

# Hybrid Architecture based on RNN-SVM for Multilingual Online Handwriting Recognition using Beta-elliptic and CNN models

This paper was downloaded from TechRxiv (<https://www.techrxiv.org>).

LICENSE

CC BY 4.0

SUBMISSION DATE / POSTED DATE

11-02-2021 / 24-05-2021

CITATION

Hamdi, Yahia; Boubaker, Houcine; Rabhi, Besma; Ouarda, Wael; Alimi, Adel (2021): Hybrid Architecture based on RNN-SVM for Multilingual Online Handwriting Recognition using Beta-elliptic and CNN models. TechRxiv. Preprint. <https://doi.org/10.36227/techrxiv.13903661.v3>

DOI

[10.36227/techrxiv.13903661.v3](https://doi.org/10.36227/techrxiv.13903661.v3)

# Hybrid Architecture based on RNN-SVM for Multilingual Online Handwriting Recognition using Beta-elliptic and CNN models

Yahia Hamdi<sup>a</sup>, Houcine Boubaker<sup>a</sup>, Besma Rabhi<sup>a</sup>, Wael Ouarda<sup>a</sup>,  
Bidyut B. Chaudhuri<sup>b,c</sup>, Adel M. Alimi<sup>a,d</sup>

<sup>a</sup>*University of Sfax, National Engineering School of Sfax, REGIM-Lab.: REsearch Groups in Intelligent Machines, LR11ES48, 3038, Sfax, Tunisia.*

<sup>b</sup>*Techno India University, Kolkata-700091, India.*

<sup>c</sup>*CVPR Unit, Indian Statistical Institute, Kolkata-700108, India.*

<sup>d</sup>*Department of Electrical and Electronic Engineering Science, Faculty of Engineering and the Built Environment, University of Johannesburg, South Africa.*

---

## Abstract

Currently, deep learning-based approaches have proven successful in the handwriting recognition field. Despite this, research in this area is still required, particularly in multilingual online handwriting recognition by exploiting new network architectures and combining relevant parametric models. In this paper, we propose a multi-stage deep learning-based algorithm for multilingual online handwriting recognition based on hybrid deep Bidirectional Long Short Term Memory (DBLSTM) and SVM networks. The main contributions of our work lie partly in the composition of a new multi-stage architecture of deep learning networks associated with effective feature vectors that integrate dynamic and visual characteristics. First, the proposed system proceeds by pre-treating the acquired script and delimiting its Segments of Online Handwriting Trajectories (SOHTs). Second, two types of feature vectors combining Beta-Elliptic Model (BEM) and Convolutional Neural Network (CNN) are extracted for each SOHT. Then, the SOHT feature vectors are fuzzily classified using DBLSTM

---

\*Corresponding author

*Email addresses:* [yahia.hamdi.tn@ieee.org](mailto:yahia.hamdi.tn@ieee.org) (Yahia Hamdi),  
[houcine-boubaker@ieee.org](mailto:houcine-boubaker@ieee.org) (Houcine Boubaker), [besma.rebhi.2015@ieee.org](mailto:besma.rebhi.2015@ieee.org) (Besma Rabhi), [wael.ouarda@ieee.org](mailto:wael.ouarda@ieee.org) (Wael Ouarda), [bbcisical@gmail.com](mailto:bbcisical@gmail.com) (Bidyut B. Chaudhuri), [adel.alimi@ieee.org](mailto:adel.alimi@ieee.org) (Adel M. Alimi)

neural networks for both online and offline branches. This classification is based on  $K$  sub-groups determined using an unsupervised fuzzy k-means algorithm. Finally, we combine the trained models to strengthen the discrimination power of the global system using SVM engine. Extensive experiments on three data sets were conducted to validate the performance of the proposed method. The experimental results show the effectiveness of the individual modules as well as the advantage of their fusion.

*Keywords:* Online Handwriting Recognition, Multilingual context, Beta-Elliptic Model, CNN, DBLSTM, SVM

---

## 1. Introduction

Automatic handwriting recognition has regained more importance with the diffusion of intelligent mobile devices such as PDAs, tablet-PC, and smartphones equipped with sensitive pen or touch screens. These devices allow us to easily record online handwriting input in so-called digital *ink* format. This format describes temporal and spatial information on the sequence of points  $M_i(x_i, y_i, t_i)$  sampling the pen or touch trajectory [1] (where  $x_i, y_i$  denote position coordinates at  $t_i$ -th instant). The high popularity of these hand-held devices and their smart way to input data, coupled with recent major progress in machine learning especially the deep learning techniques in speech recognition, has invited researchers to adopt and develop deep learning approaches to build more efficient online handwriting recognition systems.

The writing style variability coupled with the incorporation of multi-language words in the same online handwriting script pose some challenges for its recognition task. In fact, besides shape variation and opposed direction of script evolution such as Arabic and Latin, there are more character category depending on their positions in the word especially, for Arabic script. This makes it difficult to directly adapt the existing recognition systems for such tasks and invites to adopt new architecture based on the combination of deep learning and relevant complementary model.

One of the most relevant criteria to succeed in an online recognition system is the choice of a powerful handwriting model. A wide number of hand-crafted features such as structural, parametric, and global features are described in the literature to overcome specific problems like occlusions, ligatures, and scale variations, etc. Moreover, many other properties are reported with human handwriting movements such as speed, amplitude and script size which can be modified involuntarily without alternating the global shape of velocity profile modeling the handwriting trajectory [2]. The hybrid kinematics and geometric aspect of online handwriting generation movements are presented by the Beta-elliptic model (BEM) [3] which has attained immense success in many fields such as handwriting regeneration [4], writer identification [5], [6] and temporal order recovery [7]. The handwritten trajectory is normally segmented into  $n$  simple movements limited by curvature or velocity extrema called *strokes*. In our work, we are interested to exploit the great benefit of this BEM in online handwriting trajectory modeling.

Lately, deep learning technologies [8] have attracted immense research works and industrial applications. These new categories of methods provide a powerful representation for end-to-end solutions. For instance, CNN is a prominent deep learning method which automatically extracts useful features from low to high level. It has been widely used in pattern recognition field with excellent results. Also, Recurrent neural networks (RNNs) have been employed successfully in online handwriting recognition. It is very effective with LSTM for handwriting sequence generation [9] and produces top results in online handwriting recognition [10]. Motivated by these ideas and noting the great diversity of writing styles, we are interested in applying a CNN and RNN-based model in our problem. In this paper, we present a multi-stage system that is used to recognize online handwritten multi-script text based on DBLSTM and SVM algorithm. The main idea is to combine two different approaches for handwritten trajectory representation: the enhanced BEM which is described by a combination of two aspects of online data modeling such as the geometric and dynamic profiles, and the CNN as an efficient feature extraction model applied after transforming

the online trajectory into image-like representations. We are interested in overcoming the limits of the online module by allowing the recognition system to apprehend the different writing styles that differ in the timeline of the *ink* plot  
55 but are similar in the final layout. The feature set we employed is described in section 6. Indeed, we compare the performance of single classifiers and their combination with the aim of enhancing the global system discrimination using three available datasets described later.

The remainder of this paper is organized as follows: In section 2, we present  
60 an outline of the most prominent related works. Section 3 provides a brief description of the script on which we mainly worked. An overview of our proposed recognition process is described in section 4. Section 5 summarizes the pre-processing techniques that are used in our work. We describe in section 6 the proposed strategy for handwritten trajectory modeling and pre-classification  
65 step of SOHT. The recognition process that includes SOHT fuzzy classification, regularization, data augmentation and script recognition are presented in section 7. Finally, section 8 describes the evaluation of the proposed system using three databases and discusses the obtained results before ending with a conclusion and future scope of work.

## 70 **2. Related work**

Many studies have been done on online handwriting recognition [1], [2]. Besides, it remains an active area of research because practical applications are relatively recent, and its technology has been applied in many fields such as artificial intelligence, computer engineering, image analysis, etc. In the literature,  
75 studies dealing with online handwriting recognition can be classified into two main categories: the conventional or traditional approaches and deep learning-based approaches.

### *2.1. The conventional approaches*

Generally, conventional approaches require an enormous effort of engineer's  
80 expertise to design a representative features extractor. Various studies have been

made in this context such as decomposition of the online signal into elementary stroke based on Beta-Elliptic model, characteristic strokes [3], or grapheme unit like presented in [11] which is based on baseline detection algorithm. Also, tangent differences and histograms of tangents features are used to represent online  
85 Arabic characters [12]. A framework based on statistical features of online Arabic character recognition are investigated in [13]. Within the context, some studies implement algorithms of the conventional pattern classification which include decision trees [14], template matching [15], Hidden Markov model (HMM) [16], neural networks [17], and support vector machines (SVM) [18]. More recently,  
90 Zitouni et al. [19] have proposed a two-stage SVM classifier for online Arabic script recognition which is based on a combination of beta-elliptic and fuzzy perceptual code representation. Also, a reinforcement learning based-approach for online handwriting recognition is proposed by [20]. It consists of extracting structural features using Freeman chain codes and elementary perceptual codes  
95 (EPCs), parametric features employing beta stroke theory after segmenting the handwriting trajectory into strokes.

## *2.2. Deep learning-based approaches*

Deep learning models extract features automatically from the raw sequential data. Recently, the performance of online handwriting recognition systems  
100 has been significantly enhanced thanks to the employment of models such as convolutional neural networks (e.g., [18]), [21]), Deep Belief Network (DBN) (e.g., [22]) which have been broadly applied for handwriting recognition. It has considered as a powerful tool for image classification. Some research work use CNNs on the online characters after converting the handwriting trajectory to  
105 image-like representations. Also, Yuan et al. [23] proposed word-level recognition for Latin script by employing CNN architecture. In this article, CNN is used for recognizing online handwritten document after word segmentation. Likewise, CNNs have been widely applied for handwritten Chinese text recognition [24]. It is considered as a powerful model that automatically generates  
110 multiple lower-level and high-level features.

Furthermore, RNN has been used successfully for online handwriting recognition due to their ability to model sequential data. It gives good results with Long Short-term Memory (LSTM) for online Arabic character recognition based on a rough path signature [25]. Likewise, Ghosh et al. [26] have performed  
115 a word recognition for both cursive and non-cursive Devanagari and Bengali scripts based on LSTM and BLSTM recurrent networks. In this article, each word is partitioned into three horizontal zones (upper, middle, lower). Then, the middle zone is re-segmented into basic strokes in each directional and structural feature before carrying out training the LSTM and BLSTM. To improve the  
120 performance of deep BLSTM in online Arabic handwriting recognition, Maalej et al. [27] employed three techniques such as dropout, Maxout, and ReLU activation function. These techniques are used in a different position on BLSTM layers and provide a good result. The dropout technique is used to solve the overfitting problem. The second technique is based on rectified linear units  
125 ReLU which is localized between the LSTM hidden layers in order to reduce the label error rate. For the third technique, the Maxout units are integrated inside the LSTM nodes, and also stacked on a separate layer after the BLSTM layers.

Combinations of these techniques have also been investigated. An instance  
130 of this is CNN with deep BLSTM (e.g., [28]) in which CNN allows automatically generating features from the scanned ink sequence, while BLSTM has modeled frame dependency within the sequence. RNN with BLSTM (e.g., [29]) and BLSTM with gated recurrent unit (GRU) have been used for online Chinese character recognition and generation [30].

135 Further, several researches focus specifically on mono-language script, for instance, Arabic [12], Chinese [7], Tamil [31], Japanese [26] etc. Similarly, there exist multi-script and multi-language systems for online handwriting recognition like those presented in [32], [33] as well as other commercial systems such as those developed by Apple [34] and Microsoft [29] organizations.

140 **3. Language specification**

In this section, we briefly provide an overview of Arabic and Latin scripts supported by our system, considering the writing diversity of these scripts compared to others.

The *Arabic* script is used in multiple languages such as Persian, Pashto, 145 Kurdish, and Urdu [13]. More than 420 million people of the world use the Arabic as a major language and script (UNESCO 2012) [35]. Arabic script is generally written in a cursive manner, most of these letters are connected to their neighboring characters. It generally contains 28 basic letters with other 10 marks that can change the meaning of the word. Also, there are 10 digits for 150 usual digit recognition tasks. While there are 26 alphabetic letters for English, there are over 28 in Arabic.

Table 1: Some Arabic characters in different forms

Characters	End	Isolated	Middle	Beginning
Alif (أ)	ا	أ		أ
Baa (ب)	ب	ب	ب	ب
Taa (ت)	ت	ت	ت	ت
Thaa (ث)	ث	ث	ث	ث
Ain (ع)	ع	ع	ع	ع
Raa (ر)	ر	ر		ر
Seen (س)	س	س	س	س

In running text, Arabic characters are written from right to left, and also contains small marks and dots. Moreover, the letters have multiple shapes that depend on their position in the word. We can distinguish four different forms 155 (Beginning, Middle, Isolated and End form) of the Arabic letters according to their position within a word, as shown in Table. 1. However, some characters have the same beginning and isolated shapes (eg., Alif 'أ', Raa 'ر'). Also, several letters have the same body (e.g., Baa 'ب', Thaa 'ت') but differ only in position

and the number of dots and diacritical marks which may be accidentally mis-  
160 placed in handwriting which emphasizes the fact that Arabic character is more  
difficult than many other scripts. Further detail of Arabic characteristics and  
difficulties of writing are available in [36].

The Latin script serves as the most widely adopted writing system in the  
world. It is used as the standard set of writing glyphs for 59 languages. De-  
165 pending on the writing style, Latin script can be written cursive or semi-cursive  
and the characters' shapes vary accordingly. In addition to the basic  $2 * 26$   
characters (capital and minuscule shapes), it also supports accented characters.  
Unlike Arabic script, the Latin script is written from left-to-right.

#### 4. System overview

170 In this section, we introduce an outline of our proposed system of develop-  
ing a multi-stage architecture based on a hybrid of DBLSTM recurrent neural  
networks and SVM for multilingual online handwriting recognition. Indeed, the  
main purpose of our work consists of how to adapt and exploit the effective-  
ness of BEM and CNN models in online multilingual handwriting recognition.  
175 As shown in Fig. 1, the architecture of the proposed system proceeds as fol-  
lows: First, the input signal  $(x, y)$  is normalized and denoised by a preprocess-  
ing technique. Second, the handwriting trajectory is divided into continuous  
components called SOHTs (Segment of Online Handwriting Trajectory) each  
of which represents trajectory segments limited between two successive points:  
180 starting points (pen-down) and ending points (pen-up). Third, we extracted  
two types of feature vectors for each SOHT: the online hand-drawn trajectory  
parameters are extracted using BEM, and the generated generic features by  
using the last CNN layer after transforming the handwriting cursive trace to  
bitmap image. After that, SOHTs are fuzzy classified into  $k$  sub-class defined  
185 by k-means unsupervised algorithm in the training phase. The SOHT's fuzzy  
classification module uses as input the extracted BEM and CNN feature vec-  
tors which train two DBLSTMs networks integrated with the online and offline

branches, respectively. Finally, the description of the fuzzy output obtained by the two DBLSTMs are combined using SVM to improve the discriminating power of the global system. In the following sections, we introduce each module in detail.

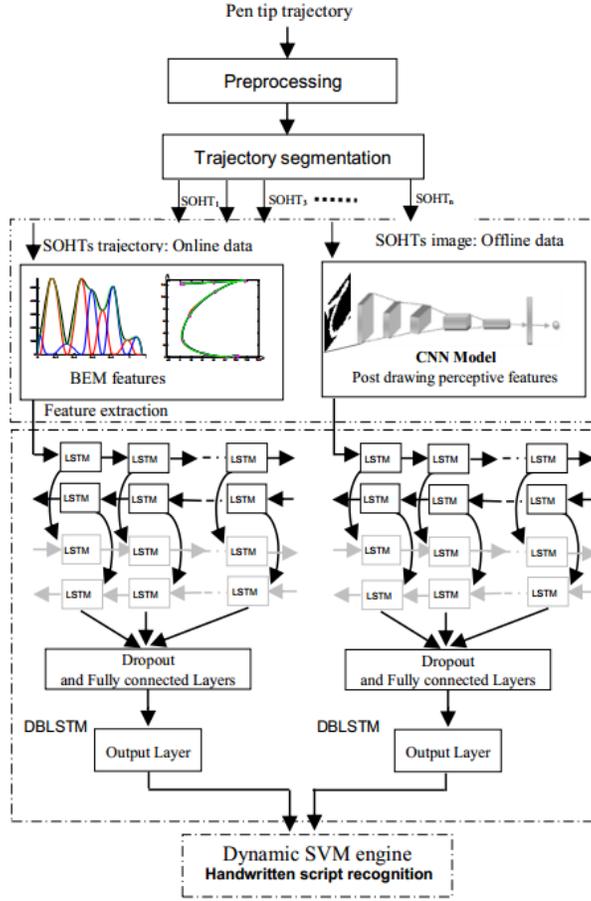


Figure 1: System architecture.

## 5. Preprocessing

The handwriting trajectories are collected online via a digitizing device. These trajectories are characterized by high variation which requires geometric

195 correction and denoising steps to reduce the handwriting variability, eliminate  
noise, and normalize the handwriting size. Given the handwriting trajectory,  
the Chebyshev type II low-pass filtering with a cut-off frequency of  $f_{cut}= 10\text{Hz}$   
is applied to attenuate the effect of noise and errors caused by spatial and tem-  
poral quantification of the acquisition system. The value of cut-off frequency  
200 results from a trade-offs between the conservation of the handwriting ripple pro-  
duced by young agile writers and the elimination of the physiological tremors  
generated by other older writers [37].

The used filter has a magnitude transfer function given by:

$$|H(j\omega)| = \frac{1}{\sqrt{1+\varepsilon^2 C_N^2(\omega_s/\omega)}} \quad (1)$$

Where

$$C_N(\omega_s/\omega) = \begin{cases} \cos[N\cos^{-1}(\omega_s/\omega)], & |\omega| \geq \omega_s \\ \cosh[N\cosh^{-1}(\omega_s/\omega)], & |\omega| \leq \omega_s \end{cases} \quad (2)$$

$\omega_s$  is a constant of frequency scaling, and  $\varepsilon$  is a constant that adjusts the in-  
205 fluence of  $C_N(\omega_s/\omega)$  in the denominator of  $|H(j\omega)|^2$ . Consequently, it is noted  
that the hyperbolic cosine is employed in Eq. 2 for low frequencies, and, from  
Eq. 1 that this results in a response near unity; the trigonometric cosine is  
used for high frequencies beyond  $\omega_s$  resulting in a rippling response of small  
magnitude.

210 As shown in Fig.2, the frequency response  $R(f)$  of the low-pass filter Cheby-  
shev type II is described by the following Eq. 3.

$$R(f) = \frac{\cos[N\cos^{-1}[\alpha\cos(\frac{\pi m}{N})]]}{\cosh[N\cosh^{-1}(10\frac{A}{-20})]}, m = 0, 1, 2, \dots, N - 1 \quad (3)$$

Where  $\alpha = \cosh(\frac{1}{N}\cosh^{-1}(10\frac{A}{-20}))$  and A is the needed sidelobe attenu-  
ation in decibel (dB). Fig.3 shows an example of applying the low-pass filter  
Chebyshev type II on online handwriting trajectory of Arabic character 'ي'.

215 Three decomposition zones namely upper, core, and lower regions are ex-  
tracted from the level of the horizontal fulcrum of handwriting trajectory. There-  
fore, a handwriting size normalizing procedure is applied to adjust its height to

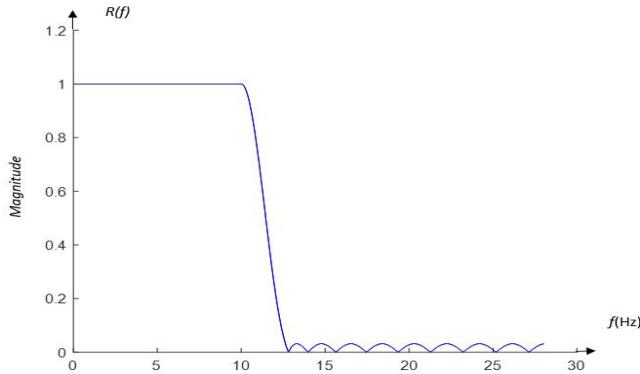


Figure 2: Filter frequency response.

a fixed value  $h = 128$ , while retaining the same length/height ratio [3], [4]. The pre-processing techniques are used and tested for the supported scripts such as Arabic, Latin, and digits. After denoising and eliminating the handwriting vari-

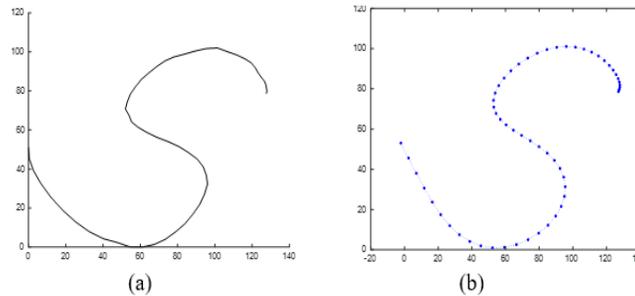


Figure 3: (a) An acquired trajectory of Arabic character 'ي' and (b) after Low-pass filtering and smoothing.

220

ation of the input *ink* signal using the preprocessing technique, the next step is to divide the online signal into SOHTs components, which are used subsequently for the pre-classification phase.

## 6. Feature extraction and SOHTs pre-classification

225

In this section, we describe the feature extraction method for the shape trajectory representation and SOHTs pre-classification step which serves to ame-

liorate the recognition rates. Indeed, we identify two types of feature classes in our work:

1. Progressive plot features are extracted for each stroke after segmentation using BEM. In this context, we benefit from this model to represent the dynamic (velocity) and static (geometric) profiles of the online handwritten trajectory.
2. Post-drawing or perceptive features are extracted from the bitmap of the image using the CNN model.

### 6.1. Beta-Elliptic modeling (BEM)

The BEM is derived from the kinematic Beta model with a juxtaposed analysis of the spatial profile. It considers a simple movement as the response to the neuromuscular system, which is described by the sum of impulse signals [38] as the Beta function [1]. The specificity of BEM is the combination of two aspects of the online handwriting stroke modeling: the velocity profile (dynamic features) represented by a beta function that culminates at a time  $t_C$  coinciding with a local extremum (maximum, minimum, or double inflexion point) as shown in Fig. 4.a), and an elliptic arc as illustrated in Fig. 4.b) modeling the static (geometric) profile for each stroke of the segmented trajectory. We describe in the following sub-section how the BEM process works.

#### 6.1.1. Velocity model

In the dynamic profile, the curvilinear velocity  $V_\sigma(t_c)$  shows a signal that alternates between extremums (minima, maxima, and inflexion points) which determine and delimit the number of trajectory strokes. For BEM,  $V_\sigma(t_c)$  can be reconstructed by overlapping Beta signals where each stroke corresponds to the generation of one beta impulse described by Eq. 4:

$$pulse\beta(t, p, q, t_0, t_1) = \begin{cases} K \left( \frac{t-t_0}{t_c-t_0} \right)^p \left( \frac{t_1-t}{t_1-t_c} \right)^q & \text{if } t \in [t_0, t_1] \\ 0 & \text{if not} \end{cases} \quad (4)$$

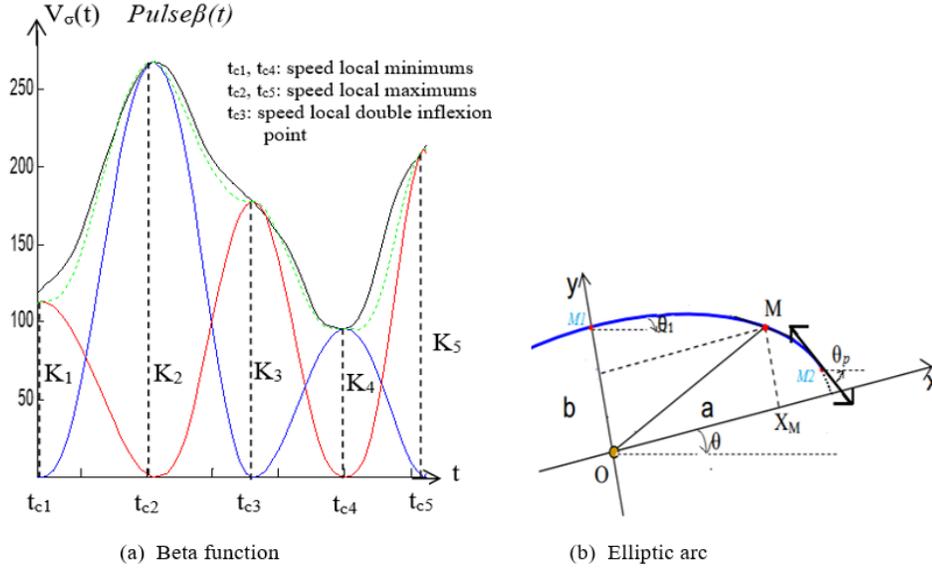


Figure 4: Online handwriting modeling using BEM.

with

$$t_c = \frac{p * t_1 + q * t_0}{p + q} \quad (5)$$

Where  $t_0$  and  $t_1$  are respectively the starting and the ending times of the generated impulse which delimiting the correspondent trajectory stroke,  $t_c$  is the instant when the beta function reaches its maximum value as depicted in Eq. 5,  $K$  called impulse amplitude,  $p$  and  $q$  are intermediate shape parameters.

$$V_\sigma(t) = \sum_{i=1}^n pulse\beta_i(k_i, t, p_i, q_i, t_{0i}, t_{1i}) \quad (6)$$

As described in Eq. 6, the velocity profile can be reconstructed by the overlapped beta signals. Some examples of the velocity profile modeling of the online Arabic script like character 'ح' and word 'علم', and Latin character 'a' was presented in Fig. 5.a), c), and e), respectively.

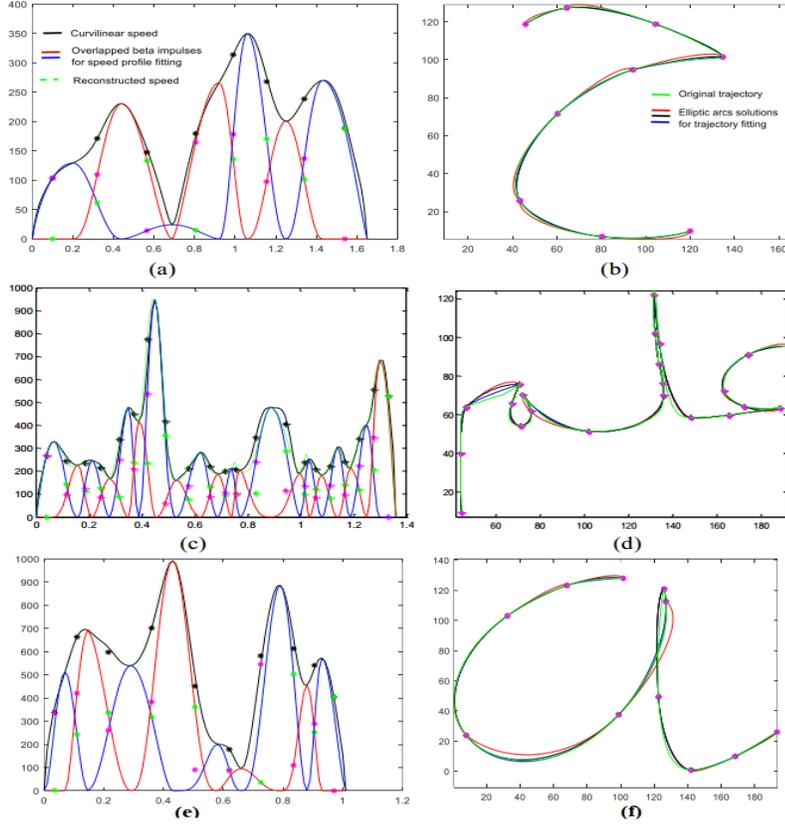


Figure 5: Online handwriting modeling of Arabic character 'ح' and word 'علم' and Latin character 'a' with BEM. (a), (c) and (e) describe the velocity profile, (b), (d) and (f) represent the geometric profile.

### 6.1.2. Trajectory modeling

In the space domain, many studies have been introduced for generating online handwriting trajectories . In Bezine et al. [4], each stroke located between two points  $M_1$  and  $M_2$  is included to an elliptic trajectory checking Eq. 7, where  
265  $X$  and  $Y$  denote the cartesian coordinates along the elliptic stroke,  $a$  and  $b$  are respectively the minor and major axis dimensions.

$$\frac{X^2 + Y^2}{a^2 + b^2} = 1 \quad (7)$$

Also, an elliptic arc is described by the trajectory tangent on their endpoints  $M_1$  and  $M_2$  which is invented by [39]. Indeed, each elementary beta stroke situated between two successive velocity extrema times can be represented by  
 270 an elliptic arc described by four geometric characteristics such as:  $a$ ,  $b$ ,  $\theta$ ,  $\theta_p$  as shown in Fig. 4. b). There  $a$  and  $b$  define respectively the half dimensions of the major and the minor axes of the elliptic arc,  $\theta$  is the inclination angle of the ellipse major axis, and  $\theta_p$  denotes the trajectory tangent inclination at the minimum velocity endpoint. These parameters reflect the geometric properties  
 275 of the end effector (pen or finger) trace, dragged by the set of muscles and joints involved in handwriting. Fig. 5.b), d), and f) depict respectively some examples of modeling geometric profile of the same chosen Arabic and Latin samples.

### 6.1.3. Hybrid Beta and Elliptical models

As mentioned previously, each simple stroke is described by 10 parameters  
 280 summarized in Table 2. The first six beta parameters give the overall temporal properties of the neuromuscular networks implicated in motion generation, whereas the last four elliptical parameters present the global geometric properties of all the muscles and joints inducted to execute the movement.

### 6.2. CNN features

285 In this sub-section, we briefly explain the use of CNN model in our context. Inspired by the recent progress of deep learning technologies in different topics, we have employed CNN architecture in the off-line script bitmaps reconstructed from the SOHTs database. The latter represents only the final layout of the hand drawing that skirts the chronologic style of its generation. Indeed,  
 290 the CNN features are extracted from the offline SOHTs by a convolution network that contains multiple convolutions and max-pooling layers. Also, the batch normalization and dropout techniques are applied to enhance the feature extraction step. The details of the CNN architecture are described in the experimental section. The input image ( $32 \times 32$ ) is transformed into CNN-feature  
 295 of size ( $Bachsize, L, D$ ). Specifically,  $Bachsize$  is fixed to 32,  $L$  and  $D$  are the

Table 2: Features Extraction generated by using BEM.

Features section	Parameters and formulas	Signification
Dynamic	K	Beta impulse amplitude
	$\Delta_t = t_1 - t_0$	Beta impulse duration
	$\frac{t_c - t_0}{t_1 - t_0}$	Rapport of Beta impulse asymmetry or culminating time
	p	Beta shape parameters.
Static	$V_i$	Initial training velocity amplitude.
	A	Maximal slope of the velocity training component.
	a	Ellipse large axis half-length
	b	Ellipse small axis half-length
	$\theta$	The ellipse major axis inclination angle.
	$\theta_p$	Angle of inclination of the tangents at the endpoint $M_2$ .

lengths and the depth of the CNN entity, respectively. As shown in Fig. 6, the CNN output in the trajectory modeling step is noticed by  $F = (F_1, F_2, \dots, F_N)$  and  $F_i \in R_D$ .

### 6.3. SOHTs pre-classification

300 After segmentation of the handwriting trajectory into SOHTs and the extraction of their offline visual features, we obtained an extensive database of multi-lingual handwriting segments with an indefinite number of labels due to the large variability of writers' handwriting styles, especially when mixing cursive and discrete styles. Thus, having failed to manually assign a label to  
305 each SOHT, we chose to use the k-means unsupervised clustering algorithm for the SOHTs pre-classification in the training track (see Fig. 6, training track).

The considered number of sub-groups  $K$  is defined empirically to maximize the recognition rate. Indeed, the change of the value  $K$  leads to the modification of the network's accuracy of each classifier and consequently on the overall fusion recognition system.

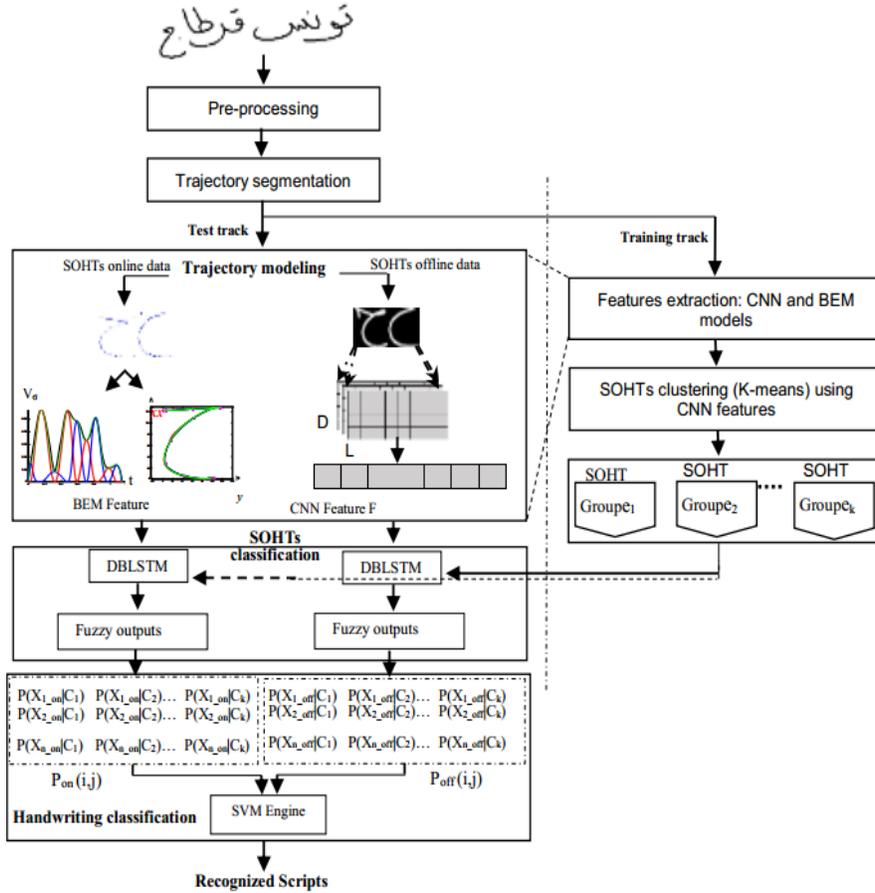


Figure 6: SOHTs pre-classification and online handwriting script recognition process.

310

## 7. Sequence Recognition

In our work, we introduced hybrid online and offline models based on DBLSTM for SOHTs fuzzy classification and SVM for handwriting sequence (word, let-

ters, digits, etc) recognition. In this section we present different modules of our  
 315 recognition process.

7.1. DBLSTM for SOHTs fuzzy classification

This step consists of training all SOHTs into their  $K$  groups using deep RNN  
 with BLSTM version. The choice of this type of network is demonstrated by the  
 fact that the handwriting trajectories is composed of a sequence of beta strokes  
 320 over time, and the ability of the RNN to model sequential data at each time.  
 The LSTM recurrent networks have memory, since they maintain a state vector  
 that implicitly contains information about the history of all the past elements  
 of a sequence. Hochreiter et al. [40] have introduced this LSTM network which  
 is often employed to learn longer-term dependencies and reduce the vanishing  
 gradient problem [41].

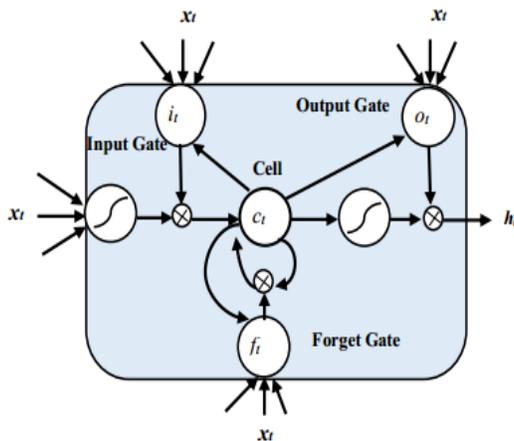


Figure 7: LSTM memory block with one cell. (Graves et al. [8])

325 As shown in Fig. 7, an LSTM layer is a special kind of RNN composed of  
 multiple subnets called memory blocks which are recurrently connected. Each  
 block includes a set of internal units known as cells which are used both for  
 storing and accessing information over a long time duration. The activation of  
 330 these cells is carry out by three multiplicative ‘gate’ units. At time-step  $t$ , there

are input gate  $i_t$ , forget gate  $f_t$ , output gate  $o_t$ , which are presented by the three equations Eq. 8 - Eq. 10.

$$i_t = \text{sigm}(W_i x_t + U_i h_{t-1} + b_i) \quad (8)$$

$$f_t = \text{sigm}(W_f x_t + U_f h_{t-1} + b_f) \quad (9)$$

$$O_t = \text{sigm}(W_o x_t + U_o h_{t-1} + b_o) \quad (10)$$

Where  $W_*$  represents the input-to-hidden weight matrix,  $U_*$  is the state-to-state recurrent weight matrix, and  $b_*$  is the bias vector.

$$\tilde{c} = \text{tanh}(W_c x_t + U_c h_{t-1} + b_c) \quad (11)$$

$$c_t = i_t \circ \tilde{c} + f_t c_{t-1} \quad (12)$$

$$h_t = O_t \circ \text{tanh}(c_t) \quad (13)$$

<sup>335</sup>  $i_t$  and  $f_t$  are used to control the updating of  $c_t$  which in turn saves the long-term memory.  $\circ$  is the operation denoting the element-wise vector product. The output gate  $o_t$  is used to control the updating of  $h_t$  as shown in Eq. 13. Given an input sequence  $[x_1, x_2, \dots, x_t]$ , we obtain a hidden state sequence of  $[h_1, h_2, \dots, h_t]$  by passing the input signal through the forward LSTM.

<sup>340</sup> In various tasks, it is necessary to employ both past and future complementary contexts instead of the RNN that only uses past contexts. In fact, to build a bidirectional LSTM (BLSTM) model, we combine two LSTM sub-layers in forwarding and backward directions [28]. Indeed, we can obtain another hidden state sequence of  $[h'_1, h'_2, \dots, h'_t]$  by passing the reversed sequence of  $[x_t,$   
<sup>345</sup>  $x_{t-1}, \dots, x_1]$  over the backward LSTM. All the hidden states are then fed to a fully connected layer followed by a softmax layer for final classification. The details of BLSTM architecture are described in the experimental section.

As mentioned previously, each trajectory  $T$  is composed of a number of *SOHTs*,  $T=\{SOHT_1, SOHT_2, \dots SOHT_n\}$ . In the training track, each *SOHTs* sample is assigned to the most likely group  $C_j$ ,  $j=1..K$ , according to its visual offline features stemming from CNN. Thereafter, this unsupervised assignment will serve to train two DBLSTM networks used for *SOHTs* fuzzy labeling. Indeed, the *SOHTs* labeling stage considers two types of feature sets for each handwriting segment  $SOHT_i$ ,  $i=1..n$ . The first set  $X_{i_{on}}$  combines dynamic and geometric features extracted using BEM model while the second  $X_{i_{off}}$  contains the post drawing features extracted by CNN model. As shown in Fig. 6, the output of each DBLSTM is a vector of size  $K$ :  $\{P(X_{i_{on}}|C_1)P(X_{i_{on}}|C_2)\dots P(X_{i_{on}}|C_K)\}$  and  $\{P(X_{i_{off}}|C_1)P_{i_{off}}(X_{off}|C_2)\dots P(X_{i_{off}}|C_K)\}$  representing the membership probabilities of the  $i^{th}$  analyzed handwritten trajectory segment to the  $K$  *SOHTs* groups for both online and offline branches respectively.

## 7.2. Regularization and data augmentation

Regularization is an important process to enhance the efficiency of deep recurrent neural networks. The most commonly used regularization method is the dropout technique [42]. It is used in the training phase to avoid the overfitting problem by randomly dropping hidden units. In our case, we use the dropout technique in both input and fully-connected layers with the probability of 0.3 which is similar to the approach taken in [30]. Also, the wide number of training data is another key to the success of deep neural networks. To increase our training dataset, we adopt the data augmentation strategy which is widely used in image-based recognition systems [43]. This approach proceeds by generating randomly distorted samples from those composing the original dataset. For this, we modified some parameters like the inclination angle of the trajectory, baselines, handwriting italicity, smoothing, etc. These techniques increase the training set, generate more training samples, and also bring more variation in the training set.

### 7.3. SVM fusion engine for script recognition

After SOHTs fuzzy labeling, the outputs of DBLSTM of both online and of-  
 380 fine sub-system are treated as input for the SVM classifier to make decision fu-  
 sion. Indeed, SVM is considered as an efficient tool for both linear and nonlinear  
 classification based on a supervised learning algorithm. It has proven high suc-  
 cess in several practical applications such as pattern classification. As shown in  
 Fig.8.a), SVM usually determines an optimal separating hyper-plane by adopt-  
 ing a novel technique that maps the sample points into a high-dimensional  
 385 feature space using a nonlinear transformation. It was originally designed to  
 solve binary classification problems. However, it can be employed also to solve  
 multiclass problems (see Fig.8.b) using several methods such as one-versus-all  
 [44] based on dot product functions (kernel functions) in feature space. Exam-  
 ples of these functions, linear, Radial Basis Function (RBF), sigmoid function,  
 etc. In our context, we use RBF function as a kernel in the hybrid model which

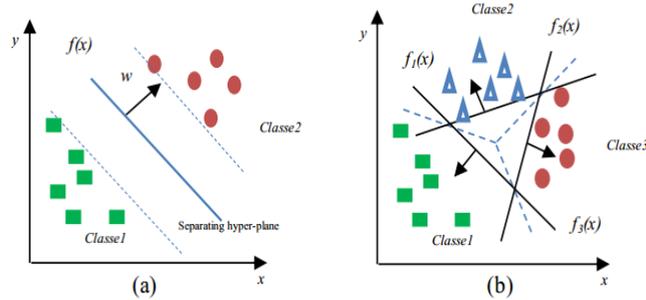


Figure 8: Principle of SVM. (a) two-class hyper-plane example, (b) one-versus-all method

390

is described as:

$$k(x, x') = \phi(x)\phi(x') = \exp(-\gamma\|x - x'\|^2) \quad (14)$$

Where  $\gamma$  are the parameters that will be determinate empirically,  $x$  and  $x'$  denote the input vectors respectively,  $\phi$  is a nonlinear transform features space.

For the recognition of the overall handwritten script, the SOHTs vectors  
 395 belonging to the same online trajectory (word, letter, digit, etc) are gathered to

form the fuzzy SOHTs membership matrix  $P_{on}(i, j)$  and  $P_{off}(i, j)$  provided as input layer for the SVM which allows to merge local decisions and determines the classification of the overall handwritten script.

## 8. Experiments and results

400 In this section, we describe our experimentation which is performed on multi-lingual online handwriting recognition of the scripts. The utilized datasets are initially presented followed by the conducted ablation studies and discussion of the results. Here, two metrics: the Character Error Rate (CER) and Word Error Rate (WER) were employed to evaluate the proposed system defined as  
405 the percentage of characters or words recognized correctly. These metrics are evaluated with the output of each DBLSTM and also with the fusion of them using SVM engine. Next, we compare our approach with those of the state-of-the-art approaches using the same databases. Finally, we present some strengths and limitations of the proposed system in the error analysis section. All tests  
410 were implemented using core i7 processor of 3.2 GHz, 8 GB of RAM.

### 8.1. Datasets overview

One of the most difficult problems of online handwriting recognition is the requirement for a standard database that serves a variety of writing styles and includes the various classes in the target language. In order to test the per-  
415 formance and efficiency of our proposed system, we have used three datasets: Online-KHATT [45] for isolated Arabic characters, ADAB [46] for Arabic words, and UNIPEN [47] for Latin characters and digits. In this section, we describe these publicly available datasets.

#### 8.1.1. Online-KHATT dataset

420 Online-KHATT is a new open-vocabulary Arabic dataset proposed by [36]. It is composed of 10,040 lines of Arabic online text taken from 40 books, which are written by about 623 participants using android and windows-based devices. The authors claim that this database is challenging which consists of many

problems such as thickness, writing styles, dots number, and position. In our  
 425 experiment, we use a subset of the segmented characters of this dataset [13]  
 that contain 44,795 data after using data augmentation technique. The subset  
 is divided into a training set with 30,795 samples, a validation set of 4,000  
 samples, and a test set of 10,000 data. As described in the previous section,  
 the shapes of the letters differ depending on the position in the word they are  
 430 found, which makes a total of 114 different shapes in this dataset.

### 8.1.2. Unipen benchmark dataset

The Unipen dataset contains three sections 1a, 1b, and 1c for digits, upper,  
 and lowercase Latin characters. The size of each data section is 16K, 28K, and  
 61K respectively. This database is divided into training, validation, and test  
 435 sets. Table 3 summarizes the total number of characters for each section.

Table 3: UNIPEN Data set statistic.

Dataset	Number of characters	
Section 1.a	Train	15,953
	Test	5,598
	Validation	3,000
Section 1.b	Train	61,359
	Test	27,470
	Validation	10,000
Section 1.c	Train	40,092
	Test	16,560
	Validation	10,000

### 8.1.3. ADAB dataset

The ADAB (Arabic DAtaBase) has been employed extensively in the litera-  
 ture. It consists of more than 33,000 Arabic words handwritten by 170 different  
 writers. The text contains 937 Tunisian town/village names. As presented in

440 Table 4, this database is partitioned into six distinct sets which were collected for the ICDAR 2011 online Arabic handwriting recognition competition [46].

Table 4: ADAB dataset description.

Sets	Number of words	Number of pseudo words	Writers
Set 1	5037	40296	56
Set 2	5090	25450	37
Set 3	5031	15093	39
Set 4	4417	22085	25
Set 5	1000	4000	6
Set 6	1000	8000	3
Total	21575	114924	166

In order to increase the training accuracy, we have applied the data augmentation technique. We used sets 1, 2, and 3 which contain more than 45,158 words for the training process. We employed sets 5 and 6 which contain 4,000 samples for validation. Then we tested our system using 8,417 words from set 4.

### 8.2. Ablation studies

To study the impact of the proposed architecture for recognizing multilingual scripts using hybrid DBLSTM and SVM engine on both online and offline branches, we have designed three groups of experiments.

The first experiment is related to the pre-classification step which consists in choosing the value  $K$  corresponding to the best grouping of SOHTs. After trajectory segmentation, we construct a dataset composed of more than 200.000 SOHTs. These were later classified into  $K$  groups by using k-means clustering algorithm based on extracted CNN parameters. After several tests, the value  $K$  is fixed empirically to 210, which represents the number of groups that return the smallest sums of intra-cluster points to their centroid distances. 70% of this

dataset is used for training the next stage composed of DBLSTM and the rest for the classification tests.

Table 5: CNN configuration: Conv means Conv, Norm-ReLU layer. The convolution and max-pooling layers' parameters are represented as "Conv: the size of filter" and "Max pool: filter size", respectively.

CNN1	CNN2	CNN3
32x32	32x32	32x32
		Conv: $3 \times 3$ (50)
		Conv: $3 \times 3$ (100)
		Conv: $3 \times 3$ (100)
	Conv: $3 \times 3$ (50)	Max-Pool: $2 \times 2$
	Conv: $3 \times 3$ (100)	Conv: $3 \times 3$ (150)
	Max-Pool: $2 \times 2$	Conv: $3 \times 3$ (200)
Conv: $3 \times 3$ (50)	Conv: $3 \times 3$ (150)	Conv: $3 \times 3$ (200)
Max-Pool: $2 \times 2$	Conv: $3 \times 3$ (200)	Max-Pool: $2 \times 2$
Conv: $3 \times 3$ (150)	Max-Pool: $2 \times 2$	Conv: $3 \times 3$ (250)
Max-Pool: $2 \times 2$	Conv: $3 \times 3$ (250)	Conv: $3 \times 3$ (300)
Conv: $3 \times 3$ (250)	Conv: $3 \times 3$ (300)	Conv: $3 \times 3$ (300)
Max-Pool: $2 \times 2$	Max-Pool: $2 \times 2$	Max-Pool: $2 \times 2$
Conv: $3 \times 3$ (350)	Conv: $3 \times 3$ (350)	Conv: $3 \times 3$ (350)
Max-Pool: $2 \times 2$	Conv: $3 \times 3$ (400)	Conv: $3 \times 3$ (400)
	Max-Pool: $2 \times 2$	Conv: $3 \times 3$ (400)
		Max-Pool : $2 \times 2$
Accuracy 87.24%	94.48%	98.72%

#### 460 8.2.1. Structure of CNN feature extraction

The second experiment consists in choosing the architecture that offers the best CNN feature. For this, we tried several settings of CNN to train the offline system on the SOHTs training dataset. As illustrated in Table 5, three variations of CNN model were designed by changing the number of layers and filters

465 in each layer. We chose CNN3 with 13 convolution layers and 4 Max-Pooling  
layers similar to Zhang et al. [30] because it is the most efficient architecture  
which generates a high accuracy on SOHTs dataset. The input layer is of size  
32x32 of gray-level image. The applied filter of convolutional layers is 3x3 with a  
fixed convolution stride of one. The feature maps dimension in each convolution  
470 layer is increased gradually from 50 in layer-1 to 400 in layer-12. After three  
convolutional layers, a max-pooling 2x2 window with stride 2 is implemented  
to halve the size of the feature map. The network is trained using its param-  
eters: a stochastic gradient descent with momentum SGDM, the learning rate  
was 0.001, and 100 mini-batch size. We use also the dropout technique with 0.2  
475 only in the last layer.

### 8.2.2. DBLSTM parameters

In the proposed system, we used the same architecture of DBLSTM for  
SOHTs fuzzy classification on both online and offline data, respectively. The  
network parameters in training data are fixed after several tests. The best  
480 topology of our network architecture is composed of three bidirectional hidden  
layers which give good results. The size of the input layer depends on the  
dimension of feature vector using BEM or CNN respectively. We have also  
changed the number of nodes (32, 64, 128, 256, and 400) in the forward and  
backward hidden layers. The best number of nodes applied in our DBLSTM  
485 is 400 nodes as described in the experimental results section. The size of the  
output layer is defined by the number of SOHTs class. Dropout is used in  
fully connected layer with a probability of 0.3. The network is trained using its  
parameters: a SDGM and mini-batch of size 200. The training process is started  
with an initial learning rate of 0.001 and 400 maximum number of epochs. A  
490 categorical cross-entropy loss function is used to optimize the network. After  
each epoch, the training data is shuffled to make different mini-batches.

### 8.3. Experimental Results

One of the most important problems of handwriting recognition is the selection of the relevant set of features, that has been the subject of several studies [48]. Since the focus of our work is to show that the used method which is based on the combination of two approaches of handwriting representation described above, can be useful in online handwriting recognition. To understand the effectiveness of our method, we have planned three groups of experiments: one is based on dynamic and geometric features obtained by using BEM, the second is based on bitmap feature from CNN model while the third is on the fusion of the two.

#### 8.3.1. Experiments on Online-KHATT

To evaluate the performance of our proposed system for Arabic characters, we used Online-KHATT dataset described previously. As shown in Fig. 9 and Table 6, we conduct further study by changing the number of layers and nodes per layer for both BEM and CNN features to determine the optimal size of the BLSTM model. We observe that for the two input features, the use of 3 layers surpasses more shallow networks, and the addition of more layers brings almost no improvement. Further, using 256 nodes per layer is enough, as larger networks provide only limited improvements, if at all.

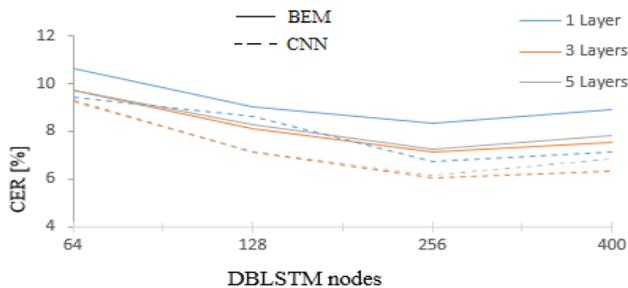


Figure 9: CER of trained DBLSTM models on the Online-KHATT dataset with different numbers of LSTM layers and nodes using BEM and CNN inputs. The solid lines illustrate the results using BEM, and dashed lines indicate the results with CNN model.

510

Table 6: CER (%) on the Online-KHATT dataset for different BLSTM layers and nodes.

System	DBLSTM	128 Nodes	256 Nodes	400 Nodes
BEM	1 Layer	9.02	8.35	8.93
	3 Layers	8.13	<b>7.13</b>	7.56
	5 Layers	8.29	7.25	7.84
CNN	1 Layer	8.62	6.75	7.14
	3 Layers	7.13	<b>6.05</b>	6.35
	5 Layers	7.16	6.19	6.84

We can see a significant improvement in the CER using CNN than BEM model which is explained by the efficient offline features extracted by CNN model allowing a fair compromise between discrimination of character class and different writing style gathering. For the last stage of the proposed system, we combined the outputs of the DBLSTMs in order to improve the results. To do this, we carried out some tests using SVM engine with different kernel functions. As shown in Table 7, the CER on Online-KHATT dataset has been decreased with RBF function to 5.25% with the hybrid of the two models.

Table 7: Error Rate (ER) on Online-KHATT and ADAB datasets using SVM engine with different kernel function.

Dataset	Kernel function			
	Linear	Sigmoid	RBF	Polynomial
Online-KHATT	7.75	5.33	<b>5.25</b>	6.39
ADAB	2.65	1.44	<b>1.28</b>	2.18

Moreover, we compare our system to others that also experimented on online Arabic character recognition. The results are summarized in Table 8 which present for each work: the used classifier, feature extractor model, and accuracy of the results. We can see that the performance obtained by our system is better than the two presented systems which are also trained on the same training dataset. We observe that the obtained results on online-KHATT dataset are

525 significantly better than those found by [13] and [25]. This is due to the use of the  
 complementarity models for handwriting representation on the one hand which  
 is described simultaneously by the geometric and dynamic features extracted by  
 using BEM and the strength of a deep learning CNN model, which represents the  
 discriminating power to differentiate Arabic characters. On the other hand, the  
 530 combination models allow the enhancement of the global system discriminating  
 power.

Table 8: CER on Online-KHATT in Comparison to the state of the Art.

System	Feature	CER (%)
ANN [13]	Statistical features + chain codes	18.00
LSTM [25]	Path signature feature	7.43
Our system using DBLSTM-SVM	BEM + CNN	5.25

### 8.3.2. Experiments on UNIPEN

We evaluate the performance of our system on the test set of online signals  
 from UNIPEN dataset distributed in three sections 1a, 1b, 1c. Table 9 shows  
 535 comparison results with other previous studies using the same database. We  
 observe that the results achieved by our system using the combination of the  
 two models are better than the ‘writer-independent’ experiment reported by  
 [49] that uses cluster generative statistical dynamic time warping (CSDTW).  
 This indicated that our proposed method is also robust and very promising for  
 540 online handwriting Latin characters and digits. Again, we refer to more recent  
 work [50] which uses LSTM with CTC. We note that the recognition rate is  
 very competitive. It would be noted that UNIPEN contains very difficult data  
 because of wide variety of writers, mislabeled, or noisy data.

Table 9: CER on UNIPEN Data compared to the State of the Art.

Section	Our system				
	BEM	CNN	Fusion	[50]	[49]
1a	3.35	2.80	1.70	0.80	2.90
1b	7.76	6.10	4.70	3.10	7.20
1c	8.62	7.78	6.50	5.10	9.30

### 8.3.3. Experiments on ADAB

545 We have also carried out experiments on the test set of the ADAB database using the same network architecture. As summarized in Table 10 and Fig 10, the obtained WER results show the effectiveness of our architecture using single models. In the same way, the lower WER is achieved when we use 3 layers of BLSTM with 256 nodes.

550 Further, the performance of our system has been gained when we combined both DBLSTM models. As shown in Table 7, the low WER is 1.28% which was obtained employing SVM with RBF function.

To better understand the efficiency and the robustness of our system, we discuss the performance compared to the previous state-of-the-art systems. It may be noted from Table 11 that our system provides better results compared to other works.

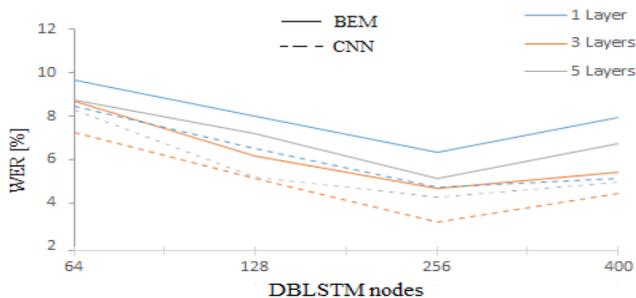


Figure 10: WER of DBLSTM models trained on the ADAB dataset by modifying the numbers of LSTM layers and nodes using BEM and CNN inputs.

Table 10: WER (%) on ADAB Data for different BLSTM layers and Nodes.

System	DBLSTM	128 Nodes	256 Nodes	400 Nodes
BEM	1 Layer	8.02	6.35	7.73
	3 Layers	6.15	<b>4.70</b>	5.43
	5 Layers	7.19	5.15	6.74
CNN	1 Layer	6.52	4.75	5.12
	3 Layers	5.16	<b>3.13</b>	4.46
	5 Layers	5.19	4.25	4.94

Indeed, despite the association of on-line and off-line data are previously used in related works for handwriting recognition, our developed system is distinguished from the literature [18], [16] by the integration of multi-stage deep learning networks for the analysis and the fuzzy classification of both online and offline data. Also, we adopt the decision model merger between the fuzzy classification results obtained by the two branches classifiers of respectively online and offline data using SVM engine. This new architecture has demonstrated its efficiency in the test phase by achieving advantageous WER 1.28% concerning the state of the art and in particular, comparing to the system of [51] which uses deep learning network only on the branch of offline data processing.

Table 11: WER on ADAB Data compared to the State of the Art.

System	Feature	WER (%)
SVM [51]	BEM + CDBN	2.49
HMM [16]	Directional features + Offline features	2.22
MLP/HMM [18]	BEM features	3.55
SVM [19]	BEM + perceptual code	1.42
Our system using DBLSTM-SVM	BEM + CNN	1.28

#### 8.4. Error Analysis

We demonstrate in this sub-section the strengths and limitations of the present system by depicting some examples of correct and misclassified sequences. Fig. 11 shows some examples of handwritten characters, words, and digits from the used datasets, processed by our recognition system. Each item is annotated with the label of the corresponding handwritten proposition.

We can identify some samples that are not recognized by one of the used models and corrected by the other and vice versa. For instance, the first and third top samples of corrected partition are well classified by the online BEM and not handled well by CNN. Likewise, the second, fourth, and right fifth samples are recognized only by CNN model. Based on these observations, the accuracy of our global system can be increased by using the combination of the two modules which explain the complementarity of them.

However, despite the good results achieved by our system, we notice that the result obtained using Online-KHATT and UNIPEN datasets remains to be improved. This is due to the strong similarity between several characters and the great diversity of writing styles. We note that some examples are quite distorted such as the first sample of 'ح', the third sample of 'ر' for the Arabic script as well as the '3' and '9' for digit samples in error samples partition.

Also, we notice that the first and the fourth sample of 'ر' on the one hand, the first and the third sample of 'د' on the other hand are legible but misclassified, most likely because they have an appearance rare among this letter's training examples. Likewise, a confusion between different forms of the position of the same character such as isolated 'و' and the end 'و', and the same forms of position like middle 'ب' and middle 'ي' which can be explained by the wrong identification of the delayed stroke. Other examples are not recognized by our system for Latin scripts such as the three samples of the uppercase characters 'F' and 'C', the lowercase of characters 'b' which can be explained by the diversity of writing style and the similarity between many characters of these scripts.

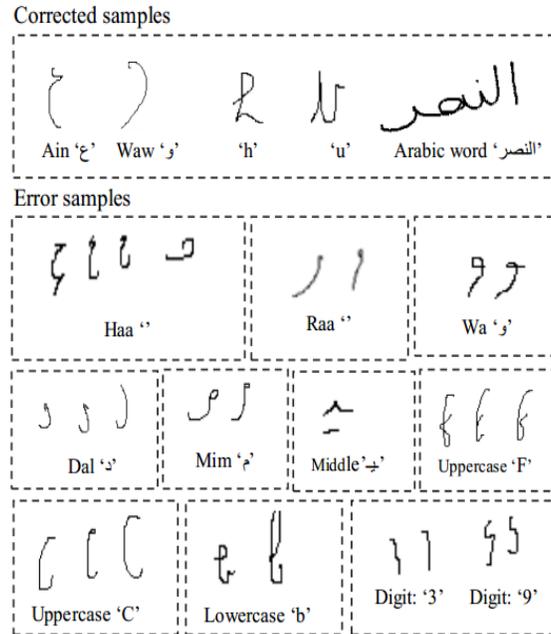


Figure 11: Some corrected and misclassified samples of Arabic, Latin characters, and digits.

## 9. Conclusion and Future Work

We presented in this study a multi-lingual online handwriting recognition system based on the combination of DBLSTM and SVM. The proposed system divides the online handwriting trajectory into SOHTs and the pre-classified into sub-groups. Then, two online handwriting representation models are employed: the BEM which is characterized by the presence of dynamic and geometric features of online pen-tip trajectory, and a powerful feature extracted using CNN model. Indeed, we employ the extracted features to evaluate the classification system and compare the performance of single classifiers using DBLSTM. We proceed also by combining the obtained results using SVM to improve the discrimination power of the global system.

The effectiveness of the corresponding models, and of a combination of these, was evaluated in experiments using three databases; online-KHATT for on-line Arabic characters, ADAB for Arabic words, and Unipen for Latin characters and

610 digits. By combining models, the obtained results suggest that our proposed method is well suited to online handwriting recognition compared to other works in the state-of-the-art which can be demonstrated by the complementary and ascendancy of both models.

We have also demonstrated that the used features extraction models are  
615 rather generic and their applicability in the other scripts such as Persian, English, Chinese, etc., is interesting and we consider as future work. Again, we schedule to extend our method for continuous text recognition.

## References

- [1] A. M. Alimi, An evolutionary neuro-fuzzy approach to recognize on-  
620 line arabic handwriting, 4th International Conference Document Analysis and Recognition (ICDAR '97) 2 (18–20) (1997) 382–386. doi:10.1016/S0031-8914(53)80099-6.
- [2] S. J. Cho, J. Kim, Bayesian network modeling of strokes and their relationships for on-line handwriting recognition, 6th International Conference  
625 Document Analysis and Recognition (ICDAR '2001) 2 (10–13) (2001) 86–90. doi:10.1109/ICDAR.2001.953760.
- [3] H. Boubaker, A. Chaabouni, N. Taguougui, M. Kherrallah, A. M. Alimi, Handwriting and hand drawing velocity modeling by superposing beta impulses and continuous training component, International Journal of Computer  
630 Science Issues (IJCSI) 10 (10–13) (2013) 57–63.
- [4] H. Bezine, A. M. Alimi, N. Sherkat, Generation and analysis of handwriting script with the beta-elliptic model, in: Ninth International Workshop on Frontiers in Handwriting Recognition, 2004, pp. 515–520. doi:10.1109/IWFHR.2004.45.
- 635 [5] T. Dhieb, N. Rezzoug, H. Boubaker, P. Gorce, A. M. Alimi, Effect of age on hand drawing movement kinematics, Computer Methods in Biomechanics and Biomedical Engineering 22. doi:10.1080/10255842.2020.1714235.

- [6] T. Dhieb, W. Ourda, H. Boubaker, A. M. Alimi, Beta-elliptic model for writer identification from online arabic handwriting, *J. of Information Assurance and Security* 11.
- [7] L. Sun, T. Su, C. Liu, R. Wang, Deep lstm networks for online chinese handwriting recognition, in: 2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2016, pp. 271–276. doi:10.1109/ICFHR.2016.0059.
- [8] A. Graves, Generating sequences with recurrent neural networks (2014). arXiv:1308.0850.
- [9] A. Graves, M. Liwicki, H. Bunke, J. Schmidhuber, S. Fernández, Unconstrained on-line handwriting recognition with recurrent neural networks, in: J. Platt, D. Koller, Y. Singer, S. Roweis (Eds.), *Advances in Neural Information Processing Systems*, Vol. 20, Curran Associates, Inc., 2008, p. 577–584.
- [10] Y. Hamdi, H. Boubaker, T. Dhieb, A. Elbaati, A. M. Alimi, Hybrid dblstm-svm based beta-elliptic-cnn models for online arabic characters recognition, in: 2019 International Conference on Document Analysis and Recognition (ICDAR), 2019, pp. 545–550. doi:10.1109/ICDAR.2019.00093.
- [11] H. M. Eraqi, S. A. Azeem, An on-line arabic handwriting recognition system: Based on a new on-line graphemes segmentation technique, in: 2011 International Conference on Document Analysis and Recognition, 2011, pp. 409–413. doi:10.1109/ICDAR.2011.90.
- [12] N. Mezghani, A. Mitiche, M. Cheriet, Bayes classification of online arabic characters by gibbs modeling of class conditional densities, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30 (7) (2008) 1121–1131. doi:10.1109/TPAMI.2007.70753.
- [13] B. M. Al-Helali, S. Mahmoud, A statistical framework for online arabic character recognition, *Cybernetics and Systems* 47 (2016) 478 – 498.

- [14] 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), 2017, pp. 153–157. doi:10.1109/ASAR.2017.8067778.
- 670 [15] M. S. El-Wakil, A. A. Shoukry, On-line recognition of handwritten isolated arabic characters, Pattern Recognition 22 (2) (1989) 97–105. doi:https://doi.org/10.1016/0031-3203(89)90058-7.
- [16] I. Abdelaziz, S. M. Abdou, Altecondb: A large-vocabulary arabic on-line handwriting recognition database, CoRR abs/1412.7626. arXiv: 1412.7626.  
675 URL <http://arxiv.org/abs/1412.7626>
- [17] S. Kubatur, M. Sid-Ahmed, M. Ahmadi, A neural network approach to online devanagari handwritten character recognition, in: 2012 International Conference on High Performance Computing Simulation (HPCS), 2012, pp. 209–214. doi:10.1109/HPCSim.2012.6266913.  
680
- [18] N. Tagougui, M. Kherallah, Recognizing online Arabic handwritten characters using a deep architecture, in: A. Verikas, P. Radeva, D. P. Nikolaev, W. Zhang, J. Zhou (Eds.), Ninth International Conference on Machine Vision (ICMV 2016), Vol. 10341, International Society for Optics and Photonics, SPIE, 2017, pp. 107 – 111. doi:10.1117/12.2268419.  
685
- [19] R. Zitouni, H. Bezine, N. Arous, Online handwritten arabic scripts recognition using stroke-based class labeling scheme, International Journal of Computational Intelligence Systems 14 (2020) 187–198. doi:https://doi.org/10.2991/ijcis.d.201024.001.
- 690 [20] R. Zouari, H. Boubaker, M. Kherallah, Towards online handwriting recognition system based on reinforcement learning theory, in: Neural Information Processing - 27th International Conference, ICONIP 2020, Bangkok,

- Thailand, November 18-22, 2020, Proceedings, Part IV, Vol. 1332 of Communications in Computer and Information Science, 2020, pp. 562–570. doi:10.1007/978-3-030-63820-7\\_64.
- 695
- [21] A. N., Al-Turaiki, Arabic handwriting recognition system using convolutional neural network, *International Journal of Computational Intelligence Systems* 33 (2020) 2249–2261. doi:doi.org/10.1007/s00521-020-05070-8.
- 700 [22] X. . Zhou, J. . Yu, C. . Liu, T. Nagasaki, K. Marukawa, Online handwritten japanese character string recognition incorporating geometric context, in: *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, Vol. 1, 2007, pp. 48–52. doi:10.1109/ICDAR.2007.4378673.
- 705 [23] Y. Yin, X. Zheng, B. Hu, Y. Zhang, X. Cui, Eeg emotion recognition using fusion model of graph convolutional neural networks and lstm, *Applied Soft Computing* 100 (2021) 106954. doi:https://doi.org/10.1016/j.asoc.2020.106954.
- [24] 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 Recognition* 65 (2017) 251–264. doi:https://doi.org/10.1016/j.patcog.2016.12.026.
- 710
- [25] D. Wilson-Nunn, T. Lyons, A. Papavasiliou, H. Ni, A path signature approach to online arabic handwriting recognition, in: *2018 IEEE 2nd International Workshop on Arabic and Derived Script Analysis and Recognition (ASAR)*, 2018, pp. 135–139. doi:10.1109/ASAR.2018.8480300.
- 715
- [26] R. Ghosh, C. Vamshi, P. Kumar, Rnn based online handwritten word recognition in devanagari and bengali scripts using horizontal zoning, *Pattern Recognition* 92 (2019) 203–218. doi:https://doi.org/10.1016/j.patcog.2019.03.030.
- 720

- [27] R. Maalej, M. Kherallah, N. Arous, Improving the dblstm for on-line arabic handwriting recognition, *Multimedia Tools and Applications* 79. doi:10.1007/s11042-020-08740-w.
- [28] M. Schuster, K. K. Paliwal, Bidirectional recurrent neural networks, *IEEE Transactions on Signal Processing* 45 (11) (1997) 2673–2681. doi:10.1109/78.650093.
- [29] L. S. YaegerBrandyn, J. WebbRichard, F. Lyon, Combining neural networks and context driven search for on-line, printed handwriting recognition in the Newton, Vol. 7700, Springer, Berlin, Heidelberg, 2012.
- [30] X.-Y. Zhang, Y. Bengio, C.-L. Liu, Online and offline handwritten chinese character recognition: A comprehensive study and new benchmark, *Pattern Recognition* 61 (2017) 348–360. doi:https://doi.org/10.1016/j.patcog.2016.08.005.
- [31] B. A., S. Madhvanath, Hidden markov models for online handwritten tamil word recognition, in: *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, Vol. 1, 2007, pp. 506–510. doi:10.1109/ICDAR.2007.4378761.
- [32] V. Carbune, P. Gonnet, T. Deselaers, H. A. Rowley, A. Daryin, M. Calvo, L.-L. Wang, D. Keysers, S. Feuz, P. Gervais, Fast multi-language lstm-based online handwriting recognition (2020). arXiv:1902.10525.
- [33] 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 (9) (2019) 12103–12123. doi:10.1007/s11042-018-6764-0.
- [34] J. A. Pittman, Handwriting recognition: Tablet pc text input, *Computer* 40 (9) (2007) 49–54. doi:10.1109/MC.2007.314.

- [35] UNESCO, World arabic language day., in: UNESCO: [www.unesco.org/new/en/unesco/events/prizes-and-celebrations/celebrations/international-days/world-Arabic-language-day](http://www.unesco.org/new/en/unesco/events/prizes-and-celebrations/celebrations/international-days/world-Arabic-language-day), 2012, p. 2.
- 750 [36] B. M. Al-Helali, S. A. Mahmoud, Arabic online handwriting recognition (AOHR): A survey, *ACM Comput. Surv.* 50 (3) (2017) 33:1–33:35. doi:10.1145/3060620.  
URL <https://doi.org/10.1145/3060620>
- [37] J. H. McAuley, C. D. Marsden, Physiological and pathological tremors and  
755 rhythmic central motor control, *Brain* 123 (8) (2000) 1545–1567. doi:10.1093/brain/123.8.1545.
- [38] R. Plamondon, C. Feng, A. Woch, A kinematic theory of rapid human movement. part IV: a formal mathematical proof and new insights, *Biol. Cybern.* 89 (2) (2003) 126–138. doi:10.1007/s00422-003-0407-9.
- 760 [39] H. Boubaker, N. Rezzoug, M. Kherrallah, P. Gorce, Alimi, Spatiotemporal representation of 3d hand trajectory based on beta-elliptic models, *Comput Methods Biomech Biomed Engin* 15 (2015) 1632–47. doi:10.1080/10255842.2014.940331.
- [40] S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural Computation* 9 (1997) 1735–1780.  
765
- [41] Y. Bengio, P. Simard, P. Frasconi, Learning long-term dependencies with gradient descent is difficult, *IEEE Transactions on Neural Networks* 5 (2) (1994) 157–166. doi:10.1109/72.279181.
- [42] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov,  
770 Dropout: A simple way to prevent neural networks from overfitting, *Journal of Machine Learning Research* 15 (56) (2014) 1929–1958.
- [43] G. P. Van Galen, J. F. Weber, On-line size control in handwriting demonstrates the continuous nature of motor programs, *Acta Psychologica*

- 100 (1) (1998) 195–216. doi:[https://doi.org/10.1016/S0001-6918\(98\)00034-1](https://doi.org/10.1016/S0001-6918(98)00034-1).
- 775
- [44] G. P. Van Galen, J. F. Weber, On-line size control in handwriting demonstrates the continuous nature of motor programs, *Acta Psychologica* 100 (1) (1998) 195–216. doi:[10.1016/S0001-6918\(98\)00034-1](https://doi.org/10.1016/S0001-6918(98)00034-1).
- [45] S. A. Mahmoud, H. Luqman, B. M. Al-Helali, G. BinMakhashen, M. T. Parvez, Online-khatt: An open-vocabulary database for arabic onlinetext processing, *The Open Cybernetics & Systemics Journal* 12 (2018) 42–59.
- 780
- [46] H. Boubaker, N. Rezzoug, M. Kherrallah, P. Gorce, Alimi, Online arabic databases and applications, Märgner V., El Abed H. (eds) *Guide to OCR for Arabic Scripts*. Springer, London. (2012) S41–S57doi:[10.1007/978-1-4471-4072-6\\_22](https://doi.org/10.1007/978-1-4471-4072-6_22).
- 785
- [47] I. Guyon, L. Schomaker, R. Plamondon, M. Liberman, S. Janet, UNIPEN project of on-line data exchange and recognizer benchmarks, in: 12th IAPR International Conference on Pattern Recognition, Conference B: Patern Recognition and Neural Networks, ICPR 1994, Jerusalem, Israel, 9-13 October, 1994, Volume 2, 1994, pp. 29–33. doi:[10.1109/ICPR.1994.576870](https://doi.org/10.1109/ICPR.1994.576870).
- 790
- [48] V. Onnia, M. Tico, J. Saarinen, Feature selection method using neural network, in: *Proceedings 2001 International Conference on Image Processing (Cat. No.01CH37205)*, Vol. 1, 2001, pp. 513–516 vol.1. doi:[10.1109/ICIP.2001.959066](https://doi.org/10.1109/ICIP.2001.959066).
- [49] C. Bahlmann, H. Burkhardt, The writer independent online handwriting recognition system frog on hand and cluster generative statistical dynamic time warping, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26 (3) (2004) 299–310. doi:[10.1109/TPAMI.2004.1262308](https://doi.org/10.1109/TPAMI.2004.1262308).
- 795
- [50] V. Carbune, P. Gonnet, T. Deselaers, H. Rowley, A. Daryin, M. C. Lafarga, L.-L. Wang, D. Keysers, S. Feuz, P. Gervais, Fast multi-language lstm-
- 800

based online handwriting recognition, *International Journal on Document Analysis and Recognition (IJ DAR)* 23 (2020) 89–102.

- [51] M. Elleuch, R. Zouari, M. Kherallah, Feature extractor based deep method to enhance online arabic handwritten recognition system, in: *ICANN*, 2016, pp. 136–144.

805