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

Yahia Hamdi ¹ · Houcine Boubaker ¹ · Besma Rabhi ¹ · Wael Ouarda ¹
Bidyut B. Chaudhuri ² · Adel M. Alimi ¹

Received: date / Accepted: date

Abstract Deep learning approaches have proven successful in the areas of handwriting recognition. Despite this, research in this field is still needed, especially in multilingual online handwriting recognition context by adopting 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 pretreating 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 SOHTs feature vectors are fuzzy classified using DBLSTM 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 and complementarities 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 advances 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

Yahia Hamdi
E-mail: yahia.hamdi.tn@ieee.org

Houcine Boubaker
E-mail: boubaker_houcine@yahoo.fr

Besma Rabhi
E-mail: besma.rebhi.2015@ieee.org

Wael Ouarda
E-mail: wael.ouarda@ieee.org

Bidyut Baran Chaudhuri
E-mail: bbcisical@gmail.com

Adel M. Alimi
E-mail: adel.alimi@ieee.org

¹ University of Sfax, National Engineering School of Sfax, REGIMLab.: REsearch Groups in Intelligent Machines, LR11ES48, 3038, Sfax, Tunisia.

² Computer Vision & Pattern Recognition Unit, Indian Statistical Institute, Kolkata, India.

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 characters' category as their positions in the word especially, for Arabic script. This makes it difficult to directly adapt existing recognition systems for such tasks and invites to adopt new architecture based on deep learning and relevant complementary models' combination.

One of the most relevant criteria to succeed online recognition system is the choice of the best handwriting model. A wide number of designed handcrafted features such as structural, parametric, and global features are described in the literature to overcome specific problems like oclusions, 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 also 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 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 model in online handwriting trajectory modeling.

Lately, deep learning technologies [8] have attracted immense research works and industrial attention. These new categories of methods provide an alternative handwritten representation for end-to-end solutions. For instance, CNN is one of the most prominent deep learning methods which allows to automatically extract 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 seeing 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 based on DBLSTM and SVM. The main idea is to combine two different approaches for handwritten trajectory representation: the enhanced BEM which is characterized by a description of the dynamic and geometric aspect of online data modeling and the CNN as a powerful feature extraction model applied after trans-

forming 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 styles of writing that differ in the timeline of the *ink* plot 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 aims of enhancing the global system discriminating using three available datasets described later.

The remainder of this paper is organized as follows: In section 2, we present an outline of the most prominent related works. Section 3 provides a brief description of the supported language. 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 SOHTs pre-classification step. 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.

2 Related work

Many studies have been done on online handwriting recognition topics [1], [2]. Besides, it remains an active area of research because practicable 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, 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 expertise to design a representative features extractor. Various studies have been investigated in this context such as decomposition of the online signal into elementary stroke based on Beta-Elliptic model, characteristic strokes [3], or grapheme unit based on baseline detection algorithm like in [12]. Tangent differences and histograms of tangents are used as features to represent online Arabic characters [13]. A framework of online Arabic character recognition based

on statistical features without considering the variability and change of writing style of characters are investigated in [14]. Within the context, some studies implement algorithms of the conventional pattern classification which include decision trees [15], template matching [15], hidden Markov modeling (HMM) [17], neural networks [18], and support vector machines (SVM) [19]. More recently, Zitouni et al. [20] proposed a two-stage SVM classifier for online Arabic script recognition 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 [21]. It consists of extracting structural features using Freeman chain codes and elementary perceptual codes (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 has been greatly improved by using these models such as deep belief network (DBN) (e.g., [22]), convolutional neural networks (e.g., [23], [24]) which have been broadly applied for handwriting recognition. It is considered as a powerful tool for image classification. Some research work uses CNNs to online characters after converting the handwriting trajectory to image-like representations. Also, Yuan et al. [25] proposed word-level recognition for Latin script by employing CNN architecture. In this article, CNN is used for online handwritten document recognition after online word segmentation. Likewise, CNNs have been widely applied for handwriting Chinese text recognition [26]. It is considered as a powerful feature extraction model that generates automatically 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 [27]. Likewise, Ghosh et al. [28] perform cursive and non-cursive word recognition for both Devanagari and Bengali scripts based on LSTM and BLSTM recurrent networks. In this article, each word is divided into three horizontal zones (upper, middle, lower). Then, the middle zone is re-segmented into basic strokes in each structural and directional feature before carrying out training the LSTM and BLSTM. To improve the performance of deep BLSTM in online Arabic handwriting

recognition, Maalej et al. [29] employed three enhancement 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.

Combinations of these technologies have also been investigated. For instance, CNN with deep BLSTM (e.g., [30]) in which CNN is used to generate multiple features automatically from the scanned input handwritten sequence, while BLSTM has modeled frame dependency within the sequence. RNN with BLSTM (e.g., [31]) and BLSTM with gated recurrent unit (GRU) have been used for online Chinese character recognition and generation [32].

Further, several researches focus specifically on mono-language script, for instance, Arabic [13], Chinese [33], Tamil [34], Japanese [35] etc. Similarly, there exist multi-script and multi-language systems for online handwriting recognition like those presented in [35], [21] as well as other commercial systems such as those developed by Apple [25] and Microsoft [36].

3 Language specification

In this section, we briefly give 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, Kurdish, Urdu, and Pashto [30]. More than 420 million people of the world use the Arabic as a major language (UNESCO 2012) [35]. Arabic script is generally written cursive, most of these letters are connected to their neighbors. It generally contains 28 basic letters with other 10 marks that can change the meaning of the word. Also, there are 10 digits for usual digit recognition tasks. While there are 26 alphabetic letters for English, there are over 28 for in Arabic.

Table 1 Some Arabic characters in different forms

Characters	End	Isolated	Middle	Beginning
Alif (أ)	ا	آ		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

Arabic letters 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 (Beginning, Middle, Isolated, 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 (e.g., Alif 'ا', Raa 'ر'). Also, several letters have the same body (e.g., Baa 'ب', Thaa 'ث') but differ only in the number and position of the dots and diacritical marks which may be accidentally misplaced in handwriting and thus Arabic character recognition is more problematic than many other scripts. Further detail of Arabic characteristics and difficulties of writing are available in [37].

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. Depending on the writing style, Latin script can be written cursive or semi-cursive and the characters' shapes vary accordingly. In over of the basic $2 * 26$ characters (capital and minuscule shapes), it also supports accented characters. Unlike the Arabic script, the Latin script is written from left-to-right.

4 System overview

In this section, we present an outline of our proposed system of developing a multi-stage architecture based on a hybrid of DBLSTM recurrent neural networks and SVM for multilingual online handwriting recognition. Indeed, the original purpose of our work consists of how to adapt and exploit the effectiveness of BEM and CNN models in online multilingual handwriting recognition. As shown in Fig. 1, the architecture of the proposed system proceeds as follows: First, the input pen-tip trajectory (x, y) is denoised and normalized by a pre-processing module. 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: starting points (pen-down) and ending points (pen-up). Third, two types of feature vectors are extracted for each SOHT: the online hand-drawn trajectory features are extracted using BEM, and the generic features generated by the last CNN layer after converting the cursive trace to bitmap image. After that, SOHTs are fuzzy classified into k sub-class defined by k-means unsupervised algorithm in the training phase. The SOHT's fuzzy classification uses as input the extracted BEM and CNN feature vectors 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 enhance the global system discriminating power. In the following sections, we introduce each module in detail.

Fig. 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 correction and denoising steps to minimize handwriting variabilities, normalize the size, and reduce noise. Given the raw trajectory, the low-pass filtering Chebyshev type II with a cut-off frequency of $f_{cut} = 10\text{Hz}$ is used to mitigate the effect of noise and errors due to temporal and spatial quantification introduced by the acquisition system (see Fig.2). The value of cut-off frequency results from a compromise between the conservation of the handwriting undulations produced by young agile writers and the elimination of the physiological tremor of other older writers [38]. The horizontal fulcrums level of character handwriting decomposes their drawing area

in three zones namely upper, core, and lower regions respectively. Consequently, a procedure for normalizing the size of the handwriting is applied to adjust its height to a fixed value $h = 128$, while keeping the same ratio length/height [3], [4]. Both the preprocessing technique are used and tested for the supported scripts such as Arabic, Latin as well as digits. After denoising and

Fig. 2 (a) An acquired trajectory of Arabic character 'ي' and (b) after Low-pass filtering and smoothing.

eliminating the handwriting variation 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 Features extraction and SOHTs pre-classification

In this section, we describe the feature extraction method for the shape trajectory representation and SOHTs pre-classification step which serves to ameliorate 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 strategy 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 offline 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 [39] 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. 3.a), and an elliptic arc as illustrated in Fig. 3.b) modeling the static (geometric) profile for each stroke of the segmented trajectory. We describe in the following sub-section how the BEM process.

Fig. 3 Online handwriting modeling using BEM.

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 delimit and define 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 represented by the following expression:

$$pulse\beta(t, p, q, t_0, t_1) = \begin{cases} K \left(\frac{t-t_0}{t_c-t_0} \right)^p \left(\frac{t_1-t}{t_1-t_c} \right)^q & \text{if } t \in [t_0, t_1] \\ 0 & \text{if not} \end{cases} \quad (1)$$

with

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

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. 2, K called impulse amplitude, p and q are intermediate shape parameters.

$$V_\sigma(t) = \sum_{i=1}^n pulse\beta_i(k_i, t, p_i, q_i, t_{0i}, t_{1i}) \quad (3)$$

As described in Eq. 3, 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. 4.a), c), and e), respectively.

6.1.2 Trajectory modeling

In the space domain, many studies have been investigated for handwriting generation. In Bezine et al. [4], each stroke located between two points M_1 and M_2 is assimilated to an elliptic path verifying Eq. 4, where X and Y denote the cartesian coordinates along the elliptic stroke, a and b are respectively the small and large axis dimensions.

$$\frac{X^2 + Y^2}{a^2 + b^2} = 1 \quad (4)$$

Also, an elliptic arc is described by the tangent of the trajectory on their endpoints M_1 and M_2 which is invented by [40]. Indeed, each elementary beta stroke located between two successive extrema speed times can be modeled by an elliptic arc characterized by four geometric parameters such as: a , b , θ , θ_p as shown in Fig.3. b). There a and b represent respectively the half dimensions of the big and the small axes of the elliptic arc, θ is the angle of the ellipse major axis inclination, and θ_p denotes the trajectory tangent inclination at the minimum velocity endpoint. These parameters reflect the geometric properties of the end effector (pen or finger) trace, dragged by the set of muscles and joints involved in handwriting. Fig. 4.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 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 Post drawing CNN feature

In this sub-section, we briefly explain the use of CNN model in our context. Inspired by the recent successes of deep learning extracted features in different topics, we have employed CNN architecture in the off-line bitmaps

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
	θ	The ellipse major axis inclination angle.
	θ_p	Angle of inclination of the tangents at the endpoint M_2 .

reconstructed from the SOHTs database. The latter represents just the final layout of the hand drawing that skirts the chronologic style of its generation. Indeed, CNN features are extracted from the offline SOHTs by a convolution network containing multiple convolution and max-pooling layers. Also, we use batch normalization and dropout techniques to improve feature extraction. The details of the CNN module are described in the experimental section. The input image (32×32) is transformed into CNN-feature of size ($Bachsize, L, D$). Specifically, $Bachsize$ is fixed to 32, L and D are the lengths and the depth of the CNN entity, respectively. As shown in Fig. 5, 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

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 cursives and discrete styles. Thus, failing to manually assign a label to each SOHT, we chose to use the k-means unsupervised clustering algorithm for the SOHTs pre-classification in the training track (see Fig. 5, 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

Fig. 4 Online handwriting modeling of Arabic character 'ح', word 'علم' and Latin character 'a' with BEM. (a), (c) and (e) describe the velocity profile, (b), (d) and (f) represent the geometric profile.

accuracy of each classifier and consequently on the overall fusion recognition system.

7 Sequence Recognition

In our work, we introduced hybrid online and offline models based on DBLSTM for SOHTs fuzzy classifi-

cation and SVM for handwriting sequence (word, letters, digits, etc) recognition by melting decision. In this section we present different modules of our recognition process.

Fig. 5 SOHTs pre-classification and online handwriting script 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 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. [41] have introduced this LSTM network which is frequently used to learn longer-term dependencies and reduce the vanishing gradient problem [42].

As shown in Fig. 6, 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 these cells is performed by three multiplicative ‘gate’ units. At time-step t , there are input gate i_t , forget gate f_t , output

Fig. 6 LSTM memory block with one cell. (Graves et al. [8])

gate o_t , which are presented by the three equations Eq. 5 - Eq. 7.

$$i_t = \text{sigm}(W_i x_t + U_i h_{t-1} + b_i) \quad (5)$$

$$f_t = \text{sigm}(W_f x_t + U_f h_{t-1} + b_f) \quad (6)$$

$$O_t = \text{sigm}(W_o x_t + U_o h_{t-1} + b_o) \quad (7)$$

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

$$\tilde{c} = \text{tanh}(W_c x_t + U_c h_{t-1} + b_c) \quad (8)$$

$$c_t = i_t \circ \tilde{c} + f_t c_{t-1} \quad (9)$$

$$h_t = O_t \circ \text{tanh}(c_t) \quad (10)$$

i_t and f_t are used to control the updating of c_t (Eq.10) 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. 10. In various tasks, it is required to use both past and future complementary contexts instead of the RNN that only uses past contexts. To build a bidirectional LSTM model, we combine two LSTM sub-layers in both directions forward and backward [43]. 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 likelihood group C_j , $j=1 \dots K$, 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 $SOHT_i$, $i=1 \dots n$. 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. 5, 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 $\{P(X_{i_off}|C_1)P(X_{i_off}|C_2) \dots P(X_{i_off}|C_k)\}$ 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 performance of deep recurrent neural networks. The most commonly used regularization method is the dropout technique [43]. It is used in the training phase to avoid the over-fitting 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 [32]. 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 [44]. 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, italicity, smoothing, etc. These techniques broaden the training set, generating more training samples, and simultaneously bringing more variation into the training set.

7.3 SVM fusion engine for scripts recognition

After SOHTs fuzzy labeling, the outputs of DBLSTM of both online and offline sub-system are treated as input for the SVM classifier to make decision fusion. Indeed, SVM is considered a powerful tool for linear and nonlinear classification based on a supervised learning algorithm. It has shown high success in many practical applications such as pattern recognition. As shown in Fig.7.a), SVM is mostly used to determine an optimal separating hyper-plane by adopting a novel technique that maps the sample points into a high-dimensional feature space using a nonlinear transformation. It was originally designed to solve binary classification problems. However, it can be employed also to solve multi-class problems (see Fig.7.b) using several methods such as one-versus-all [45] based on dot product functions (kernel functions) in feature space. Among these functions, we can find linear, Radial Basis Function (RBF), sigmoid, etc. In our context, we use RBF function as a kernel in the hybrid model which is defined as:

$$k(x, x') = \phi(x)\phi(x') = \exp(-\gamma\|x - x'\|^2) \quad (11)$$

For the recognition of the overall handwritten script, the SOHTs vectors 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.

Fig. 7 Principle of SVM. (a) two-class hyper-plane example, (b) one-versus-all method

8 Experiments and results

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 incorrectly Error Rate (WER) were employed to evaluate the proposed system defined as 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 present system in the error analysis section. All tests were implemented using core i7 processor of 3.2 GHz, 8 GB of RAM.

8.1 Datasets overview

One of the most difficult attitudes 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 performance and efficiency of our proposed system, we have used three datasets: Online-KHATT [46] for isolated Arabic characters, ADAB [47] for Arabic words, and UNIPEN [48] for Latin characters and digits. In this section, we describe these publicly available datasets.

8.1.1 Online-KHATT dataset

Online-KHATT is a new open-vocabulary Arabic dataset proposed by [14]. It is composed of 10,040 lines of Arabic online text taken from 40 books, which is 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 experiment, we use a subset of the segmented characters of this dataset [37] that contain 44,795 data after using data augmentation technique. The subset is partitioned into a training set with 30,795 samples, a validation set of 4,000 samples and 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 found, which makes in total 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 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 literature. It consists of more than 33,000 Arabic words handwritten by 170 different writers. The text contains 937 Tunisian town/village names. As shown in Table 4, this database is divided into six distinct sets which were collected for the ICDAR 2011 online Arabic handwriting recognition competition [47].

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 multi-lingual scripts using hybrid DBLSTM

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

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 of choosing the value K that corresponds 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 using the extracted CNN parameters by applying the k-means clustering algorithm. The value K is fixed empirically after several tests to 210, which represents the number of groups that return the least within-cluster sums of 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.

8.2.1 Structure of CNN feature extraction

The second experiment consists of selecting the best CNN feature. We tried several parameters 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 in each layer. We chose CNN3 with 13 convolution layers and 4 Max-Pooling layers similar to Zhang et al. [32] 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 dimension of feature maps in each convolution 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. We train the network using its parameters: a stochastic gradient descent ‘sgdm’ with momentum and mini-batch of size 100, the learning rate was 0.001. We use also the dropout technique with 0.2 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 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 is 400 nodes as described in the experimental results section. The output layer size is defined by the number of SOHTs class. Dropout is used in fully connected layer with a probability of 0.3. We train the network using its parameters: a stochastic gradient descent ‘sgdm’ with momentum and mini-batch of size 200. We also start the training with an initial learning rate of 0.001 and 400 maximum number of epochs. A categorical cross-entropy loss function is utilized to optimize the network. After each epoch, we shuffle the training data 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 [49]. 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. 8 and Table 6, we conduct further study of the number of layers and nodes per layer for both BEM and CNN features to determine the optimal size of the BLSTM network. We observe that for the two input features, the use of 3 layers outperforms more shallow networks, and the addition of more layers brings almost no improvement. Further, using 256 nodes per layer is sufficient, as larger networks give only limited improvements, if at all.

We can see a significant improvement in the CER using CNN than BEM model which is explained by the useful offline features generated by CNN model allowing

Table 5 CNN configuration: Conv means Conv, Norm-ReLU layer. The parameters of the convolution and max-pooling layers are represented as “Conv: filter size (number of filters)” and “Max pool: filter size”, respectively.

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

Fig. 8 CER of DBLSTM models trained 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.

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	7.13	7.56
	5 Layers	8.29	7.25	7.84
CNN	1 Layer	8.62	6.75	7.14
	3 Layers	7.13	6.05	6.35
	5 Layers	7.16	6.19	6.84

a fair compromise between character class discrimination 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	5.25	6.39
ADAB	2.65	1.44	1.28	2.18

Moreover, we compare our system to others described in the literature that also performed on online Arabic character recognition. The results are summarized in Table 8 which present for each work: the used classifier, feature extractor model, and its accuracy. We can see that the performance achieved by our system is better than the two already mentioned systems which are trained on the same training dataset. We observe that the obtained results on online-KHATT dataset are significantly better than those found by [14] and [53]. This is due to the use of the complementarity models for handwriting representation on the one hand which is described by the simultaneously 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 combination models allow the enhancement of the global system discriminating power.

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 comparison results with other previous studies using the same database. We observe that the results achieved by our

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

System	Feature	CER (%)
ANN [14]	Statistical features + chain codes	18.00
LSTM [53]	Path signature feature	7.43
Our system	BEM + CNN	5.25
DBLSTM-SVM		

system using the combination of the two models are better than the ‘writer-independent’ experiment reported by [51] that uses cluster generative statistical dynamic time warping (CSDTW). This indicated that our proposed method is also robust and very promising for on-line handwriting Latin characters and digits. Again, we refer to more recent work [52] 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			[52]	[51]
	BEM	CNN	Fusion		
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

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 9, 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.

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.

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 [53], [17] by the integration of multi-stage deep learning networks for the analysis and the

Fig. 9 WER of DBLSTM models trained on the ADAB dataset with different 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	4.70	5.43
	5 Layers	7.19	5.15	6.74
CNN	1 Layer	6.52	4.75	5.12
	3 Layers	5.16	3.13	4.46
	5 Layers	5.19	4.25	4.94

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 [53] 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 [53]	BEM + CDBN	2.49
HMM [17]	Directional features + Offline features	2.22
MLP/HMM [19]	BEM features	3.55
SVM [20]	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. 10 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 correspondent handwritten proposition.

We can identify some samples that are not recognized by one of the using 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 handle 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 remain to be improved due to the great diversity of writing styles and the similarity between many characters. 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.

Fig. 10 Some corrected and misclassified samples of Arabic, Latin characters, and digits.

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 'ج', and

middle 'ج' which can be explained by the wrong identification of the delayed stroke. Other examples are not recognized by our system of 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.

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 use the extracted features to make the classification and compare the performance of single classifiers using DBLSTM. We proceed also by combining the obtained results using SVM to enhance the global system discriminating power.

The efficiency 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 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 observed that the used features extraction models are rather generic and their applicability in the other scripts such as Persian, Chinese, English, etc., is interesting and we envisage as future work. Again, we schedule to accomplish our method for text recognition.

References

1. Alimi, A. M.: An evolutionary neuro-fuzzy approach to recognize online Arabic handwriting, in: Document Analysis and Recognition, 1997., Proceedings of the Fourth International Conference on, IEEE. pp. 382–386, (1997)
2. Cho, S.J., Kim., J.H., (2004). Bayesian network modeling of strokes and their relationships for on-line handwriting recognition. Pattern Recognition 37 (2), 253–264.
3. Boubaker, H., Chaabouni, A. Tagougui, N., Kherallah M. and Adel M. Alimi, (2013). "Handwriting and Hand drawing Velocity Modeling by Superposing Beta Impulses and Continuous Training Component," International Journal of Computer Science Issues (IJCSI), Vol. 10, Issue 5, pp 57 – 63.

4. Bezine, H., Adel M. Alimi, S. Nasser, (2004). "Generation and analysis of handwriting script with the beta-elliptic model", In the proceeding of the 9th International Workshop on Frontiers in Handwriting Recognition. IWFHR-9, pp. 515 – 520.
5. Dhieb, T., Rezzoug, N., Boubaker, H., Gorce, P., and A. M. Alimi, (2019) "Effect of age on hand drawing movement kinematics," Computer Methods in Biomechanics and Biomedical Engineering, Vol. 22 Issue sup1: 44th Congress of the Société de Biomécanique, DOI: 10.1080/10255842.2020.1714235.
6. Dhieb, T., Ouarda, W., Boubaker, H., and A. M. Alimi, (2016). "Beta-Elliptic Model for Writer Identification from Online Arabic Handwriting", J. of Information Assurance and Security, ISSN: 1554-1010, Volume 11, Issue 5, pp. 263-272.
7. San, L., T. Su, C. Liu, and R. Wang, (2015). Deep LSTM Networks for Online Chinese Handwriting, Recognition in: Document Analysis and Recognition (ICDAR), 2015, pp. 1304–1308.
8. Graves, A.: (2014). Generating Sequences with Recurrent Neural Networks, arXiv:1308.0850, 5.
9. Graves, A., Liwicki, M., Bunke, H., Schmidhuber, J., Fernández, S. (2008): Unconstrained on-line handwriting recognition with recurrent neural networks. In: Advances in Neural Information Processing Systems, pp. 577–584.
10. Hamdi, Y., Boubaker H., Dhieb., T., Elbaati, A, Alimi, A. (2019). Hybrid DBLSTM-SVM based Beta-elliptic-CNN Models for Online Arabic Characters Recognition. In International Conference on Document Analysis and Recognition (ICDAR), pages 803-808.
11. Boubaker, H., Tagougui, N., Haikal ElAbed, Monji Kherallah and Adel M. Alimi, (2014) "Graphemes Segmentation for Arabic On-line Handwriting Modelling," Journal of Information Processing System (JIPS), Vol. 10, No. 4, pp 503 –522.
12. Eraqi, H., Azeem, S.A., (2011). An on-line Arabic handwriting recognition system: Based on a new on-line graphemes segmentation technique, in: Document Analysis and Recognition (ICDAR), International Conference on, IEEE. pp. 409–413.
13. Mezghani, N., Mitiche, A., and Cheriet, M. (July 2008). "Bayes Classification of Online Arabic Characters by Gibbs Modeling of Class Conditional Densities," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 30, Issue 7, pp. 1121-1131.
14. Al-Helali B. M., and Mahmoud, S. A. (2016) "A Statistical Framework for Online Arabic Character Recognition," Cybernetics and Systems, vol. 47, no. 6, pp. 478–498.
15. Akouaydi, H., Abdelhedi, S., Njah, S., Zaied, M., Adel M. Alimi: (2017). "Decision trees based on perceptual codes for on-line Arabic character recognition". In International Workshop on Arabic Script Analysis and Recognition (ASAR): 153-15.
16. El-wakil, M.S., and Shoukry, A.A. (1989) "On-Line Recognition of Handwritten Arabic Character Recognition," Pattern Recognition, vol. 22, no. 2, pp. 97 105.
17. Abdelaziz, I., and Abdou, S., (2014). "AltecOnDB: A Large-Vocabulary Arabic Online Handwriting Recognition Database", arXiv preprint arXiv: 1412.7626.
18. Kubatur, S., Sid-Ahmed, M., Ahmadi, M. (2012): A neural network approach to online devanagari handwritten character recognition. In: International Conference on High Performance Computing and Simulation (HPCS).
19. Tagougui, N., Kherallah, M., (March 2017). "Recognizing online Arabic handwritten characters using a deep architecture", Proc. SPIE 10341, Ninth International Conference on Machine Vision, 17.
20. Zitouni, R., Hala, B., Najet, A., (2020). Online Handwritten Arabic Scripts Recognition Using Stroke-Based Class Labelling Scheme. International Journal of Computational Intelligence Systems, ISSN: 1875-6891, DOI: <https://doi.org/10.2991/ijcis.d.201024.001>
21. Zouari R., Boubaker H., Kherallah M. (2020). Towards Online Handwriting Recognition System Based on Reinforcement Learning Theory. In: Neural Information Processing, (ICONIP), vol1332,doi:https://doi.org/10.1007/978-3-030-63820-7_64.
22. UNESCO. (2012). World Arabic language day. UNESCO: www.unesco.org/new/en/unesco/events/prizes-and-celebrations/celebrations/international-days/world-arabic-language-day/ (accessed July 22, 20).
23. Zhou, X., Yu, J., Liu, C., Nagasaki, T., and Marukawa, K., (2007) "Online handwritten Japanese character string recognition incorporating geometric context," in Proc. Int. Conf. Doc. Anal. Recognition., pp. 48–52.
24. Altwaijry, N., Al-Turaiki, I. (2020). Arabic handwriting recognition system using convolutional neural network. Neural Comput and Applic, doi: <https://doi.org/10.1007/s00521-020-05070-8>.
25. Yongqiang, Y., Xiangwei, Z., Bin, H., Yuang, Z., Xinchun, C. (2021). EEG emotion recognition using fusion model of graph convolutional neural networks and LSTM. Applied Soft Computing, Volume 100, 06954, doi.org/10.1016/j.asoc.2020.106954.
26. Zhuo Chen, Fei Yin, Xu-Yao Zhang, Qing Yang, Cheng-Lin Liu, (2020), Multilingual Handwritten Text Recognition via Multi-task Learning of Recurrent Neural Networks, Pattern Recognition, 108: 107555.
27. Wu, Y. C., Yin, F., Liu, C. L: (2017) "Improving handwritten Chinese text recognition using neural network language models and convolutional neural network shape models," Pattern Recognition, Vol. 65, pp.251-264.
28. Ghosh, R., Chirumavila, V., Prabhat, K.: (2019)" RNN Based Online Handwritten Word Recognition in Devanagari and Bengali Scripts using Horizontal Zoning," Pattern Recognition, doi.org/10.1016/j.patcog.2019.03.030.
29. Maalej, R., Kherallah, M., (2020). Improving the DBLSTM for on-line Arabic handwriting recognition, Multimedia Tools and Applications, 1-22.
30. Schuster, M., and Paliwal, K., (1997) "Bidirectional recurrent neural networks," IEEE Trans. Signal Processing, vol. 45, no. 11, pp. 2673–2681.
31. Yaeger, L., B. Webb, and R. Lyon, (1998) "Combining neural networks and context driven search for on-line, printed handwriting recognition in the Newton," in AAAI AI Magazine. Berlin, Germany, Springer.
32. Zhang, X.-Y. Bengio, Y. C.-L. Liu., (2017) Online and offline handwritten chinese character recognition: A comprehensive study and new benchmark, Pattern Recognition 61, 348–360,
33. Sen, S., Shao D., Paul S., Sarkar R., Roy K.: (2018) Online Handwritten Bangla Character Recognition Using CNN: A Deep Learning Approach. In: Advances in Intelligent Systems and Computing, pp 413-420.
34. Bharath, A., and Madhvanath, S., (2007) "Hidden Markov models for online handwritten Tamil word recognition," in Proc. Int. Conf. Doc. Anal. Recognition., pp. 506–510.

35. Victor C., Pedro G., Thomas D., Henry A. Rowley, Alexander D., Marcos C., Li-Lun W., Daniel K., Sandro F., Philippe G., (2020) "Fast multi-language LSTM-based online handwriting recognition," *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 23, no. 6, pp. :89–102.
36. Pittman, J. A., (2007) "Handwriting recognition: Tablet PC text input," *IEEE Comput.*, vol. 40, no. 9, pp. 49–54, Sep.
37. Al-Helali, and S. A. Mahmoud. (2016). Arabic Online Text Recognition (AOTR): A survey. Submitted for publication.
38. Physiological and pathological tremors and rhythmic central motor control (2000) " J. H. McAuley, C. D. Marsden *Brain*, Volume 123, Issue 8, Pages 1545–1567;
39. Plamondon, R. (1995). A kinematics theory of rapid human movements. Part I: Movement representation and generation. *Biological Cybernet.* 72, 295–307.
40. 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.
41. Hochreiter, S., Schmidhuber, J., (1997). Long short-term memory. *Neural Comput* 9(8):1735–1780.
42. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014) "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958.
43. Swethalakshmi, H., Jayaram, A., Chakraborty, V.S., Sekhar, C.C: Online handwritten character recognition of devanagari and telugu characters using support vector machines. In: *Proc. 10th IWFHR*, pp. 367–372.
44. Galen, G.V., Weber, J., (1998) On-line size control in handwriting demonstrates the continuous nature of motor programs. *Acta Psychol.* 100, pp.195–216.
45. Burges, C.J. A tutorial on support vector machines for pattern recognition. *Data Min Knowl Disc* 2(2):121–167.
46. Mahmoud, S., Luqman, H., Al-Helali, B.M., Bin-Makhashen, G., MT Parvez: (2018) "Online-KHATT: An Open-Vocabulary Database for Arabic OnlineText Processing". *The Open Cybernetics & Systemics Journal*, vol. 12, pp. 42–59.
47. Boubaker, H., Elbaati, A., Tagougui, N., Kherallah, M., H., El Abed, and A.M., Alimi, (2012). "Online Arabic Databases and Applications," Book chapter in: Margner, V., El Abed, H. (eds) "Guide to OCR for Arabic Scripts," Chp. Part IV pp S41–S57, Springer.
48. Guyon, I., Schomaker, L.R.B., Plamondon, R., Liberman, M., and S. Janet, "UNIPEN Project of On-Line Data Exchange and Recognizer Benchmarks," *Proc. 12th Int'l Conf.*
49. Onnia, V., Tico, M., and Saarinen, J., (2001) "Feature Selection Method Using Neural Network," *Proc. Int'l Conf. Image Processing*, vol. 1, pp. 513–516.
50. Wilson-Nunn, D., Lyons, T., Papavasiliou, A., Ni, H. (2018): "A Path Signature Approach to Online Arabic Handwriting Recognition ", *International Workshop on Arabic and Derived Script Analysis and Recognition (ASAR)*: 135–139.
51. Bahlmann, C., and Burkhardt, H., (2004) "The writer independent online handwriting recognition system frog on hand and cluster generative statistical dynamic time warping," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 3, pp. 299–310.
52. Namboodiri AM, Jain AK (2004) Online handwritten script recognition. *IEEE Trans Pattern Anal Mach Intell* 26(1):124–130.
53. Elleuch, M., Ramzi Z., and Monji K., (2016) "Feature extractor based deep method to enhance online arabic handwritten recognition system." *International Conference on Artificial Neural Networks*. Springer, Cham.