



Multi-language online handwriting recognition based on beta-elliptic model and hybrid TDNN-SVM classifier

Ramzi Zouari¹  · Houcine Boubaker¹ · Monji Kherallah²

Received: 17 February 2018 / Revised: 14 September 2018 / Accepted: 7 October 2018 /
Published online: 13 October 2018
© Springer Science+Business Media, LLC, part of Springer Nature 2018

Abstract

Recently, several researches were carried on handwritten document analysis field thanks to the evolution of data capture technologies. For a given document, multiple components could be treated as text, signatures and graphics. In this study, we present a new framework for a Multilanguage online handwritten text analysis where both script identification and recognition are made. The proposed system proceeds by segmenting the script into continuous trajectories delimited between two successive pen-down and pen-up moments. These segments are clustered and trained using Time Delay Neural Network (TDNN) according to their beta-elliptic parameters. In script identification process, the segments belonging to the same script are gathered and brought to a Recurrent Neural Network with Long Short Term Memory (RNN-LSTM) in order to identify its language. For script recognition stage, the samples from the already selected language database are trained and tested using the fuzzy output description obtained by the TDNN coupled to a Support Vector Machines (SVM). The Experiments were made on a large multi-language database containing 45686 online handwriting words from Latin, Arabic and digit scripts and shows very promising results that exceed the recognition rate of 99%.

Keywords Online handwriting · Multi-language · Beta-elliptic model · Velocity beta impulse · Pseudo-word · RNN-LSTM · TDNN · Receptive field

✉ Ramzi Zouari
ramzi.zouari@gmail.com

Houcine Boubaker
houcine-boubaker@ieee.org

Monji Kherallah
monji.kherallah@enis.mu.tn

¹ National School of Engineers of Sfax, University of Sfax, Sfax, Tunisia

² Faculty of Sciences of Sfax, University of Sfax, Sfax, Tunisia

1 Introduction

Online handwriting recognition is very useful in daily life in accordance with the increased number of handwriting documents. Influenced by the technological revolution in data entry devices, great interest is given for online document analysis against offline documents where only a scanned image of the handwritten is available. With online handwriting analysis, dynamic informations about the trajectory are available. In fact, the different strokes constituting a script could be extracted and we have access to the x , y coordinates in real time. Handwriting is a cursive script which represents the concatenation of a basic graphics shapes called letters. They are often connected through valleys joining the baseline. In some cases, a letter contains dots and diacritics in addition to its main body. They are commonly known as delayed strokes and they are frequently used in Arabic script and slightly less in Latin script. Furthermore, handwriting has the propriety of multi-variability existing both between and within writers. Indeed, the writing style changes from one person to another and even for the same writer depending to his actual state. Moreover, in most touch-sensitive devices, the handwriting input may contain errors consist of discontinuity in the trajectory, disconcerted and redundant points. Thus, a handwriting recognition system should include a preprocessing step to remove irregularities from trajectory in order to improve recognition performances. In document automation process, script identification aims identifying the language of writing, while recognition module allows recognizing the script. This requires a robust handwriting modeling method, especially with a multi-language recognition system. It arises when scripts have different characteristics as for Latin and Arabic languages. In fact, both writings are cursive but Latin words are written from left to right whereas Arabic script is written from right to left and most of letters change their shapes according to their position in the word. According to the previous works, scripts identification and recognition have been treated separately. In this paper, the main contribution consists on developing a multi-language recognition system dealing with Latin, Arabic and digit scripts based on beta-elliptic model which integrate a script identification stage. The beta-elliptic approach has been applied to extract a relevant combined dynamic and geometric features vector constituting a parametric description, of the online acquired trajectory, which is used to train a Time Delay Neural Network for clustering its composing pseudo-words. Thereafter, the crisp outputs are used by the RNN for the script identification and consequently target lexicon limitation whereas the fuzzy outputs are brought to the SVM for the script recognition stage. This paper is organized as follow: in the next section, the main online identification and recognition approaches are overviewed. Section 3 presents the application of the beta-elliptic approach for online handwriting modeling. In Section 4, the framework for multi-language online handwriting recognition is presented. Experimental results and discussions are given in Section 5, and finally we present a conclusion with some future works.

2 Related works

Handwriting modeling is a crucial step for online handwriting recognition process. In this context several computational models have studied the kinematics theory of rapid human movements. In 1981, Hollerbach proved that handwriting movement arises from superimposition of orthogonal oscillations along horizontal and vertical directions [16]. The vertical oscillations are responsible for producing writing shape while horizontal oscillations control the writing sweep. In the velocity domain, these oscillations are modulated by sinusoids

functions to control the corner shapes and the writing slant. In other study, Viviani et al. [39] proposed a new law relating the radius of the curvature at any point along the trajectory with the corresponding tangential velocity, known by two-third power law. In particular, the analysis of the development of motor control has demonstrated that velocity and curvature are always related by a power function that attains the value of $1/3$. Plamondon et al. [32] proposed a model based on the intrinsic representation of handwritten curves without reference to any fixed axis system. The curvilinear velocity have an asymmetric discontinuous bell-shaped profile that is fitted with two half Gaussian functions. In literature, several recognition systems are based on delta-lognormal model [33]. The writing movement is the result of two opposite agonist and antagonist neuromuscular subsystems characterized by their lognormal equations. The velocity profile can be described by the subtraction of lognormal curves. O'Reilly [30] proposed a sigma-lognormal model. However, the handwriting trajectory is considered as being a movement resulting from a single lognormal equation scaled and shifted over the time occurrence. In [2], Alimi supposed that writing speed is the result of the activation of N neuromuscular subsystems characterized by an impulse response that converges with a Beta curve shape. Thus, the velocity profile curve can be fitted by the algebraic sum of overlapped beta impulses. The approaches dealing with online handwriting recognition are generally classified into two categories: holistic and analytic. For analytic approaches, the input signal is segmented into sub-units according to significant points. Among these approaches, we can cite the system proposed by Samanta et al. [34] who presents a recognition approach based on Discrete Curve Evolution algorithm. Its principle consists on dividing the trajectory into sub-units according to a set of best contour points that preserve the visual shape of the curve. The temporal sequence of feature vectors forms the observation sequence for the Hidden Markov Model (HMM). Khelif et al. [23] presented a new model with two levels of segmentation. The first level relies on segmentation-free strategy. Indeed, for each point of the trajectory a set of temporal, directional and curvature features are computed and decoded with HMM. The second level relies on explicit grapheme segmentation where both online and offline parameters are extracted and classified with Support Vector Machines (SVM). These two systems are combined in order to take advantage of their complementarities. Several methods are proposed for Bangla script recognition. In [4] a valid segmentation point is based on the position of its left and right extrema points relative to a virtual midline. With this method, a character may produce at most 3 sub-strokes. Thereafter, a sequence of geometric parameters is extracted and trained with SVM. In other work, Ghosh et al. [12] presented a recognition system based on the segmentation of the handwritten curve into local zones of size $(n \times m)$. For each one, geometric features as direction, slopes and deviation are calculated. For the recognition stage, SVM is used with different size of local zones and reached an average rate of 92%. The same techniques of segmentation and feature extraction are reproduced with HMM [13], and show better results than those obtained with SVM. These two systems are also applied for Devanagari scripts and they have achieved the same performances. Zhu et al. [43] addressed the problem of online handwriting recognition for Latin scripts. They combined two methods of segmentation. The first method attempts to define a set of candidate segmentation points at handwritten boundaries while the second uses the MRF model to select the path with the optimal evaluation criterion value. The proposed approach outperforms the system presented in [11] where Recurrent Neural Network (RNN) is applied. In other study, Nguyen et al. [29] presented a semi-incremental recognition method where the segmentation candidate points are updated after receiving new written strokes. This technique is evaluated with SVM classifier and attained a word recognition rate of 70% on IAM online database.

Besides systems based on analytic approaches, some researchers also proposed systems based on holistic methods. In this case, the recognition process is done without segmentation. For Arabic script, Nakkach et al. [28] presented an approach for online handwriting recognition based on the extraction of normalized chain code and the Fourier descriptors. These features are evaluated with SVM classifier and attained a recognition rate of 92.43% on a database of 2000 characters. The same features are combined with Dynamic Time Warping (DTW) parameters and prove their effectiveness on ADAB database [10]. In [25], deep RNN with CTC network architecture has been proposed. The the input sequence data are the (x, y) coordinates which dropout technique was applied to protect the network against overfitting. Sun et al. [35] presented an online Chinese handwriting recognition method. They used the same features and architecture cited in [25] with a modified CTC beam search decoding algorithm to integrate some linguistic constraints. For the experiments, they used the public CASIA-OLHWDB2.0 database containing a set of 2764 classes written by 815 persons. The proposed approach outperforms the system of Zhou et al. [42] based on minimum-risk training method, with an improvement rate of 3% in test set. Yang et al. [41] adopted a Convolutional Neural Network (CNN) for Chinese character recognition. They incorporate it with a variety of domain specific knowledge, including deformation, non-linear normalization, imaginary strokes, path signature and 8 directional features. Their system achieved an accuracy of 97.20% and 96.87% on CASIA-OLHWDB1.0 and CASIA-OLHWDB1.1 respectively.

During the last decade, beta-elliptic approach has been applied for online handwriting recognition. The system proposed by Kherallah et al. [21] consists on extracting the visual codes corresponding to the arcs obtained from the beta-elliptic modeling. Thereafter, the basic operations of Genetic algorithm have been applied to select the combination of strokes with the high scores according to a fitness function. This approach attained a recognition rate of 99% on a LMCA database. In other work, hybridization between Multi-Layer Perceptron (MLP) and HMM is investigated [36]. The class character probabilities of the neural network are considered as the probability density functions emitted by the HMM states. Compared to a standard discrete HMM, the recognition rate was improved by 3.51%. In [20], beta-elliptic approach was applied for digit recognition. The interaction between fuzzy KNN, MLP and Kohonen Map contributes to improve performances. Elleuch et al. [9] combined beta-elliptic parameters with offline features using Convolutional Deep Belief Network (CDBN) classifier. The test results on LMCA and ADAB databases proved the effectiveness of this architecture compared to systems based on MLP [5] and HMM respectively [1].

The above algorithms have focused on recognition of individual language and few reports have been seriously made in a multilingual environment. The first work existing attempts to do recognition of multiple languages simultaneously using hierarchical HMM [24]. In fact, a HMM network is constructed by interconnecting basic components as characters and intermediate ligatures. HMM network is built for each language and they are combined into a single network structure. The recognition corresponds to finding optimal path in the network according to Viterbi algorithm. This architecture was applied for Korean and English scripts and attained recognition rates of 80.7% and 82.11% respectively. Recently, Keysers et al. [18] presented an online handwriting system that supports 22 scripts and has been publicly available in several Google products such as Google Translate. They used heuristic segmentation method based on the script language. Thereafter, a Neural Network has been trained to determine which hypothetical cut points are valid for character segmentation. Therefore, a segmentation lattice graph is built with the goal of determining the characters most likely to have been written. Experiments are performed on several publicly databases

and attained an average error rates of 4.3%. Indhu et al. [17] proposed a multilingual recognition system for Indian, Urdu and digit scripts. It is to highlight that Urdu script is written from right-to-left and it has several common letters with Arabic language. Their system is based on extracting some geometric features like trajectory coordinates, vicinity aspect and slope. The recognition phase is done using transition probability between the set of strokes forming a character and train them with Viterbi-algorithm in order to find the most probable output for a given sequence of strokes. This technique achieved a character recognition rate of 94%.

In this section, we have presented the main methods used for online handwriting recognition. Most of the segmentation methods proposed by the analytic approaches are based on geometric characteristics that could be done in offline mode. In our study, we exploit the dynamic aspect of the input data in order to split the handwritten on significant points corresponding to the extremums of velocity profile. One of the major contributions of this work is the implementation of a script identification step before the recognition module. Its purpose is to identify the language of the handwritten. Therefore the recognition process will be executed on the selected database.

3 Beta-elliptic modeling

The handwriting movement is considered as a skilled motor process. It is usually planned in advance and represented by bio-motor program that can be projected in the velocity domain. In this context, the generation of a complex trajectory is the result of the activation of n neuromuscular subsystems characterized by an impulse response that converges to a beta curve shape [2].

3.1 Beta velocity profile

Form handwritten trajectory coordinates (x, y) , the curvilinear velocity $V_\sigma(t)$ is obtained using a second-order derivative filter with finite impulse response:

$$V_\sigma(t) = \sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2} \tag{1}$$

The curvilinear velocity shows a signal that alternates a successive extremums of velocity and inflection points. These points are considered significant since they are used to segment the handwritten into simple movements called strokes. In velocity domain, each stroke converges to a beta curve shape. So, the number n of strokes is equal to the number of beta impulses and the generation of a complex trajectory pattern is the result of an algebraic addition of these n strokes (see Fig. 1).

$$V_\sigma(t) \approx \sum_{i=1}^n K_i \times \beta_i(t, q_i, p_i, t_{0i}, t_{1i}) \tag{2}$$

with

$$\beta_i(t, q_i, p_i, t_{0i}, t_{1i}) = \begin{cases} \left(\frac{t-t_{0i}}{t_{ci}-t_{0i}}\right)^{p_i} \left(\frac{t_{1i}-t}{t_{1i}-t_{ci}}\right)^{q_i} & \text{if } t \in [t_{0i}, t_{1i}] \\ 0 & \text{elsewhere} \end{cases} \tag{3}$$

$$t_{ci} = \frac{(p_i \times t_{1i}) + (q_i \times t_{0i})}{p_i + q_i} \tag{4}$$

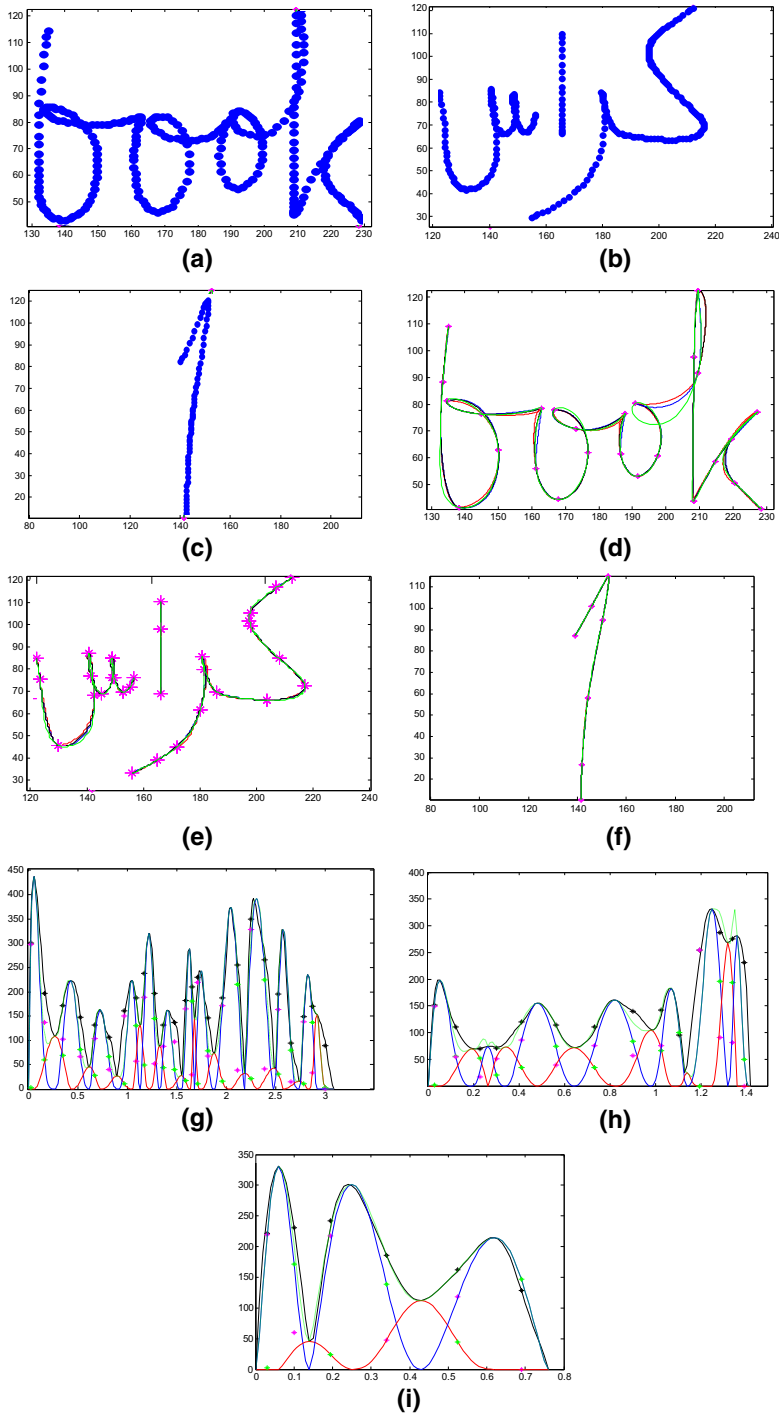


Fig. 1 a, b, c original handwritten trajectories of Latin word ‘book’, Arabic word ‘koras’ and digit ‘1’. d, e, f Trajectories rebuilding by elliptic arcs. g, h, i. velocity profile modeling by overlapped beta impulses strategy

where p_i, q_i are intermediate parameters which have an influence on the symmetry and the width of beta shape, t_{0i} the starting time of i^{th} beta function, t_{ci} the instant when it reaches the culmination amplitude and t_{1i} the ending time of i^{th} beta function.

3.2 Trajectory modeling

In the space domain, several approaches have been proposed for cursive handwriting generation. They commonly segment the trajectory into strokes located between two successive extrema speed times t_1 and t_2 . As proposed by Bezine et al. [7], each stroke executed from an initial position M_1 at time t_1 and moving to position M_2 at time t_2 is assimilated to an elliptic path verifying equation 5 where X and Y are the cartesian coordinates along the elliptic stroke, a and b are the small and large axis dimensions.

$$\frac{X^2}{a^2} + \frac{Y^2}{b^2} = 1 \tag{5}$$

Kherallah et al. [19] proposed a basic elliptic model where the segmented stroke is assimilated to a quarter of ellipse. This method is efficient especially when the directions of tangents to the trajectory at these points are orthogonal. In this context, Boubaker et al. [6] invented a method in which an elliptic arc is defined by the tangent of the trajectory on their endpoints M_1 and M_2 . Indeed, the calculation of the ellipse parameters takes into account the position of the two endpoints of the arc $M_1(x_1, y_1)$ and $M_2(x_2, y_2)$ as well as the angles of inclination of the tangent. This method gives the minimum value of rebuilding trajectory error compared to quarter of ellipse, arc oblique projection and five points methods. So, an elliptic arc is characterized by four geometric parameters a, b, θ and θ_p where a and b are respectively the half dimensions of the large and the small axes of the elliptic arc, θ is the angle of the ellipse major axe inclination and θ_p is the angle of inclination of the tangents at the stroke endpoint M_2 . In fact, each stroke is modeled by a features vector of 9 parameters as presented in Table 1.

Figure1 demonstrates the application of beta-elliptic approach on Latin, Arabic and digit scripts respectively. The first line contains the original handwritten trajectories, the second line shows the rebuilding trajectories using elliptic arcs and the last line presents the curvilinear velocity signal fitted by overlapped beta impulses.

Table 1 Beta-elliptic parameters

	Parameters	Explanation
Dynamic profile	K	Beta impulse amplitude
	$\delta t = (t_1 - t_0)$	Beta impulse duration
	$Rap = \frac{p}{p+q}$	Rapport of beta impulse asymmetry or culminating time
	P	Beta shape parameters
	$\frac{K_i}{K_{i+1}}$	Rapport of successive Beta impulse amplitude
Geometric profile	a	Ellipse major axis half length
	b	Ellipse small axis half length
	θ	Ellipse major axis inclination angle
	θ_p	Angle of inclination of the tangents at the stroke endpoint M_2

4 Framework for multi-language online handwriting recognition

In this section, we present the different steps for the developed multi-language online handwriting recognition system. In the preprocessing stage, we have applied a Chebyshev second type low pass filter with a cutoff frequency $f_{cut} = 12$ Hz to eliminate the noise generated by spatial and temporal sampling. Furthermore, the vertical dimension of the script lines is adjusted to obtain a normalized script size.

4.1 Pre-classification stage

For the segmentation processes, the handwriting trajectory is divided into continuous pen traces, called pseudo-words, delimited between two successive pen-down and pen-up moments. We obtained a large database of feature vectors X_{ij} containing the beta-elliptic parameters of strokes belonging to a same pseudo-word. Since the developed recognition

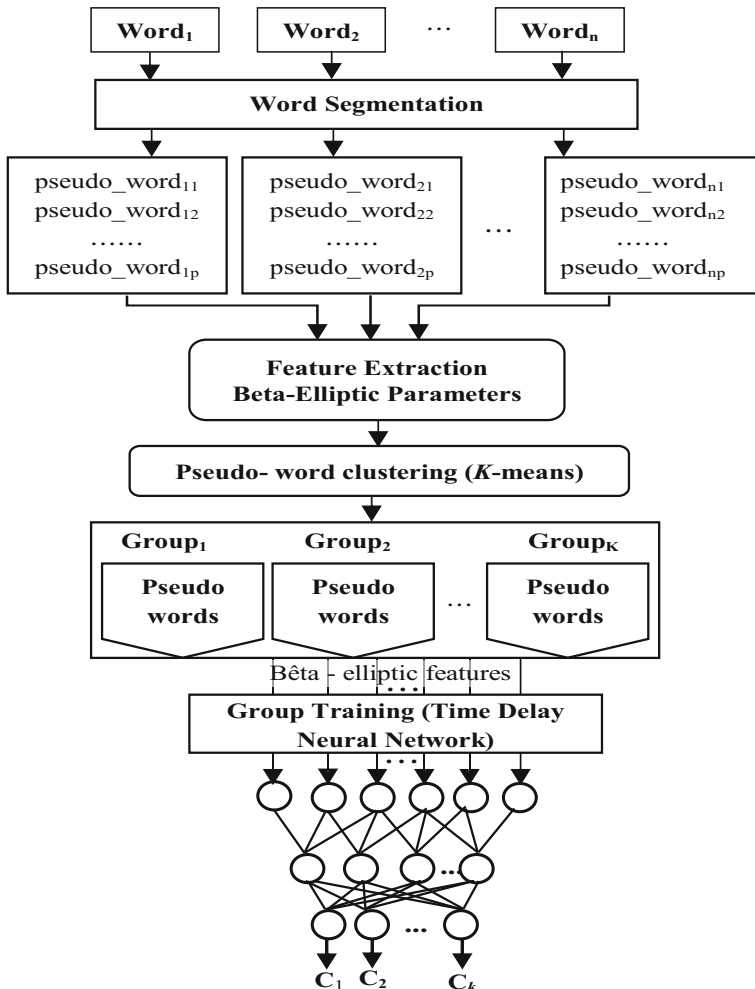


Fig. 2 Script segmentation process

algorithm is addressed for multi-writer application, the pseudo-word trajectory shape and length change from one person to another depending to the handwriting style. For this reason, we cannot manually associate a label for each pseudo-word since their number is unknown in advance. Hence, we have applied the unsupervised clustering algorithm k-means to classify automatically all pseudo-words into k groups. Its principle consists on starting with K groups of a single random point, and thereafter adding each new point to the nearest group. After a point is added, the mean of that group is adjusted. This procedure is iterated until no point is added to the K groups [26]. In our case, the number of groups is defined empirically and we select the value of K that returns the least within-cluster sums of points to centroid distances (Fig. 2).

4.2 Group training with time delay neural network

This step aims to train all pseudo-words into their k groups. Since a pseudo-word is composed of a succession of beta strokes over the time, we have to choose a machine learning algorithm that have the ability to represent relationships between events in time. For this reason, we opted for Time Delay Neural Network (TDNN) which has an efficient architecture able to assimilate the sequential aspect of the input handwritten data [44] and requires less computational time than Hidden Markov Models. TDNN was firstly introduced for phoneme recognition [40]. It is composed of two parts: extraction and classification. The first part contains convolution layers while the second is similar to a multilayer perceptron network. The input data is organized in two dimensions where the horizontal and vertical directions represent respectively the temporal and the characteristic axis. Moreover, each neuron in the convolution layer is connected to a local window from the previous layer, called receptive field. For a given layer, this window is shifted over the time axis with the same delay value (Fig. 3). Thus, the number of neurons $nb_{neurons}$ in the convolution layer i can be obtained through the following formula:

$$nb_{neurons_i} = \frac{nb_{neurons_{i-1}} - receptive\ field}{delay} + 1 \tag{6}$$

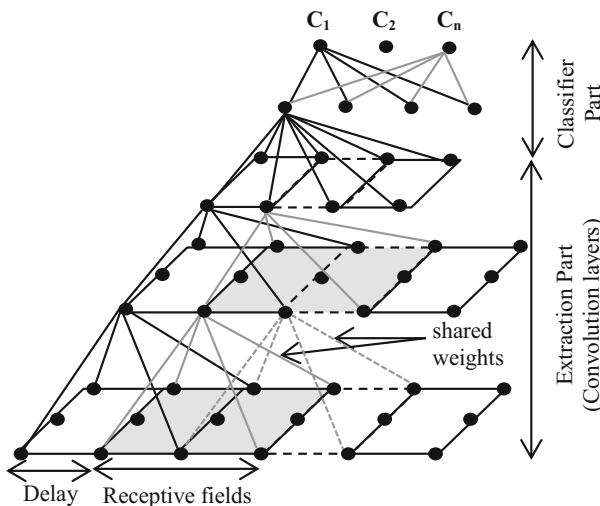


Fig. 3 Time delay neural network architecture

where $nb_{neurons_{i-1}}$ represents the number of neurons in the previous layer, *receptive field* the size of the receptive field in the layer $i - 1$ and *delay* the temporal shift between two consecutive receptive fields. One of the advantages of TDNN is the weight sharing constraint. Indeed, the neurons of different receptive fields shared the same weights. This reduces the number of parameters in the system and facilitates the generalization process even with limited amounts of training data. In the second part, named extraction, each neuron is connected to all the neurons of the previous layer. Since a pseudo-word is composed of several beta strokes, the TDNN input layer contains the the succession of strokes along horizontal direction, while the vertical direction contains the kinematic and geometric features. Since the number of strokes varies from one pseudo-word to another, we have applied the zero-padding technique for the input data to fix the variation of size. It consists on adding new strokes with zero values. In the training process, TDNN was trained using backpropagation algorithm with descent gradient. The number of neurons in the output layer is equal to the desired output classes that represent the k groups obtained from the pre-classification stage. Besides, the values returned by the LogSoftMax layer are used to get a probability for each group i.e. a pseudo-word does not belong only to one group (crisp classification), but to the k groups with a membership probability P_i . Once the training is finished, each pseudo-word will be modeled by a vector of size k representing its degrees of belonging to the k groups.

4.3 Script identification with recurrent neural network

Script identification is the first step in document automation process that attempt to predict the language of a given script in a multilingual environment. In literature, few works exist with regards to the identification of online handwritten scripts. Among these works, Namboodiri et al. [27] proposed a script identification algorithm for recognizing six major scripts. After the preprocessing stage, they extracted spatial features of strokes as direction, density and length. The identification is done using SVM classifier and attained an overall accuracy of 87.1% at the word level. In other work, Tan et al. [37] proposed a system for identifying Arabic, Roman and Tamil scripts. They used the same methods of preprocessing and features extraction described in [27] to obtain groups of prototypes using the k -means clustering algorithm. Thereafter, the distributions frequencies of extracted vectors from the test set are computed to map these vectors to the previous prototypes based on the $tf - idf$ measurement. The last stage consists on comparing the distribution of $tf - idf$ vectors of the test document to that of the three script families. So, the document is assigned to the family with the minimum Chi-square distance. In our framework, we proposed a script identification method based on Recurrent Neural Network who is a powerful architecture for sequential data thanks to its internal states. It has demonstrated great success in sequence labeling and prediction tasks. RNN is trained by stochastic gradient descent using Backpropagation through Time algorithm (BPTT). A major problem with gradient descent for standard RNN architectures (Vanilla RNN) is that error gradients vanish exponentially quickly with long-term dependencies, due to what is called the vanishing/exploding gradient problem. To address this drawback, Hochreiter et al. [15] have designed a Long Short Term Memory (LSTM) network which can in principle store and retrieve information over long time periods. LSTM explicitly designs a memory block inside a hidden node that has the following ingredients: memory cell which stores information about the past and input C_t , output, and forget gates that control the flow of information within and among the memory blocks. Figure 4 shows the structure of a single LSTM memory block, which replaces a simple hidden node used in ordinary RNN. LSTM network

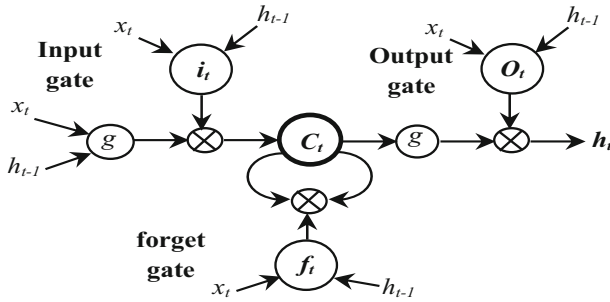


Fig. 4 LSTM cell

recurrently applies the following series of equations to obtain the sequence of hidden node outputs, $h = (h_1, h_2, \dots, h_t), h_t \in R^m$:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{Ci}C_{t-1} + b_i) \tag{7}$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{Cf}C_{t-1} + b_f) \tag{8}$$

$$C_t = f_t C_{t-1} + i_t \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \tag{9}$$

$$O_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{Co}C_{t-1} + b_o) \tag{10}$$

$$h_t = O_t \tanh(C_t) \tag{11}$$

where the symbols i, f, O and C represent respectively stand for the input gate, forget gate, output gate, and memory cell state vector. W denotes weight matrices, the b terms denote bias vectors and σ is the logistic sigmoid function. Note that for the gates, there are not only the recurrent connections from the hidden node outputs from the previous time stamp, but also the peephole connections from the cell states. With explicitly designed cell and gate structures as above, LSTM learns W and b from the training data so that it can determine when to receive input signals to the cell, output the hidden node activations from the memory blocks, and reset the cell states to refresh the memory. In addition to LSTM blocks, another method can be applied in order to prevent RNN from overfitting during training stage, called Dropout. It is a popular regularization technique for the feed-forward neural networks where some network units are randomly masked during training. This method was typically applied only at non-recurrent layers. Recently, it is applied at the recurrent layers, i.e. dropout is applied on the output at step t before it is used to compute the output at step $t+1$ [31].

In our script identification system, RNN input layer contains vectors of pseudo-words belonging to the same word obtained from the trained TDNN (see Section 4.2). Before recurrent layers, these vectors are converted into “one-hot” encoding vectors i.e. vectors of zeros with 1 at a single position that corresponds to the higher probability. Thereafter, dropout technique was applied both in LSTM and fully-connected layers to protect the network against overfitting. In order to allow RNN distinguish between three families of scripts (Latin, Arabic and digits), we added a LogSoftmax layer to classify scripts by selecting the most probably labelling. Figure 5 shows the structure of the proposed RNN. X_{t-2}, X_{t-1} and X_t present the vector of pseudo-words. The circles with crosses denote the randomly omitted nodes during training stage and the dotted arrows stand for the model weights connected to those omitted nodes.

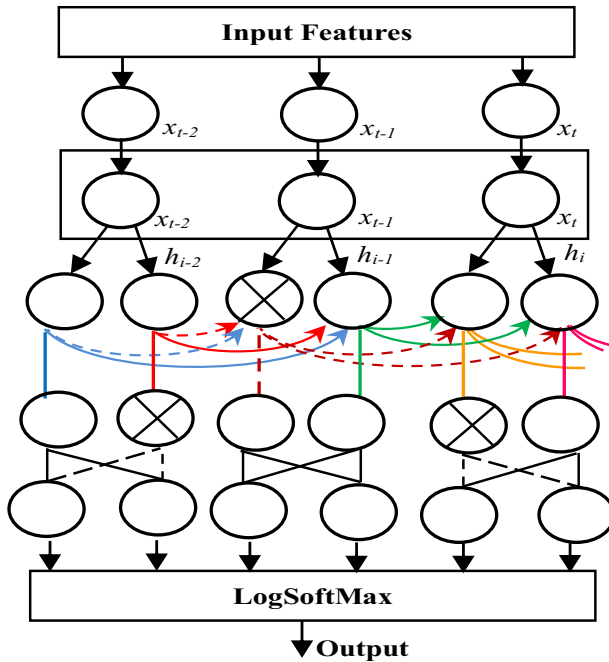


Fig. 5 RNN architecture

4.4 Hybrid TDNN-SVM for script recognition

Once script identification stage is achieved, the next step consists on script recognition. It is based on the association of TDNN with SVM classifier. Indeed, SVM is a power classifier developed by Vapnik in 1995 [38], that performs classification tasks. It maps the input points into a high-dimensional feature space and finds a separating hyperplane that maximizes the margin between two classes by using the dot product functions in feature space called kernels, to know linear, sigmoid, Radial Basis Function (RBF) and polynomial (Fig. 6). Among these kernels, RBF perform better in nonlinear separation case which is defined as follow:

$$K(x, x') = \phi(x)\phi(x') = \exp(-\gamma||x - x'||^2) \tag{12}$$

where γ is a parameter will be defined empirically, x and x' represent two input vectors, ϕ a nonlinear transform and $\phi(x)$ is the transformed features space. Hence we can define, the optimal hyper-plane (H_0) through the following formula:

$$f(x) = \text{sign} \left(\sum y_i \alpha_i K(x_i, x'_i) + b \right) \tag{13}$$

where $\text{sgn}(\cdot)$ is the sign function.

SVM was originally used for binary classification tasks (only 2 classes). However, when the numbers of classes are more than two, binary classification algorithms can be turned into multi-class classification algorithms by a variety of strategies, to know one-versus-all and one-versus-one methods [8]. In our case, we opted for one-versus-all method since the proposed system treats the recognition of several classes. In our hybrid recognizer engine, each word is composed of pseudo-words that belong to the k predefined groups according

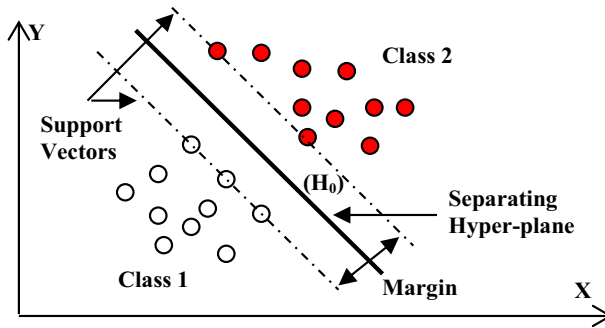


Fig. 6 SVM class discrimination hyperlane

to their beta-elliptic parameters (see Section 4.1). These groups are trained using TDNN that output for each pseudo-word a vector of size k containing the membership probabilities to the k groups (see Section 4.2). For the recognition process, the pseudo-words vectors belonging to the same word are gathered to form the input layer of SVM which allows establishing a relation between the fuzzy outputs of TDNN and the desired output. We noticed that the recognition process is applied on the database selected in the script identification stage (Fig. 7).

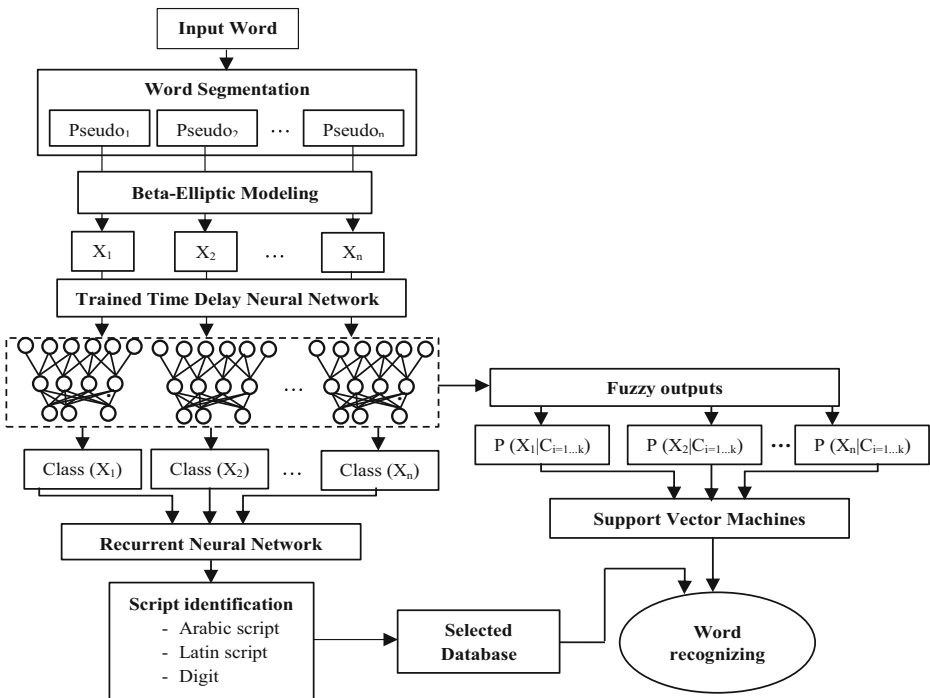


Fig. 7 Multi-language online handwriting recognition architecture

Table 2 ADAB dataset description

	Number of words	Number of pseudo words	Writers
Set1	5037	40296	56
Set2	5090	25450	37
Set3	5031	15093	39
Set4	4417	22085	25
Set5	1000	4000	6
Set6	1000	8000	3
Total	21575	114924	166

5 Experiments and results

5.1 Experimental setup

In experimentation phase, we have used three families of databases representing Arabic, Latin and digit scripts. For Arabic script, we have used ADAB database that is known as a standard benchmark in ICDAR2009 and it is applied in several handwriting applications. ADAB is divided into 6 subsets with a total of 21575 handwritten words produced by 166 writers and contains the names of 937 Tunisian towns (Table 2). One of the advantages of this database is the possibility of recovering both online and offline signals of the same handwriting. In our work, we are only interested in the online signal part. For each word, the sequence of (x, y) coordinates are stored in “UPX” file format that contains some additional information’s about the writer and the word label [22]. For Latin script, the Unipen-ICROW-03 benchmark dataset have been used. It is a multi-writer database composed of 13119 words belonging to 884 classes (Table 3). However, the pen trajectory is encoded as a sequence of segments according to the UNIPEN format [14]. The third database is called Pendigit [3]. It contains a total of 10992 digits collected from 44 writers. The samples are divided into two sets: the digits written by the first 30 writers are used for training, cross-validation and writer dependent testing, and the digits written by the other 14 are used for writer independent testing. For each digit (0-9), the pen trajectory is recorded in the UNIPEN format (Table 4). Figure 8 shows some samples from the described datasets.

To prepare data for the pre-classification stage, scripts from databases are divided into pseudo-words delimited between two successive pen-down and pen-up moments. The obtained 185362 pseudo-words are classified into k groups according to their beta-elliptic parameters (Table 5) by applying the k -means clustering algorithm. The value of $k=215$ is retained empirically as the number of groups that return the least within-cluster sums of points to centroid distances. For each group, two thirds of pseudo-word vectors have been used for the training phase and the rest has been used for the tests. These groups are trained using TDNN. The configuration of this network consists on setting the number of convolution layers with the size of receptive fields and delays. The number of neurons in the output

Table 3 ICROW-03 dataset description

	Number of words	Number of pseudo words	Writers	Lexical words
13119	56841	72	884	

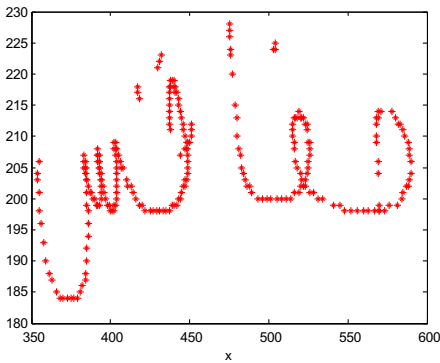
Table 4 Pendigit dataset description

Number of digits	Number of pseudos	Writers	Lexical digit
10992	13597	44	10 (0-9)

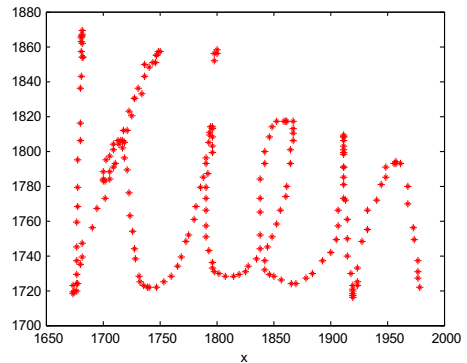
layer is equal to the number of groups. The best result has been obtained with a number of 3 convolution layers and attained a classification rate of 97.32% (Table 6).

5.2 Results and discussions

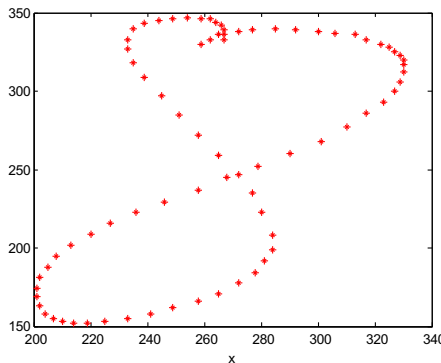
The experiments have been made on a multilingual environment where the first stage consists on script identification. Thus, the sequence of trained pseudo-words vectors belonging to the same word is gathered to form the RNN input layer. We have implemented one recurrent layer with LSTM units and two fully-connected layers. The number of neurons for each layer is respectively 128, 128 and 64 units. RNN is trained by stochastic gradient descent with a fixed learning rate of 10^{-3} . In the training stage, dropout is applied for both LSTM



(a) ADAB dataset sample



(b) Unipen dataset sample



(c) Pendigit dataset sample

Fig. 8 Multilingual database samples

Table 5 Database description

	Number of words	Number of pseudos
Training set	33593	130870
Test set	12093	54492
Total	45686	185362

and fully-connected layers with a probability $p = 0.3$. Experiments results demonstrate the effectiveness of the proposed architecture that reached the identification rate of 100% when applying the dropout technique (Tables 7 and 8).

For the recognition stage, experiments have been made on the database selected in the script identification stage. The concept is to gather the pseudo-words vectors of the same word in order to recognize the word label. Contrary to the previous stage, these vectors present the outputs of the LogSoftMax layer of TDNN. They contain the membership's probabilities to the k groups. Experiments were carried on SVM with different kernel functions. The best result has been obtained with Radial Basis Function kernel (RBF) and has reached the impressive recognition rate of 99.89% (Table 9). To evaluate our proposed system, we must compare it with already exists systems. We have difficulties to do it because few works focuses on multilingual online handwriting recognition and there are no benchmark multilingual database commonly used. Table 10 presents a comparison with some existing systems. Besides results obtained by our system, some originalities have been presented. Firstly, we have used a free segmentation method based on successive pen-down and pen-up movements and beta-elliptic approach have been applied for a variety of handwriting styles, who is more efficient than works cited in [18] and [17] turned on Latin, Arabic and digit scripts and where the segmentation module is based on parametric rules and heuristic thresholds which makes it sensitive to the variation of writing style. Secondly, the multilingual script recognition is preceded by a script identification step, that is to our knowledge, the second system using this strategy after the work of Lee et al. [24]. Despite the huge data number, our system performs well and quickly. This is due to the group training stage where the beta-elliptic parameters of pseudo-words are converted into vectors of membership probabilities. Furthermore, the dropout technique used for RNN and the propriety of shared parameters for TDNN, allow reducing significantly the computational time. Finally, the proposed system is among the few works that have used the hybridization of several machine learning methods as RNN, TDNN and SVM.

Table 6 Group training results

Convolution layers	Hidden layers	Delay	Receptive field	Classification rate
2	1	1	2	89.97%
2	2	1	3	93.14%
3	1	1	2	97.32%

Table 7 Script identification results

Epoch	Co-occurrence Matrix			Rate (%)
1	1856	927	5076	45,5%
	280	314	2142	
	1	0	3497	
2	1210	4414	2235	53.8%
	365	1030	1341	
	0	0	3498	
3	12	7685	162	47.4%
	26	1547	1163	
	0	0	3498	
4	161	7676	22	59.3%
	0	2731	5	
	0	0	3498	
5	7610	249	0	98.6%
	0	2736	0	
	1	0	3497	

Table 8 Dropout improvement

Epoch	No dropout	Dropout
5	98.6%	100%

Table 9 Experiments results

	Kernel function			
	Linear	Polynomial	Sigmoid	RBF
Recognition Rate(%)	96.52	96,92	97,12	99.89

Table 10 Results comparison

Systems	Approach	Scripts	Recognition rate
[24]	Hierarchical HMM	Korean, Latin	82,11%
[18]	Lattice Decoding	Latin, Arabic, Indian	95.70%
[17]	Geometric features	Urdu (similar to Arabic), digits	94%
Present system	Hybrid TDNN-SVM	Latin, Arabic, Digits	99.89%

6 Conclusions and future works

We presented a new framework for multilingual online handwriting recognition. It proceeds by segmenting the script into pseudo-words representing the interval between two successive pen-down and pen-up moments. In this work, we have explored the potential utility of beta-elliptic approach for online handwritten modeling. It allows extracting the dynamics and geometrics profiles of the trajectories. In this study, several machine learning methods have been used. For the pseudo-word groups training, TDNN has been chosen thanks to its ability to deal with sequential data. Moreover, its proprieties of receptive fields and shared weights allow reducing the number of parameters and thus reducing the computational time. Unlike previous studies, a step of script identification has been proposed in order to identify the script language. It based on Recurrent Neural Network with long short term memory to protect the network against the vanishing gradient problem. The input features are the outputs vectors of the trained TDNN and they are converted into “one-hot” encoding vectors. In the recognition stage, hybridization between TDNN and SVM has been established. Thus, the fuzzy vectors from LogsoftMax layer of TDNN and belonging to the same word are gathered and entered to the SVM in order to recognize the handwriting script. Experimental results obtained for script identification and handwriting recognition prove the efficiency of techniques used for handwriting modeling and recognition engine that lead to quite high recognition rates. However, our study is to be continued. We plan to work on other languages. It is worth noticing that beta-elliptic approach is rather generic and has proved its applicability for several handwriting styles. It will be interests point to apply it on other scripts as Chinese, Indic etc. We also intend expand this system on a multi-content pattern to be able to work in unlimited vocabulary.

Acknowledgements This work was supported by grants from the General Direction of Scientific Research and Technological Renovation (DGRST), Tunisia, under the ARUB program 01/UR/11/02.

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

1. Ahmed H, Azeem SA (2011) On-line Arabic handwriting recognition system based on HMM. In: International conference on document analysis and recognition (ICDAR). IEEE, pp 1324–1328
2. Alimi AM (2003) Beta neuro-fuzzy systems. *Task Q* 7(1):23–41
3. Alimolu F, Alpaydin E (2001) Combining multiple representations for pen-based handwritten digit recognition. *Turkish J Electr Eng Comput Sci* 9(1):1–12
4. Bhattacharya S, Maitra DS, Bhattacharya U, Parui SK (2016) An end-to-end system for Bangla online handwriting recognition. In: 15th international conference on frontiers in handwriting recognition (ICFHR). IEEE, pp 373–378
5. Boubaker H, Elbaati A, Tagougui N, El Abed H, Kherallah M, Alimi AM (2012) Online Arabic databases and applications. In: *Guide to OCR for Arabic scripts*. Springer, London, pp 541–557
6. Boubaker H, Rezzoug N, Kherallah M, Gorce P, Alimi AM (2015) Spatiotemporal representation of 3D hand trajectory based on Beta-elliptic models. *Comput Methods Biomech Biomed Eng J* 18(15):1632–1647
7. Bezine H, Alimi AM, Sherkat N (2004) Generation and analysis of handwriting script with the beta-elliptic model. In: *Proceedings of the 9th international workshop on frontiers in handwriting recognition*, pp 515–520
8. Burges CJ (1998) A tutorial on support vector machines for pattern recognition. *Data Min Knowl Disc* 2(2):121–167

9. Elleuch M, Zouari R, Kherallah M (2016) Feature extractor based deep method to enhance online arabic handwritten recognition system. In: 25th international conference on artificial neural networks, pp 136–144
10. Gargouri M, Kanoun S, Ogier JM (2013) Text-independent writer identification on online Arabic handwriting. In: 12th international conference on document analysis and recognition (ICDAR). IEEE, pp 428–432
11. Graves A, Liwicki M, Fernández S, Bunke H, Bertolami R, Schmidhuber J (2009) A novel connectionist system for unconstrained handwriting recognition. *IEEE Trans Pattern Anal Mach Intell* 31(5):855–868
12. Ghosh R, Roy PP (2015) Study of two zone-based features for online Bengali and Devanagari character recognition. In: 13th international conference on document analysis and recognition (ICDAR). IEEE, pp 401–405
13. Ghosh R, Roy PP (2016) Comparison of zone-features for online Bengali and Devanagari word recognition using HMM. In: 15th international conference on frontiers in handwriting recognition (ICFHR), IEEE, pp 435–440
14. Guyon I, Schomaker L, Plamondon R, Liberman M, Janet S (1994) UNIPEN project of on-line data exchange and recognizer benchmarks. In: *Pattern recognition*, vol 2, pp 29–33
15. Hochreiter S, Schmidhuber J (1997) Long short-term memory. *Neural Comput* 9(8):1735–1780
16. Hollerbach J (1981) An oscillation theory of handwriting. *Biol Cybern* 39(29, Issue 2):139–156
17. Indhu TR, Vidya V, Bhadrans VK (2015) Multilingual online handwriting recognition system: an android app. In: The fifth international conference on advances in computing and communications (ICACC), IEEE, pp 33–36
18. Keysers D, Deselaers T, Rowley HA, Wang LL, Carbone V (2017) Multi-language online handwriting recognition. *IEEE Trans Pattern Anal Mach Intell* 39(6):1180–1194
19. Kherallah M, Njah S, Alimi AM, Derbel N (2002) Recognition of on-line handwritten digits by neural networks using circular and beta approaches. In: *International conference on system: man and cybernetics*, IEEE, pp 164–169
20. Kherallah M, Haddad L, Alimi AM, Mitiche A (2008) On-line handwritten digit recognition based on trajectory and velocity modeling. *Pattern Recog Lett* 29(5):580–594
21. Kherallah M, Bouri F, Alimi AM (2009) On-line Arabic handwriting recognition system based on visual encoding and genetic algorithm. *Eng Appl Artif Intell* 22(1):153–170
22. Kherallah M, Tagougui N, Alimi AM, El Abed H, Margner V (2011) Online Arabic handwriting recognition competition. In: *International conference on document analysis and recognition (ICDAR)*. IEEE, pp 1454–1458
23. Khelif H, Prum S, Kessentini Y, Kanoun S, Ogier J (2016) Fusion of explicit segmentation based system and segmentation-free based system for on-line Arabic handwritten word recognition. In: 15th international conference on frontiers in handwriting recognition (ICFHR). IEEE, pp 399–404
24. Lee J, Kim JH (1996) A unified network-based approach for online recognition of multi-lingual cursive handwritings. In: *Proceeding of fifth international workshop on frontiers in handwriting recognition*, pp 393–397
25. Maalej R, Tagougui N, Kherallah M (2016) Online Arabic handwriting recognition with dropout applied in deep recurrent neural networks. In: 12th IAPR workshop on document analysis systems (DAS). IEEE, pp 417–421
26. MacQueen J (1967) Some methods for classification and analysis of multivariate observations. In: *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol 1, no 14, pp 281–297
27. Nambodiri AM, Jain AK (2004) Online handwritten script recognition. *IEEE Trans Pattern Anal Mach Intell* 26(1):124–130
28. Nakkach H, Haboubi S, Amiri H (2016) Online Arabic character recognition using global and local features. In: 3rd international conference on automation control engineering and computer science, pp 120–124
29. Nguyen T, Zhu B, Nakagawa M (2014) A semi-incremental recognition method for on-line handwritten english text. In: 14th international conference on frontiers in handwriting recognition (ICFHR). IEEE, pp 234–239
30. O'Reilly C, Plamondon R (2009) Development of a Sigma–Lognormal representation for on-line signatures. *Pattern Recog* 42(12):3324–3337
31. Pham V, Bluche T, Kermorvant C, Louradour J (2014) Dropout improves recurrent neural networks for handwriting recognition. In: 14th international conference on frontiers in handwriting recognition (ICFHR). IEEE, pp 285–290

32. Plamondon R, Alimi AM, Yergeau P, Franck L (1993) Modeling velocity profiles of rapid movements: a comparative study. *Biol Cybern* 69(2):119–128
33. Plamondon R (1995) A kinematic theory of rapid human movements. Part-II. Movement time and control. *Biol Cybern* 72:309–320
34. Samanta O, Anandarup R, Bhattacharya U, Ujjwal SK (2015) Script independent online handwriting recognition. In: 13th international conference on document analysis and recognition (ICDAR). IEEE, pp 1251–1255
35. Sun L, Tonghua S, Wang R (2016) Deep LSTM networks for online chinese handwriting recognition. In: 15th international conference on frontiers in handwriting recognition (ICFHR). IEEE, pp 271–276
36. Tagougui N, Boubaker H, Kherallah M, Alimi AM (2013) A hybrid MLPNN/HMM recognition system for online Arabic handwritten script. In: World congress on computer and information technology (WCCIT). IEEE, pp 1–6
37. Tan GX, Viard-Gaudin C, Kot AC (2009) Information retrieval model for online handwritten script identification. In: 10th international conference on document analysis and recognition (ICDAR). IEEE, pp 336–340
38. Vapnik V (1998) *Statistical learning theory*, vol 1, 1735,1780
39. Viviani P, Terzuolo C (1982) Trajectory determines movement dynamics. *Neuroscience* 7(2):431–437
40. Waibel A, Hanazawa T, Hinton G, Shikano K, Lang KJ (1989) Phoneme recognition using time-delay neural networks. *IEEE Trans Acoust Speech Sig Process* 37(3):328–339
41. Yang W, Jin L, Xie Z, Feng Z (2015) Improved deep convolutional neural network for online handwritten Chinese character recognition using domain-specific knowledge. In: 13th international conference on document analysis and recognition (ICDAR). IEEE, pp 551–555
42. Zhou X, Zhang Y, Tian F, Wang H, Liu C (2014) Minimum-risk training for semi-Markov conditional random fields with application to handwritten Chinese/Japanese text recognition. *Pattern Recog* 47(5):1904–1916
43. Zhu B, Shivram A, Govindaraju V, Nakagawa M (2016) Online handwritten cursive word recognition by combining segmentation-free and segmentation-based methods. In: 15th international conference on frontiers in handwriting recognition (ICFHR). IEEE, pp 417–422
44. Zouari R, Boubaker H, Kherallah M (2016) A time delay neural network for online arabic handwriting recognition. In: International conference on intelligent systems design and applications. Springer, pp 1005–1014



Ramzi Zouari is a Ph.D. student at the National School of Engineering of Sfax, Tunisia. He obtained the master degree in computer science from the University of Sfax in 2012. His main research interest is in the area of online Handwriting recognition and handwriting trajectory modeling.



Houcine Boubaker Graduated from Electrical Engineering Department in 1995. He obtained his master degree in systems analyses and digital signal processing in 1997. He is a researcher and a PhD student in Electrical & Computer Engineering Department at the University of Sfax. His research interest includes trajectory modeling and applications of intelligent methods to pattern recognition. He is an IEEE student member.



Monji Kherallah Received the Ing. Diploma degree and the PhD in electrical engineering, in 1989 and 2008, respectively from University of Sfax (ENIS), Tunisia. He is a member in Research Group of Intelligent Machines: REGIM. His research interest includes the following projects: “i-Bag”, “i-House” and “i-Car”. The techniques used are based on methods intelligent, such as neural network, logic fuzzy, genetic algorithm, etc. He is a reviewer of several international journals.