we proceeded by adding a Gaussian noise in training as well as testing sub sets. Note that Gaussian noise depends largely on the variability of the variance parameter called sigma σ . We noticed that σ varied for two values, namely (0.01 and 0.02). The trajectory profile remained unchanged for the change rate of ($\sigma = 0.01$). This effect was more visible in high values where the handwritten shape is slightly distorted ($\sigma = 0.02$) (Figure 14).

Since $\sigma = 0.02$ the recognition rate was computed in training and testing phases and presented in Table 7. The results highlight that handwritten script recognition can be affected by noise. A deeper analysis revealed that the recognition rate on MAYASTROUN and ADAB reached 99.24% and 98.53% on quiet condition vs. 95.24% and 93.83% on noisy condition. Despite these results our findings concerning the efficiency of the recognizer system prove to be still valid as the recognition rates are commonly around 90%.

- Comparison with state of art works

As already stated, many research works are available for online Arabic script recognition. Thus, we have compared the recognition performance of the proposed method with the existing system on ADAB and MAYASTROUN datasets to obtain an idea of comparative performance analysis. The considered systems are reported in the following:

- **Tagougui et al.** [41] set forward hybrid model based on hidden Markov models (HMMs) and multi-layer perceptron neural network (MLPNN) for online Arabic handwritten script recognition. Relying on the Beta-elliptic strategy, each input signal is subdivided into continuous strokes called segments. An MLPNN trained with different obtained segments is

Table 6 Script recognition results using support vector machine (SVM).

Data Set	Accuracy	
	Training (%)	Testing (%)
MAYASTROUN	99.78	99.24
ADAB	99.14	98.53

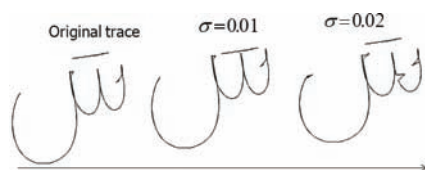


Figure 14 | Handwritten scripts and the obtained ones with proportional additional noise.

Table 7 | Script recognition results using support vector machine (SVM) under noisy conditions.

Data Set	Accuracy	
	Training (%)	Testing (%)
MAYASTROUN	96.68	95.24
ADAB	94.32	93.82

constructed to generate class script probabilities as an output. This output is encoded with HMMs to provide script level recognition. This architecture was tested on sets 4, 5 and 6 of ADAB data set and fulfilled a recognition rate around 96.45%.

- **Boubaker et al.** [42] developed a modeling-recognition system for online Arabic handwriting scripts. A grapheme segmentation task based on baseline detection as well as median zone width estimation is used for modeling handwritten trajectories. Therefore, recognition is performed using HMM on ADAB data set and fulfilled recognition rate of 87.46% for learning and 85.37% for testing.
- **Khelif et al.** [43] identified a new model for online Arabic handwritten word recognition with two levels of segmentation. The first one is a segmentation free-based system that relies on using the generative classifier HMM. The second level relies on explicit grapheme segmentation where both online and offline parameters are identified and classified with SVMs.
- **Hamdi et al.** [44] proposed to combine MLPNN with a genetic algorithm for online HR model. The MLPNN is trained with a collection of grapheme obtained in segmentation stage using BEM. A search technique, i.e., a genetic algorithm is applied to generate the recognized graphemes code.
- **Elleuch et al.** [45] constructed a system for online HR based on a hybrid BEM and convolution deep belief network (CDBN) offline feature extractor models.
- **Abdelaziz et al.** [46] proposed an HMM-based system for online Arabic handwriting in which they attempt to handle delayed strokes. On a large vocabulary data set, HMMs are trained and tested with a collection of geometric features (chain code, aspect ratio, loops, hat, etc.) and reached a recognition rate around 97.13%.
- **Akouadi et al.** elaborated in 2019 two online Arabic systems [28,29]. The first system was performed on 5 data sets and showed very promising results that exceed 98% while the second system was tested on two data sets (MAYASTROUN and ADAB) and reached a recognition rate of 98.90% and 98.41% respectively.
- **Zouari et al.** [43] proposed to combine time delay neural network (TDNN) with SVM for online HR model. The TDNN-SVM is trained and tested with a collection of features vectors obtained in segmentation stage using BEM so as to describe continuous trajectories of scripts. The recognition system was tested on multi-languages online data set and achieved a recognition rate around 100%.
- **Maalej et al.** [13] identified a new model for online Arabic HR based on deep long short-term memory (DBLSTM) that relying on two modes of Maxout integration. First, the authors added the Maxout function inside the LSTM nodes and set the size of maxout groups to 3. Thus, the error rate reduction reached 10.62%. Second, they proposed to stack Maxout layers after BLSTM layers. This architecture proved to outperform the first one as it permits a reduction of 10.99% in the label error rate.

The obtained results are summarized in Table 8. From the obtained results, we can infer how promising our system performance is compared to the state-of-the-art systems. First of all, ten different

Table 8 | State-of-the-art results.

	Authors	Features Extraction	Classifier	Recognition Rate (%)
ADAB	Tagougui [41]	Beta-elliptic features (Dynamic + static)	MLP-HMM	96.45
	Boubaker [42]	Geometric parameters + Fourier descriptors	HMM	85.37
	Khelif [43]	Online + offline feature	HMM	93.33
	Hamdi [44]	Beta-elliptic model	MLP	95.14
	Elleuch [45]	Online + offline feature	SVM	91.8
	Abdelaziz [46]	Geometric features	HMM	97.13
	Akouadi [28]	Beta parameters + perceptual codes	BCP + LSTM	90
	Zouari [47]	Beta-elliptic parameters	TDNN-SVM	100
	Maalej [13]	x, y coordinates	BLSTM	99.98
	Proposed	Beta point + perceptuel codes + spatial relation	SVMs	98.53
MAYASTROUN	Akouadi [29]	Coordinate (x, y, z)	Deep CNN	85.5
	Akouadi [28]	Beta point + perceptual codes	BCP + LSTM	94.50
	Akouadi [29]	Beta parameters + CNN + perceptual codes	Deep CNN	98.90
	Proposed	Beta point + perceptuel codes + spatial relation	SVMs	99.24

MLP, multi-layer perceptron; HMM, hidden Markov model; SVM, support vector machine; LSTM, long short-term memory; TDNN, time delay neural network.

classifier/feature combinations were tried on different sets of ADAB data set. The comparison reveals that our method has much better performance than some approaches relying on a single classifier (98.53% with SVM-based classifier(our approach) vs. 95.14% with MLP and 91.8% with SVM). This stands for an improvement of 3.39% and 6.37% respectively. In Refs [42,43,46] presented systems are using all the same classifier (HMMs) with different features as input; the recognition rate for the results of the systems mentioned does not exceed 97.2%, whereas the proposed system outperforms the results and achieves 98.53%. The obtained remains valid even though it is compared to hybridized classifier (98.53% with SVM vs. 96.45% with MLP-HMM, 90% with BCP-LSTM). However, we inferred that our system lost its first place against two recent systems [13,47] with a recognition rate around 100% for both. This can be accounted for in terms of using multi-language data set (contained scripts not only from ADAB data sets) as well as different sets on training and testing(different sets from ADAB) in Refs. [13] and [47] respectively which prevents performing a thorough comparison. Furthermore, Table 8 discloses a comparative study of the performance of our method on MAYASTROUN data set. This comparison resulted in overall accuracy of 94.50% with BCP-LSTM and 85.5% and 98.90% with Deep CNN whereas the proposed system outperforms the results and achieves 99.24%. Noting that, either we use the same data set and the same classifier as in [45]; our method achieves the best results. This proves that the features used do possess a discriminative power on recognition results.

Moreover, segmenting each handwritten script into a sequence of stroke permits to calculate features of each stroke separately. This provides a local description of the script component and allows recognizing scripts by considering their stroke class labeling scheme. Furthermore, incorporating spatial relation features on the script recognition process empowers the obtained accuracy.

5. CONCLUSION AND FUTURE PERSPECTIVES

This paper elaborates a new architecture of online handwritten script recognition system. Our work proceeds by segmenting each

input script into strokes denoting the block between pen up and pen down. Indeed, we suggested the incorporation of merits of both BEM and F-EPCs on modeling those strokes. Scanning through literature, many machine learning have been set forward. For training and classifying strokes SVM was chosen thanks to its ability to handle sequential data. To enhance the recognition performance, spatial relations within handwriting strokes were analyzed. Subsequently, the membership probabilities vector of spatial relationship belonging to the same script was reunited with stroke probability to transform the input data into a second SVM classifier so as to recognize the whole Arabic handwritten script. Comprehensive experiments using two Arabic handwritten scripts data sets ADAB and MAYASTROUN proved that the proposed combination approach can considerably brush up the classification accuracy. In fact, we can realize that our method has achieved a very promising recognition rate of 98.53% (Script error rate of 1.47%) and 99.24% (Script error rate of 0.76%) when practiced to the ADAB and MAYASTROUN data sets respectively. Resting on the results recorded in this work, it is worth noting that this framework is able to properly recognize handwritten Arabic scripts. First of all, we note that the incorporation of the sophisticated segmentation technique (i.e., BEM) allows not only improving the accuracy but also addressing the touching scripts problem. Second, integration fuzzy logic on perceptual elements permits to deal with variability of writing and uncertainty data. Furthermore, structural information can help enhance the initial Arabic recognition and spatial relations decisions. This property permits the use of any kind of data and makes it reproducible to any other PR task. In addition, in order to confirm the effectiveness and reliability of this approach a comparison established among various states of the art on both data sets is conducted and summarized in the following points: (1) In a series of experimental evaluations, we conclude that our method is better than the most state-of-the-art HR methods on both data sets. This is mostly substantial with reference to methods which are not based on BEM features (e.g., refs [42,43,45] achieving the highest ranging error rates between 6.67% to 14.63% vs. 1.47% obtained with our system on ADAB data set). (2) Compared to works [28,29,41,44] dealing with beta parameter/perceptual codes, our work still achieves the best recognition rate with an overall improvement between 2.08 and

8.53% on ADAB data set and between 0.34% and 4.74% on MAYAS-TROUN data set. This proves the effectiveness of spatial relations in HR systems. (3) With two participating systems [13,47], the recognition rates are slightly higher (around 100%) on ADAB data set. This stands for an improvement < 2% relative to our method. This outcome refers to the different size [47] and languages [13] added to training and testing data on these systems. To sum up, we would simply assert that tackling this area is extremely interesting as it lays the ground and paves the way for constructive and fruitful applications in the field. From this perspective, our work provides an embryonic notion for future works to investigate more robust features classifiers so as to train deep neural networks for tackling multi-language handwritten scripts.

CONFLICTS OF INTEREST

The authors declare that they have no conflict of interest.

AUTHORS' CONTRIBUTIONS

Rabaa Zitouni: Conceptualization, Methodology, Software, Formal analysis, Writing - original draft. Hala Bezzine: Methodology, Investigation, Writing - review & editing, Visualization. Najet Arous: Validation, Formal analysis, Supervision

Funding Statement

This research received no external funding.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve the quality of the article

REFERENCES

- [1] S. Impedovo, More than twenty years of advancements on frontiers in handwriting recognition, *Pattern Recognit.* 47 (2014), 916–928.
- [2] N. Tagougui, M. Kherallah, A.M. Alimi, Online Arabic handwriting recognition: a survey, *Int. J. Doc. Anal. Recognit.* 16 (2013), 209–226.
- [3] M. Ahmed, A.G. Rasool, H. Afzal, I. Siddiqi, Improving handwriting based gender classification using ensemble classifiers, *Expert Syst. Appl.* 85 (2017), 158–168.
- [4] Y. Chherawala, P.P. Roy, M. Cheriet, Feature set evaluation for offline handwriting recognition systems: application to the recurrent neural network model, *IEEE Trans. Cybern.* 46 (2015), 2825–2836.
- [5] D. Keysers, T. Deselaers, H.A. Rowley, L.-L. Wang, V. Carbune, Multi-language online handwriting recognition, *IEEE Trans. Pattern Anal. Mach. Intell.* 39 (2016), 1180–1194.
- [6] O. Samanta, U. Bhattacharya, S.K. Parui, Smoothing of HMM parameters for efficient recognition of online handwriting, *Pattern Recognit.* 47 (2014), 3614–3629.
- [7] H. Choudhury, S.R. Mahadeva Prasanna, Representation of online handwriting using multi-component sinusoidal model, *Pattern Recognit.* 91 (2019), 200–215.
- [8] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke, J. Schmidhuber, A novel connectionist system for unconstrained handwriting recognition, *IEEE Trans. Pattern Anal. Mach. Intell.* 31 (2008), 855–868.
- [9] Y.-C. Wu, F. Yin, C.-L. Liu, Improving handwritten Chinese text recognition using neural network language models and convolutional neural network shape models, *Pattern Recognit.* 65 (2017), 251–264.
- [10] X.-Y. Zhang, Y. Bengio, C.-L. Liu, Online and offline handwritten Chinese character recognition: a comprehensive study and new benchmark, *Pattern Recognit.* 61 (2017), 348–360.
- [11] A. Bharath, S. Madhvanath, HMM-based lexicon-driven and lexicon-free word recognition for online handwritten indic scripts, *IEEE Trans. Pattern Anal. Mach. Intell.* 34 (2011), 670–682.
- [12] R. Ghosh, C. Vamshi, P. Kumar, RNN based online handwritten word recognition in Devanagari and Bengali scripts using horizontal zoning, *Pattern Recognit.* 92 (2019), 203–218.
- [13] R. Maalej, M. Kherallah, Improving the DBLSTM for on-line Arabic handwriting recognition, *Multimedia Tools Appl.* 79 (2020), 1–22.
- [14] R. Zouari, H. Boubaker, M. Kherallah, RNN-ISTM based beta-elliptic model for online handwriting script identification, *Int. Arab J. Inf. Technol.* 15 (2018), 532–539.
- [15] H. Althobaiti, C. Lu, A survey on Arabic optical character recognition and an isolated handwritten Arabic character recognition algorithm using encoded freeman chain code, in *Proceeding of IEEE 51st Annual Conference on Information Sciences and Systems (CISS)*, Johns Hopkins University, Baltimore, MD, USA, 2017, pp. 1–6.
- [16] M. Kherallah, N. Tagougui, A.M. Alimi, H. El Abed, V. Margner, Online Arabic Handwriting Recognition Competition, in *Proceeding of 11th International Conference on Document Analysis and Recognition*, Beijing, China, 2011. pp. 1454–1458.
- [17] F. Slimane, S. Kanoun, H. El Abed, A.M. Alimi, R. Ingold, J. Hennebert, ICDAR 2011-Arabic recognition competition: multi-font multi-size digitally represented text, in *Proceeding of 11th International Conference on Document Analysis and Recognition*, Beijing, China, 2011, pp. 1449–1453.
- [18] A.K. Jain, R.P.W. Duin, J. Mao, Statistical pattern recognition: a review, *IEEE Trans. Pattern Anal. Mach. Intell.* 22 (2000), 4–37.
- [19] J. Memon, M. Sami, R.A. Khan, M. Ud-din, Handwritten optical character recognition (ocr): A comprehensive systematic literature review (slr), *IEEE Access*, 8 (2020), 142642–142668.
- [20] S. Njah, M. Ltaief, H. Bezzine, A.M. Alimi, The pertohs theory for on-line handwriting segmentation, *Int. J. Comput. Sci. Issues.* 9 (2012), 142–151.
- [21] I.E. Gordon, *Theories of Visual Perception*, Psychology Press, London, 2004.
- [22] S. Njah, H. Bezzine, A.M. Alimi, A new approach for the extraction of handwriting perceptual codes using fuzzy logic, in *ICFHR*, Montréal, Québec, 2008, pp. 302–307.
- [23] T. Dhieb, S. Njah, H. Boubaker, W. Ouarda, M.B. Ayed, A.M. Alimi, An extended beta-elliptic model and fuzzy elementary perceptual codes for online multilingual writer identification using deep neural network, *arXiv preprint arXiv:1804.05661*, 2018.

- [24] T. Dhieb, S. Njah, H. Boubaker, W. Ouarda, M.B. Ayed, A.M. Alimi, Towards a novel biometric system for forensic document examination, *Comput. Secur.* 97 (2020), 101973.
- [25] H. Bezine, A.M. Alimi, Handwriting perceptual classification and synthesis using discriminate HMMs and progressive iterative approximation, *Neural Comput. Appl.* 32 (2020), 1–22.
- [26] H. Bezine, W. Ghanmi, A.M. Alimi, A HMM model based on perceptual codes for on-line handwriting generation, in *The Sixth International Conference on Advanced Cognitive Technologies and Applications*, Citeseer, Venice, Italy, 2014, pp. 126–132.
- [27] H. Akouaydi, S. Abdelhedi, S. Njah, M. Zaied, A.M. Alimi, Decision trees based on perceptual codes for on-line Arabic character recognition, in *2017 1st International Workshop on Arabic Script Analysis and Recognition (ASAR)*, IEEE, Nancy, France, 2017, pp. 153–157.
- [28] H. Akouaydi, S. Njah, W. Ouarda, A. Samet, T. Dhieb, M. Zaied, A.M. Alimi, Neural architecture based on fuzzy perceptual representation for online multilingual handwriting recognition, *arXiv preprint arXiv:1908.00634*, 2019, 1–14.
- [29] H. Akouaydi, S. Njah, W. Ouarda, A. Samet, M. Zaied, A.M. Alimi, Convolutional neural networks for online Arabic characters recognition with beta-elliptic knowledge domain, in *Proceeding of 15th International Conference on Document Analysis and Recognition*, Sydney, Australia, 2019, pp. 41–46.
- [30] H. Bezine, A.M. Alimi, N. Sherkat, Generation and analysis of handwriting script with the beta-elliptic model, in *Proceeding of 9 International Workshop on Frontiers in Handwriting Recognition (IWFHR)*, Kokubunji, Tokyo, Japan, 2004, pp. 515–520.
- [31] H. Bezine, M. Kefi, A.M. Alimi, On the beta-elliptic model for the control of the human arm movement, *Int. J. Pattern Recognit. Artif. Intell.* 21 (2007), 5–19.
- [32] V.N. Vapnik, *The Nature of Statistical Learning Theory*, Springer-Verlag, Berlin, Heidelberg, Germany, 1995.
- [33] C.J.C. Burges, A tutorial on support vector machines for pattern recognition, *Data Mining Knowl. Discov.* 2 (1998), 121–167.
- [34] A. Delaye, E. Anquetil, Learning of fuzzy spatial relations between handwritten patterns, *Int. J. Data Mining Model. Manag.* 6 (2014), 127–147.
- [35] M. Clément, C. Kurtz, L. Wendling, Learning spatial relations and shapes for structural object description and scene recognition, *Pattern Recognit.* 84 (2018), 197–210.
- [36] F. Simistira, V. Katsouros, G. Carayannis, Recognition of online handwritten mathematical formulas using probabilistic SVMS and stochastic context free grammars, *Pattern Recognit. Lett.* 53 (2015), 85–92.
- [37] I. Bloch, Fuzzy models of spatial relations, application to spatial reasoning, in: R. Seising, E. Trillas, C. Moraga, S. Termini (Eds.), *Studies in Fuzziness and Soft Computing*, vol. 298, Springer, Berlin, Heidelberg, Germany, 2013, pp. 51–58.
- [38] Z. Rabiaa, B. Hala, A. Najet, Fuzzy rules-based system for description spatial relations among handwriting strokes, in *Proceeding 11th International Conference on Machine Vision*, Munich, Germany, 2018, pp. 466–471.
- [39] S. Njah, B.B. Nouma, H. Bezine, A.M. Alimi, Mayastroun: a multilanguage handwriting database, in *Proceeding of 12th International Conference on Frontiers in Handwriting Recognition*, Bari, Italy, 2012, pp. 308–312.
- [40] H. El Abed, M. Kherallah, V. Märgner, A.M. Alimi, On-line Arabic handwriting recognition competition, *Int. J. Doc. Anal. Recognit.* 14 (2011), 15–23.
- [41] N. Tagougui, H. Boubaker, M. Kherallah, A.M. Alimi, A hybrid NN/HMM modeling technique for online Arabic handwriting recognition, *Comput. Linguist. Res.* 4 (2013), 107–118.
- [42] H. Boubaker, N. Tagougui, H. El Abed, M. Kherallah, A.M. Alimi, Graphemes segmentation for Arabic online handwriting modeling, *JIPS.* 10 (2014), 503–522.
- [43] H. Khlif, S. Prum, Y. Kessentini, S. Kanoun, J.-M. Ogier, Fusion of explicit segmentation based system and segmentation-free based system for on-line Arabic handwritten word recognition, in *Proceeding of 15th International Conference on Frontiers in Handwriting Recognition*, Nagoya, Japan, 2016, pp. 399–404.
- [44] Y. Hamdi, A. Chaabouni, H. Boubaker, A.M. Alimi, Hybrid neural network and genetic algorithm for off-lexicon online Arabic handwriting recognition, in *Proceeding of 16th International Conference on Hybrid Intelligent Systems (HIS)*, Nagoya, Japan, 2016, pp. 431–441.
- [45] M. Elleuch, R. Zouari, M. Kherallah, Feature extractor based deep method to enhance online Arabic handwritten recognition system, in *Proceeding of 25th International Conference on Artificial Neural Networks*, Barcelona, Spain, 2016, pp. 136–144.
- [46] I. Abdelaziz, S. Abdou, H. Al-Barhamtoshy, A large vocabulary system for Arabic online handwriting recognition, *Pattern Anal. Appl.* 19 (2016), 1129–1141.
- [47] R. Zouari, H. Boubaker, M. Kherallah, Multi-language online handwriting recognition based on beta-elliptic model and hybrid TDNN-SVM classifier, *Multimedia Tools Appl.* 78 (2019), 12103–12123.