RNN-LSTM Based Beta-Elliptic Model for Online Handwriting Script Identification

Ramzi Zouari[1], Houcine Boubaker[1], and Monji Kherallah[2]
[1]National School of Engineers of Sfax, University of Sfax, Tunisia
[2]Faculty of Sciences of Sfax, University of Sfax, Tunisia

Abstract: *Recurrent Neural Network (RNN) has achieved the state-of-the-art performance in a wide range of applications dealing with sequential input data. In this context, the proposed system aims to classify the online handwriting scripts based on their labelled pseudo-words. To avoid the vanishing gradient problem, we have used a variant of recurrent network with Long Short-Term Memory. The representation of the sequential aspect of the data is done through the beta-elliptic model. It allows extracting the dynamics and kinematics profiles of different strokes constituting a script over the time. This system was assessed with a large vocabulary containing scripts from ADAB, UNIPEN and PENDIGIT databases. The experiments results show the effectiveness of the proposed system which reached a high recognition rate with only one recurrent layer and using the dropout technique.*

Keywords: *Online, pseudo, stroke, velocity, beta-elliptic, recurrent, dropout.*

Received February 15, 2018; accepted April 20, 2018

1. Introduction

Handwriting analysis still remains a challenge in the field of document analysis. Thanks to the technological revolution in touch-capturing devices, we have a very rapid growth of handwritten documents which are more difficult to treat than Optical Character Recognition (OCR) documents. This is due to the cursive style of the handwriting that contains ligatures, valleys and delayed strokes [10].

Works which deal with handwritten scripts are categorized into two branches. The first is off-line, where only a scanned image of the handwriting is available, whereas the second is online, where temporal and spatial information's about the writing are available.Several approaches were made in the topic of handwritten document analysis. They are interested in handwriting recognition, signature verification and writer identification [3, 27, 28]. Recently, multiple researches deal with script identification. Their purpose consists on identifying some characteristics, especially the language and the type (printed or handwritten).

In this paper, we propose a new method for online script recognition in order to distinguish between three classes of handwriting: Arabic words, Latin words and digits. Our method is based on the association of the beta-elliptic model with a Recurrent Neural Network (RNN) with Long Short Term Memory (LSTM). The beta-elliptic model operates on the dynamic profile of the trajectory. This is why it is suitable for different forms of handwriting. In addition, RNN is a powerful machine learning model that handles with sequential data and has shown very promising results.

This paper is structured as follows. In section 2 relevant previous works are overviewed. Section 3 presents the application of the-beta-elliptic model with RNN for script identification. Section 4 presents experiments and results. Finally, we present the conclusion with some future works.

2. Related Works

Script and language identification is the first step in document automation process. In literature, several approaches were proposed for printed and handwritten documents. These approaches are generally classified into two categories: holistic and analytic. For holistic approaches, the identification process is done without segmentation. Among these approaches, we can cite the system of Hasan *et al*. [7] who presents a novel methodology for multiple script identification that works on OCR documents. Thus, a sliding window of 1-pixel width, called frame, traverses the input text-line image and convert it into a one dimensional vector representing the raw pixel values.

For the training process, each frame is fed to the RNN that learns to classify it into one of the target classes. This approach was assessed on the synthetic English-Greek database and has shown encouraging results that exceeds the rate of 98 percent.

In other work, Kulkarni *et al*. [12] extracted visual features obtained by scanning the text along vertical and horizontal directions. This method was applied on Kannada, English and Hindi scripts with Probabilistic Neural Network (PNN) and achieved an average accuracy of 95 percent. Script identification was strongly applied for offline handwritten document.

Saidani *et al*. [21] applied respectively Histogram of Oriented Gradients (HOG) and Co-occurrence Matrix of Oriented Gradients (Co-MOG) methods for Arabic and scripts image words.

These methods are robust to varied image sizes and different writing styles. Thereafter, Genetic Algorithm (GA) and Principal Component Analysis (PCA) were applied for reducing the dimensionality of the feature vectors. For the identification stage, they used the K-Nearest Neighbor classifier (KNN) that provides good results in comparison to the others classifiers based on Support Vector Machines (SVM) and Gaussian Mixture Models (GMM). In other work [20], they combined few well-established features previously used in the literature with new structural characteristics as Loop position, presence of bottom diacritics and elongate descenders for Arabic and Latin scripts.

For the test, they used the cross validation method to divide the words into non-overlapping sets. Experiments have been carried using *IAM* and *IFN/ENIT* databases and provide better results compared to the work of Haboubi *et al*. [6] where texture descriptors are used. GMM based approach has been presented in [15]. In fact, a fixed-length sliding window traverse the word image from right to left and extract a set of 35 geometric features who distribution is captured by GMMs through the training procedure. This approach was evaluated using some parts of three public databases from Arabic and French scripts and reached average identification accuracy about 98.92%.

Besides works based on holistic approaches, some researchers also proposed systems based on analytic methods where the input signal is segmented into sub-units according to significant points and the handwritten is viewed as a sequence of feature vectors. For OCR documents, Singh and Lawahar [23] proposed a system that consists on dividing horizontally a word into upper and lower parts. For each one, a sliding window operates over the sub-word and extracted a set of twelve geometric features from a single window.

The sequence of feature vectors is then given as input to the RNN to classify the word into its corresponding language. Experiments were performed on a large corpus of 15.3 M words belonging to 15 scripts and show very promising results that exceed those obtained with Gabor features and SVM classifier. Analytic approaches that treat the handwriting scripts were carried. Singh *et al*. [24] proposed a system in which a handwritten word was firstly surrounded by a rectangular bounding box. Inside this box, a combination of maximum, sectional and concentric inscribed ellipses were done to extract elliptical parameters from different regions of the word image. In other hand, the word curve is fitted using polygonal approximation method. However, a word is composed of vertex points linked by linear segments.

The combination of elliptical and polygonal features has been evaluated on a dataset of 7000 words from Indian and roman scripts and attained an accuracy rate of 95% for 5-fold cross validation of the Multi-Layer Perceptron (MLP) with epoch size of 1500. Compared to previous works, this proposed method outperforms systems based on Gabor Filter, convex hull and texture shapes. Saidani and Echi [19] proposed a pyramid representation of HOG parameters. The principle consists on segmenting the word image into spatial grids where the number of divisions is doubled iteratively in each axis direction.

Indeed, the number of points for a given cell at one level is simply the sum over those contained in the four cells it is divided into at the next level, thus forming a pyramid representation. Experiments were carried on two public databases *IAM* and *IFN/ENIT* for Latin and Arabic words respectively and reached an error rate less than 2% with Bayes classifier. The obtained results outperforms the system proposed in [18] where HOG features are decoded using HMM.

According to the literature, the majority of systems are based on OCR and offline script identification, but there are only few reports about online script identification despite the emergence of data entry devices. Namboodiri and Jain [17] addressed the problem of script identification for online handwriting text. Their approach allows recognizing six major scripts.

Firstly, they applied the Gaussian filter on handwritten trajectory coordinates to extract strokes with equidistant points. Thereafter, they extracted spatial features from different strokes as direction, density and length. The identification is done using SVM and attained an overall accuracy of 87.1 percent on a set of 13400 words. Tan *et al*. [25] proposed a system for identifying Arabic, Roman and Tamil scripts. They used the same methods of pre-processing and features extraction described in [17] to obtain groups of prototypes using the k-means algorithm. Thereafter, the distributions frequencies of extracted vectors from the test set are computed to map these vectors to the previous prototypes based on the *tf-idf* measurement.

The last stage consists on comparing the distribution of *tf-idf* vectors to that of the three script families. So, the test document is assigned to the family with the minimum Chi-square distance. Table 1 summarizes methods presented for script identification.

Table 1. Script identification methods.

	Reference	Type of document	Approach	Classifier	Scripts	Identification rate
Holistic Approach	[7]	OCR	- Sliding window - raw pixels values	RNN	- English - Greek	98.19%
	[12]		Visual features	PNN	- Kannada - English - Hindi	95%
	[21]	Offline	- HOG - Co-MOG	KNN	- Arabic - Latin	99.85%
	[20]		Structural features	Bayes classifier	- Arabic - Latin	98.4%
	[15]		Geometric features	GMM	- Arabic - Latin	99.1%
Analytic Approach	[23]	OCR	- Sliding window - Geometric features	RNN	15 scripts	97.16%
	[24]	Offline	Elliptical and polygon parameters	MLP	- Indian - Roman	95.35%
	[19]		Pyramid HOG	Bayes classifier	- Arabic - Latin	98.3%
	[17]	Online	Spatial features	SVM	6 scripts	95.5%
	[25]		Spatial features	TF-IDF	- Arabic - Roman - Tamil	93.3%

3. System Overview

The beta-elliptic model has not yet used for online script identification. Since it has strongly applied for online handwriting recognition [26] and writer identification [4], we propose to use it for script identification. Furthermore, beta-elliptic model is suitable for any type of handwriting script because it operates especially on the dynamic profile of the trajectory. In our system, a script is divided into continuous handwriting trajectory, called pseudo-word, which represents the interval between pen-down and pen-up moments. Then, the beta-elliptic model is executed at pseudo-word level.

3.1. Beta-Elliptic Model

Handwriting movement involves the mobilization of N neuromuscular sub-systems [11]. In velocity domain, these excitations can be approximated by the sum of overlapped beta impulses. In static domain, the trajectory portion delimited between two successive sub-systems S_i and S_{i+1} is modeled by an elliptic arc

3.1.1. Velocity Profile Modeling

Form the trajectory coordinates (X,Y), the curvilinear velocity $V_\sigma(t)$ is obtained using a second-order derivative filter with finite impulse response. The velocity curve $V_\sigma(t)$ shows a signal that alternate a successive extrema of velocity where each stroke converge to the beta impulse [1]. Then, the overall signal is approximated by an algebraic addition of overlapped beta impulses (Equation 2).

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

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

With:

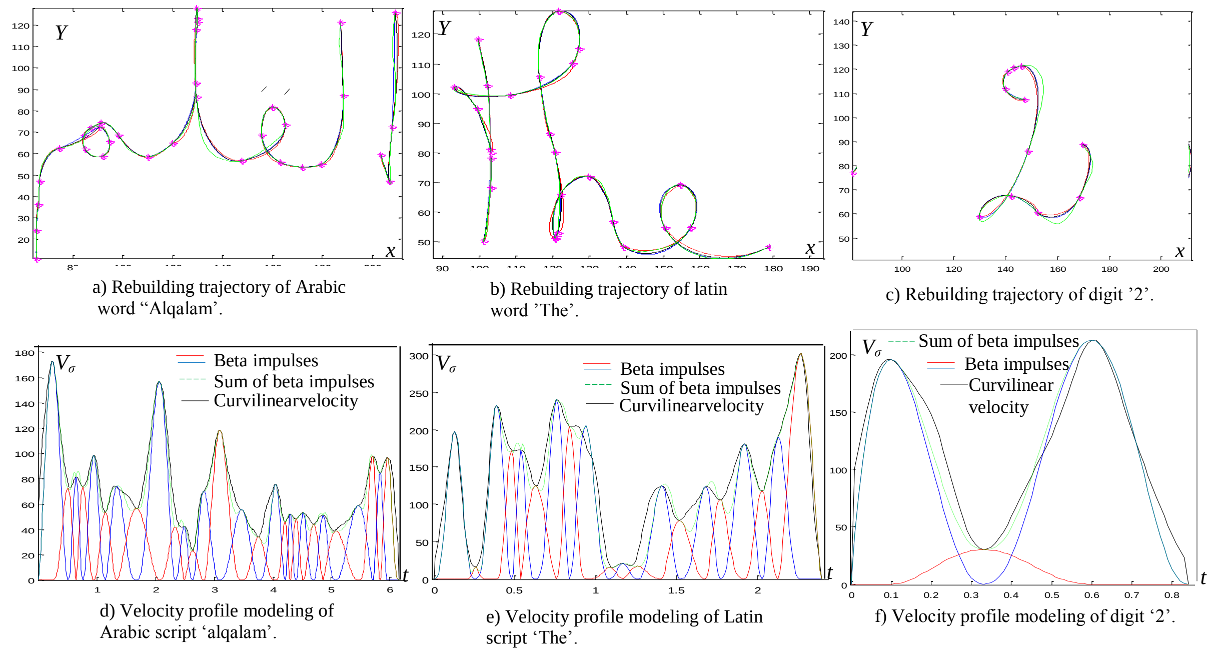
$$\beta_i\left(t,q_i,p_i,t_{oi},t_{1i}\right)=\begin{Bmatrix}\left(\frac{t-t_{0i}}{t_{ci}-t_{0i}}\right)^{pi}\left(\frac{t_{1i}-t}{t_{1i}-t_{ci}}\right)^{qi} & if\ t\in[t_{0i},t_{1i}]\\ 0\ \ elsewhere\end{Bmatrix} \tag{3}$$

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

Where t_{0i}, t_{ci}, t_{1i} are the moments witch correspond respectively to the start, the maximum amplitude and the end of the Beta function ($t_{0i}<t_{ci}<t_{1i}$), K_i represents the amplitude of i^{th} beta impulse and p_i, q_i are intermediate parameters. The Figures 1-e, 1-f and 1-g illustrate the application of the overlapped beta impulses theory for Arabic, Latin and digit scripts respectively. The black curves represent the curvilinear velocity signal $V_\sigma(t)$, red and blue curves represent the overlapped beta impulses. According to these figures, it is clear that the sum of beta impulses (green dotted curves) is almost similar to the curvilinear velocity.

3.1.2. Geometric Profile Modeling:

In the space domain, each stroke located between two successive extrema speed times can be assimilated to an elliptic arc characterized by four parameters (a, b, θ and θ_p).The parameters a, b are respectively the half dimensions of the large and the small axis of the elliptic arc. θ is the angle of the ellipse major axis inclination and θ_p is the angle of inclination of the tangents at the stroke endpoint M_2.In the Figures 1-a,1-b and 1-c the black curves represent the original handwriting trajectory while the red and the green Curves present the rebuilding trajectories fitted by elliptic arcs. Thus, a stroke is modeled with a vector of nine parameters as shown in Table 2.



a) Rebuilding trajectory of Arabic word "Alqalam". b) Rebuilding trajectory of latin word 'The'. c) Rebuilding trajectory of digit '2'.

d) Velocity profile modeling of Arabic script 'alqalam'. e) Velocity profile modeling of Latin script 'The'. f) Velocity profile modeling of digit '2'.

Figure 1. Beta-elliptic modeling for online Arabic, Latin and digit handwritten scripts.

Table 2. Beta-elliptic parameters.

	Parameters	Explanation
Dynamic parameters	K	Beta impulse amplitude
	$\Delta t = (t1 - t0)$	Beta impulse duration
	$Rap = \frac{p}{p+q}$	Rapport of beta impulse asymmetry or culminating time
	P	Beta shape parameters
	$\frac{K_i}{K_{i+1}}$	Rapport of successive Beta impulse amplitude
Geometric parameters	a	Ellipse major axis half length
	b	Ellipse small axis half length
	θ	Ellipse major axis inclination angle
	θ_p	Angle of inclination of the tangents at the stroke endpoint M_2

3.2. Word embedding technique

Word Embedding is a natural language modeling technique. It is used to map words or phrases from a vocabulary to a corresponding vector of real numbers. Methods to generate this mapping include networks, co and Word2Vecmodels [2]. In our study case, each word is seen as a set of its pseudo-words. Since no pseudo-word vocabulary already exists, we have built a vocabulary from data base samples.

The method consists on clustering the pseudo-words into *k* groups according to their beta-elliptic parameters using an unsupervised classifier that is *k*-means [13]. Thus, a pseudo-word will not be modeled by its beta-elliptic parameters but rather with its vocabulary index. With word embedding, a pseudo-word input is seen as a "one-hot" encoding vector i.e., a vector of zeroes with 1 at a single position. Figure 2 shows the decomposition of the Arabic word *'RASOUL'* on its pseudo-words and the application of word embedding technique with vocabulary size equal to 7. The indexes of pseudo-words are respectively 2, 5, and 7. Thanks to this representation, word embedding allows projecting high dimensional sparse vector into a low dimensional dense vector for word representation. It was successfully used for a variety of natural language tasks [14].

$$Word = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Figure 2. Word embedding representation.

3.3. Recurrent Neural Network

RNN is a powerful architecture for sequential data thanks to its internal states. It has demonstrated great success in sequence labeling and prediction tasks such as handwriting recognition, speech recognition and language modeling [5]. RNN is trained by stochastic gradient descent using Back Propagation through Time algorithm (BPTT). A major problem with gradient descent for standard RNN architectures is that error gradients vanish exponentially quickly with long-term dependencies, due to what is called the vanishing/exploding gradient problem.

To address this drawback, Hochreiter and Schmidhuber [8], have designed a Long Short Term Memory (LSTM) network which can in principle store and retrieve information over long time periods. LSTM explicitly designs a memory block inside a hidden node that has the following ingredients: memory cell which stores information about the past and input C_t, output, and forget gates that control the flow of information within and among the memory blocks.

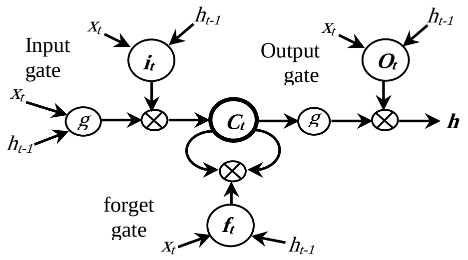


Figure 3. LSTM Memory block.

Figure3 shows the structure of a single LSTM memory block, which replaces a simple hidden node used in ordinary RNNs. LSTM network recurrently applies the following series of equations to obtain the sequence of hidden node outputs, h= $(h_1, h_2, \ldots, h_T)$, $h_t \in R^m$:

$$i_t=\sigma(W_{xi}x_t+W_{hi}\, h_{t-1}+W_{Ci}\, C_{t-1}+b_i) \quad (5)$$

$$f_t=\sigma(W_{xf}x_t+W_{hf}\, h_{t-1}+W_{Cf}\, C_{t-1}+b_f \quad (6)$$

$$C_t=f_t\, C_{t-1} +i_t tanh(W_{xc}x_t+W_{hc}\, h_{t-1} +b_c) \quad (7)$$

$$O_t=\sigma(W_{xo}x_t+Who\, h_{t-1}+W_{Co}\, C_{t-1}+b_o) \quad (8)$$

$$h_t=O_t tanh(C_t) \quad (9)$$

Where the symbols *i*, *f*, *o* and *c*, respectively, stand for the input gate, forget gate, output gate, and memory cell state vector. W denotes weight matrices, the b terms denote bias vectors and σ is the logistic sigmoid function. Note that for the gates, there are not only the recurrent connections from the hidden node outputs from the previous time stamp, but also the peephole connections from the cell states. With explicitly designed cell and gate structures as above, LSTM learns *W* and *b* from the training data so that it can determine when to receive input signals to the cell, output the hidden node activations from the memory blocks, and reset the cell states to refresh the memory [22].

In addition to LSTM blocks, another method can be applied in order to prevent RNN from over fitting during training stage, called Dropout. It is a popular regularization technique for the feed-forward neural networks where some network units are randomly masked during training. This method was typically applied only at non-recurrent layers. Recently, it is applied at the recurrent layers [16], i.e., dropout is applied on the output at step *t* before it is used to compute the output at step *t+1*.

In our proposed system, the input layer contains the beta-elliptic parameters of pseudo-words for a given script. Thereafter, these features are connected to the word embedding layer in order to obtain a low-dimensional dense vector for word representation. LSTM blocks are used in the recurrent layer and the dropout technique is applied both in recurrent and fully-connected layers. Since our system became able to classify scripts by selecting the most probably labelling, we add a LogSoftmax output layer (Figure 4). The circles with crosses denote the randomly omitted hidden nodes during training, and the dotted arrows stand for the model weights connected to those omitted nodes.

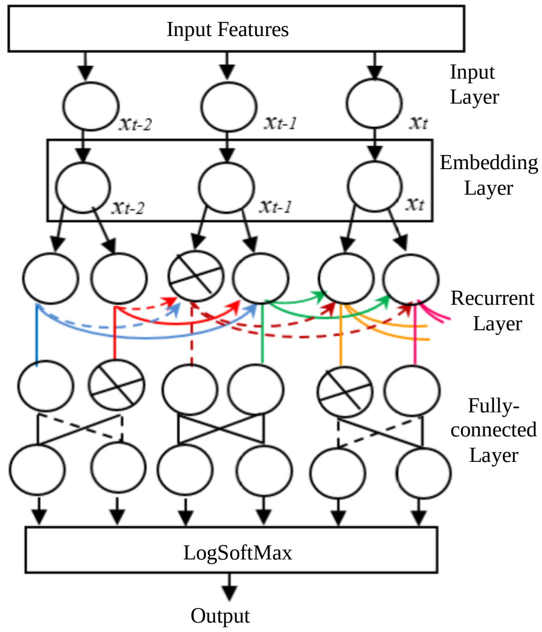


Figure 4. Recurrent Neural Network Architecture.

Figure 5 demonstrates how the components of script identification system are integrated. Firstly, all database samples are divided into pseudo-words and their geometric and dynamics parameters are computed based on beta-elliptic msodel. Thereafter, the obtained feature vectors are empirically clustered into *k* groups using the *k*-means algorithm and converted into one-hot vectors thanks to the word embedding algorithm. Finally, the pseudo-words vectors belonging to the same word are gathered and introduced to the RNN-LSTM network to predict the class of a given script.

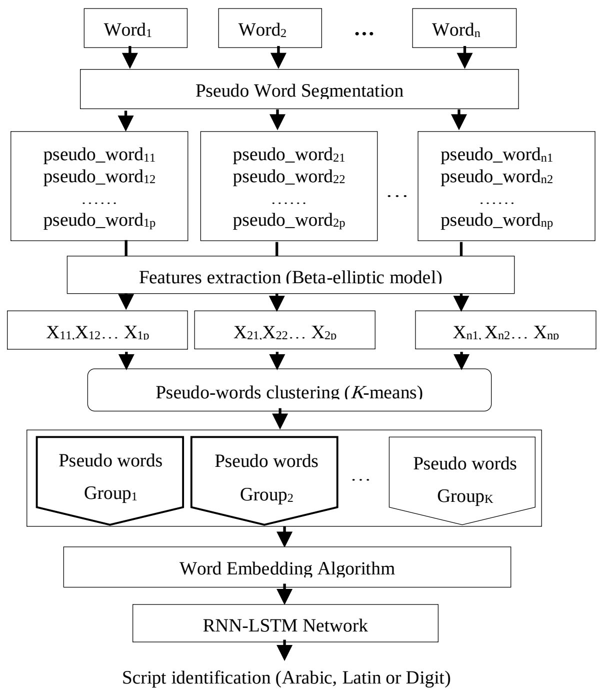


Figure 5. Script identification flowchart.

4. Experiments and Results

We performed the experiments on three public online handwriting datasets: Arabic Database (ADAB), UNIPEN and PENDIGIT. ADAB is known as a standard benchmark in the ICDAR2009 and it is composed of 6 sets with a total of 21575 Arabic handwritten words produced by 166 writers and presenting some Tunisian town names. Each sample from database is stored in "UPX" file format that contains the sequence of (x, y) trajectory and some additional information's about the writer and the word label. For Latin script, the Unipen-ICROW-03 benchmark dataset have been used. It is a multi-writer database composed of 13119 words belonging to 884 classes. However, the pen trajectory is encoded as a sequence of segments according to the UNIPEN format. The last database, called Pendigit, contains a total of 10992 digits collecting from 44 writers. For each digit (0-9), the pen trajectory is recorded in the UNIPEN format file. Thereafter, each one of these three databases is randomly split into two disjoint subsets. Two third has been used for the training stage and the rest for the test phase (Table 3).

Table 3. Databases description.

	ADAB	UNIPEN	PENDIGIT
Training set	15716	10383	7494
Test set	7859	2739	3498
Total	21575	13119	10992

Then, we obtained a total of 33593 words for the training set and 12093 for the test set, which are divided into their pseudo pseudo-words in order to applying the word embedding algorithm (Table 4).

Table 4. Word/pseudo-words numbers.

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

In contrast to the handwritten characters where the number of labels is known in advance (28 for Arabic and 26 for Latin scripts), there are multiple shapes of pseudo-words due to what any vocabulary already exists. Consequently, the number of labels (classes) is unknown and thus we cannot manually associate a label for each pseudo word. Our contribution consists on building a pseudo-word vocabulary from database samples using k-means algorithm. We obtained a vocabulary of 215 labels which is useful for word embedding layer.

The proposed architecture turns as follow; for a given script, the beta-elliptic parameters of its pseudo-words are extracted and passed to the word embedding layer. Thus, we obtained a matrix where the number of rows presents the pseudo-words and the number of columns is equal to the vocabulary size. In this model, three hidden layers are used, one of which is recurrent with LSTM units and the rest are fully-connected. These layers are composed respectively by 128, 128 and 64 units. This network is trained by stochastic gradient descent with a fixed learning rate of 10^{-3}. Dropout is applied both in LSTM and fully-connected layers with a probability *p=0,3*. Tables 5 and 6 show that experiments results attained superior performances with an identification rate of 100% after 5 epochs when applying the dropout technique.

Table 5. Script identification results.

Epoch	Co-occurrence matrix	Rate (%)
1	1856 927 5076 280 314 2142 1 0 3497	45.5
2	1210 44142235 365 1030 1341 0 0 3498	53,8
3	12 7685 162 26 1547 1163 0 0 3498	47.4
4	161 7676 22 0 2731 5 0 0 3498	59.3
5	7610 249 0 0 2736 0 1 0 3497	98.6

Table 6. Dropout improvement.

Epoch	no dropout	Dropout
5	98.6%	100%

The obtained results prove the effectiveness of the proposed system tested on a large vocabulary. Besides script identification rate, our system takes a short time in training stage despite the huge number of pseudo-words. It is thanks to the word embedding layer and the dropout technique. Moreover, dropout allows improving the script identification rate by 1.4%.

To evaluate the proposed system, we must compare it with the previous already exists systems. We have difficulties to do because few works turns on online handwriting script identification. Furthermore, no benchmark database is commonly used for script identification. Table 7 presents a comparison with some existing systems.

Table 7. Results comparison.

	System Description		
Authors	**Scripts**	**Method**	**Rate**
Tan *et al*. [25]	Arabic Latin Tamil	Information Retrieval	93.3%
Namboodiriand Jain [17]	Arabic C1yrillic Devnagari Han Hebrew	Spapial and Temporal features with KNN	96%
Jlaiel *et al*. [9]	Arabic Latin	Geometric features	96%
Proposedsystem	Arabic Latin Digits	Beta-elliptic model with LSTM	100%

5. Conclusions

We presented a new method for online handwriting script identification based on beta-elliptic model. The fundamental objective of this study is to explore the potential utility of beta-elliptic model with different forms of scripts. In this work, several contributions have been presented. Firstly, we have segmented scripts into continuous trajectories called pseudo-words in order to present the sequential aspect of data. These pseudo-words are useful to build a vocabulary. Afterward, we added a word embedding layer to LSTM network in order to map beta-elliptic inputs into 'one-hot' vector containing the pseudo-words clues in the vocabulary. The application of dropout in both LSTM and fully-connected layers allowed us to obtain a higher performance on handwritten scripts from three public databases representing Arabic, Latin and digit scripts. As perspective, the proposed system will be extended to do script recognition after the identification stage.

References

[1] Alimi A., "Beta Neuro-Fuzzy Systems," *TASK Quarterly Journal, Special Issue on Neural Networks*, vol. 7, no.1, pp. 23-41, 2003.

[2] Assawinjaipetch P., Sornlertlamvanich V., Shirai K., and Marukata S., "Recurrent Neural Network with Word Embedding for Complaint Classification," *in Proceedings of the 3rd International Workshop on Worldwide Language Service Infrastructure and Second Workshop on Open Infrastructures and Analysis Frameworks for Human Language Technologies*, Osaka, pp. 36-43, 2016.

[3] Chaabouni A., Boubaker H., Kherallah M., El Abed H., and Alimi A., "Static and Dynamic Features for Writer Identification Based on Multi-Fractals," *The International Arab Journal on Information and Technology*, vol. 11, no. 4, pp. 416-424, 2014.

[4] Dhieb T., Ouarda W., Boubaker H., Halima M., and Alimi A., "Online Arabic Writer Identification Based on Beta-Elliptic Model," *in Proceedings of the 15th International Conference on Intelligent Systems Design and Applications*, Marrakech, pp. 74-79, 2015.

[5] Graves A., Mohamed A., and Hinton G., "Speech Recognition with Deep Recurrent Neural Networks," *in Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, pp. 6645-6649, 2013

[6] Haboubi S., Maddouri S., and Amiri H., "Discrimination between Arabic and Latin from Bilingual Documents," *in Proceedings of the International Conference on Communications, Computing and Control Applications*, Hammamet, pp. 1-6, 2011.

[7] Hasan A., Afzal M., Shafait F., Liwicki M., and Breuel T., "A Sequence Learning Approach for Multiple Script Identification," *in Proceedings of the 13th International Conference on Document Analysis and Recognition*, Tunis, pp. 1046-1050, 2015.

[8] Hochreiter S. and Schmidhuber J., "Long short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997.

[9] Jlaiel M., Kanoun S., Alimi A., and Mullot R., "Three Decision Levels Strategy for Arabic and Latin Texts Differentiation in Printed and Handwritten Natures," *in Proceedings of the 9th International Conference on Document Analysis and Recognition*, pp. 1103-1107, 2007.

[10] Kherallah M., Bouri F., and Alimi A., "On-Line Arabic Handwriting Recognition System Based on Visual Encoding and Genetic Algorithm," *Engineering Applications of Artificial Intelligence*, vol. 22, no. 1, pp. 153-170, 2009.

[11] Kherallah M., Haddad L., Alimi A., and Mitiche A., "On-Line Handwritten Digit Recognition Based on Trajectory and Velocity Modeling," *Pattern Recognition Letters*, vol. 29, no. 5, pp. 580-594, 2008.

[12] Kulkarni A., Upparamani P., Kadkol R., and Tergundi P., "Script Identification from Multilingual Text Documents," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 4, no. 6, pp. 15-19, 2015.

[13] MacQueen J., "Some Methods for Classification and Analysis of Multivariate Observations," *in Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, no. 14, pp. 281-297, 1967.

[14] Mesnil G., He X., Deng L., and Bengio Y., "Investigation of Recurrent-Neural-Network Architectures and Learning Methods for Spoken Language Understanding," *in Proceedings of the 14th Annual Conference of the International Speech Communication Association*, Lyon, pp. 3771-3775, 2013.

[15] Mezghani A., Slimane F., Kanoun S., and Märgner V., "Identification of Arabic/French Handwritten/Printed Words using GMM-Based System," *in Proceedings of the 11th French Information Retrieval Conference*, Nancy, pp. 371-374, 2014.

[16] Moon T., Choi H., Lee H., and Song I., "RNNDROP: A Novel Dropout for RNNS in ASR," *in Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding*, Scottsdale, pp. 65-70, 2015.

[17] Namboodiri A. and Jain A., "Online Handwritten Script Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 1, pp. 124-130, 2004.

[18] Rouhou A., Abdelhedi Z., and Kessentini Y., "A HMM-Based Arabic/Latin Handwritten/Printed Identification System," *in Proceedings of the*

International Conference on Hybrid Intelligent Systems, Seoul, pp. 298-307, 2016.

[19] Saïdani A. and Echi A., "Pyramid Histogram of Oriented Gradient for Machine-Printed/Handwritten and Arabic/Latin Word Discrimination," *in Proceedings of the 6th International Conference of Soft Computing and Pattern Recognition*, Tunis, pp. 267-272, 2014.

[20] Saïdani A., Echi A., and Belaid A., "Identification of Machine-Printed and Handwritten Words in Arabic and Latin Scripts," *in Proceedings of the 12th International Conference on Document Analysis and Recognition*, Washington, pp. 798-802, 2013.

[21] SaïdaniA., Kacem A., and Belaid A., "Co-Occurrence Matrix of Oriented Gradients for Word Script And Nature Identification," *in Proceedings of the 13th International Conference on Document Analysis and Recognition*, Tunis, pp. 16-20, 2015.

[22] Sak H., Senior A., and Beaufays F., "Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling," *in Proceedings of the 15th Annual Conference of the International Speech Communication Association*, Singapore, pp. 1102-1128, 2014.

[23] Singh A. and Lawahar C., "Can RNNs Reliably Separate Script and Language at Word and Line Level," *in Proceedings of the 13th International Conference on Document Analysis and Recognition*, Tunis, pp. 976-980, 2015.

[24] Singh P., Sarkar R., Nasipuri M., and Doermann D., "Word-Level Script Identification for Handwritten Indic Scripts," *in Proceedings of the 3rd International Conference on Eco-friendly Computing and Communication Systems*, Mangalore, pp. 1106-1110, 2015.

[25] Tan G., Viard-Gaudin C., and Kot A., "Information Retrieval Model for Online Handwritten Script Identification," *in Proceedings of the 10th International Conference on Document Analysis and Recognition*, Barcelona, pp. 336-340, 2009.

[26] Zouari R., Boubaker H., and Kherallah M., "Hybrid TDNN-SVM Algorithm for Online Arabic Handwriting Recognition," *in Proceedings of the 16th International Conference on Hybrid Intelligent Systems*, Seoul, pp. 113-123, 2016.

[27] Zouari R., Boubaker H., and Kherallah M., "A Time Delay Neural Network for Online Arabic Handwriting Recognition," *in Proceedings of the International Conference on Intelligent Systems Design and Applications*, Porto, pp. 1005-1014, 2016.

[28] Zouari R., Mokni R., and Kherallah M., "Identification and Verification System of Offline Handwritten Signature Using Fractal Approach," *in Proceedings of the First International Conference on Image Processing Applications and Systems*, Sfax, pp. 1-4, 2014.

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

Houcine Boubaker graduated in Electrical Engineering in 1995, obtained a master degree in Systems Analyses and Digital Signal Processing in 1997. He is a researcher in Electrical and Computer Engineering at the University of Sfax and affiliate to the Research Groups in Intelligent Machines laboratory (ReGIM). His research interest includes trajectories modeling and pattern recognition. He focuses his research on drawing, handwriting and arm-hand movements modeling and Analyses.

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