ACIT'2017
The International Arab Conference on Information Technology
Yassmine Hammamet, Tunisia
December 22-24, 2017

RNN-LSTM based Beta-elliptic model for Online handwriting script identification

RAMZI ZOUARI, HOUCINE BOUBAKER and [1]MONJI KHERALLAH
NATIONAL SCHOOL OF ENGINEERS OF SFAX, TUNISIA
[1]FACULTY OF SCIENCES OF SFAX, TUNISIA

***Abstract*—Recurrent Neural Network 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 neural network with Long Short-Term Memory. The representation of the sequential data is done through the beta-elliptic model. It allows extracting the kinematics and geometrics profiles of the 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 achieved a recognition rate of 100% with only one recurrent layer and using the dropout technique.**

Keywords—online; pseudo; stroke; velocity; beta-elliptic; recurrent; LSTM; dropout

I. INTRODUCTION

Handwriting analysis still remains a challenge in the field of document analysis. Thanks to the technological revolution in touch-capturing devices (PDA, Tablet PC, etc...), we have a very rapid growth of handwritten documents which are more difficult to treat than OCR documents. This is due to the cursive style of the handwriting hence the presence of ligatures, valleys and delayed strokes [1]. Works which deals with handwritten scripts are generally categorized into two branches. The first one 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 [2-4]. Recently, multiple researches deal with script identification. Their purpose consists on identifying some characteristics from the script, 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 power 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.

II. RELATED WORKS

Script and language identification is the first step in document automation process. In literature, several approaches were proposed. We can cite the system of Singh et al. [5] who presents a sliding window technique that operates over an OCR document. The computed sequence features are then given as input to the RNN. This method has been applied both at word and line levels and reached a high performance on a large East-Asian scripts database. In the same way, Hasan et al. [6] presented an approach in which a sliding window traverses over the input image and each frame is fed into Bidirectional LSTM network in order to classify it into one of the target classes. This system achieved a script recognition accuracy of 98% on a developed printed database consisting of English and Greek scripts. Kulkarni et al. [7] applied Discrete Wavelet Transform (DWT) on the visual features extracted from multilingual text.

Works which treat the handwriting scripts were carried. A study was made by Singh et al. [8] where a combination of elliptical and polygonal features is investigated. The proposed method has been evaluated on a dataset of 7000 handwritten from six different scripts, and achieved the rate of 95% with Multi-Layer Perceptron (MLP) classifier. In [9, 10], Saidani et al. used the Co-occurrence Matrix and pyramid histogram of Oriented Gradients (HOG) for printed/handwritten and Latin/Arabic scripts. The identification is achieved by k-Nearest Neighbors KNN and Support Vector Machines SVM classifiers and obtained an error rates less than 2% on IAM and IFN\ENIT databases respectively. HOG features are also used in [11] with Hidden Markov Models (HMM) based identification that operates both on word and line levels. Another work was made by Saidani et al. [12] where new geometric features as bottom diacritics and loop position are used with Bayes classifier for printed and handwritten words for Arabic and Latin scripts. Mezghani et al. [13] presented a writing type/language identification system based on Gaussian Mixture Models

(GMMs). In fact a fixed-length sliding window was used for the feature extraction whose distribution is captured by GMMs through the training procedure.

According to the literature, most research is based on offline script identification technology, but there are only few reports about online script identification despite the emergence of data entry devices [14]. Among the online approaches, Tan et al. [15] proposed a novel approach based on the Information Retrieval model for the Arabic, Roman and Tamil scripts which attained an average accuracy of 93%. Namboodiri et al. [16] presented an online handwritten script recognition system for classifying six major scripts at word level. It is based on spatial and temporal features of the strokes and attained an overall classification accuracy of 86%.

III. SYSTEM OVERVIEW

The beta-elliptic model has not yet used for online script identification. Since it has strongly applied for online handwriting recognition [17] and writer identification [18], 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 trajectories, called pseudo-words, which represents the interval between pen-down and pen-up moments. Then, the beta-elliptic model is executed at pseudo-word level.

A. Beta-elliptic modeling

Handwriting movement involves the mobilization of N neuromuscular sub-systems [19]. 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.

1) Velocity profile modeling: the curvilinear velocity curve $V_\sigma(t)$ shows a signal that alternate a successive extremas of velocity where each stroke converge to the beta impulse [20]. Then, the overall signal is approximated by an algebric addition of overlapped beta impulses (equation 2).

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

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

With:

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

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

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 above figures illustrate the application of the beta-elliptic model for Arabic, Latin and digit scripts respectively. On the right side, the black curve represents the curvilinear velocity signal, red and blue curves represent the overlapped beta impulses and the green dotted curve the sum of beta impulses. In the figures at left, the original handwriting trajectory is drawn with black curve and the red and green curves present the rebuilding trajectories fitted by elliptic arcs (section III.2)

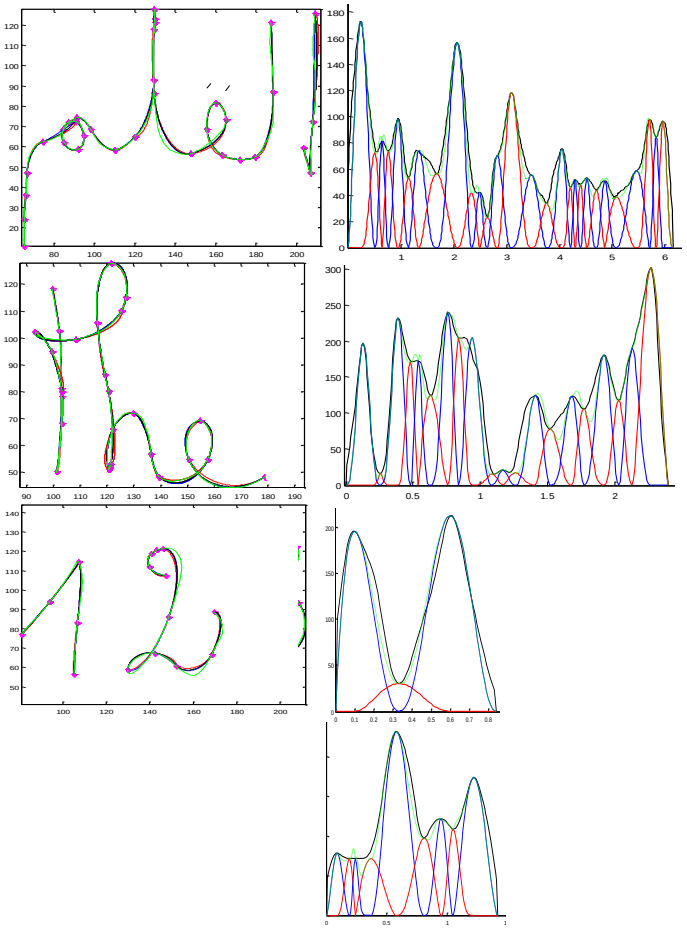


Fig. 1. Beta-elliptic modeling for online Arabic, Latin and Digit Handwriting scripts

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) as represented in Figure 2. The parameters a and b are respectively the half dimensions of the large and the small axes of the elliptic arc. θ is the angle of the ellipse major axe inclination and θ_p is the angle of inclination of the tangents at the stroke endpoint M_2.

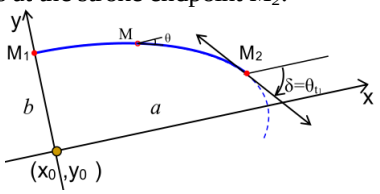


Fig. 2. Handwriting trajectory segment fitted by elliptic arc

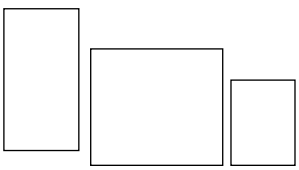
Each stroke in the pseudo-word is modeled by a vector of 10 features (see Table1). Thus, a script is the concatenation of the beta-elliptic parameters of its different pseudo-words.

TABLE I. BETA-ELLIPTIC PARAMETERS

	Parameters	Explanation
Dynamic profile	K	Beta impulse amplitude
	Δt=(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 profile	a	Ellipse major axis half length
	b	Ellipse small axis half length
	θ	Ellipse major axis inclination angle
	θ_p	Angle of inclination of the tangents at the stroke endpoint M_2
	POS	Position of the stroke

B. *Word Embedding Technique*

Word Embedding is a natural language modelling 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 neural networks, co-occurrence matrix and Word2Vec models [21]. 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 database 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 [22]. 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 (a vector of zeros with 1 at a single position). The following figure shows the decomposition of the Arabic word 'RASOUL' on its pseudo-words and the application of word embedding model with vocabulary size equal to 7.



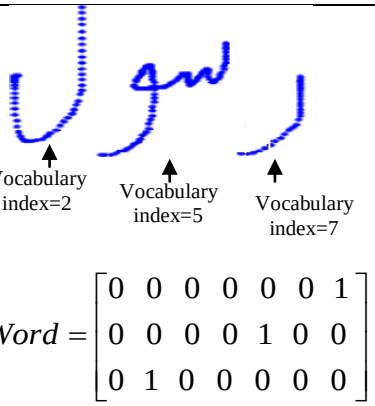


Fig. 3. Word embedding representation

With this representation, word embedding allows projecting very high-dimensional sparse vector into a low-dimensional dense vector for word representation. It was successfully used for a variety of natural language tasks [23].

C. *Recurrent Neural Network*

Recurrent Neural Networks 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 [24]. RNN is trained by stochastic gradient descent using Backpropagation through Time algorithm (BPTT). A major problem with gradient descent for standard RNN architectures (Vanilla RNN) is that error gradients vanish exponentially quickly with long-term dependencies, due to what is called the vanishing/exploding gradient problem. To address this drawback, Hochreiter & Schmidhuber (1997), have designed a Long Short Term Memory (LSTM) network which can in principle store and retrieve information over long time periods [25]. LSTM explicitly designs a memory block inside a hidden node that has the following ingredients: memory cell which stores information about the past and input C_t, output, and forget gates that control the flow of information within and among the memory blocks. Figure 4 shows the structure of a single LSTM memory block, which replaces a simple hidden node used in ordinary RNNs.

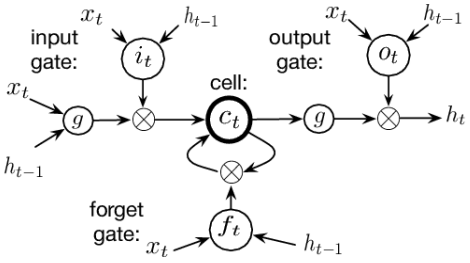


Fig. 4. LSTM memory block

An LSTM network recurrently applies the following series of equations to obtain the sequence of hidden node outputs, $\mathbf{h}=(h_1, h_2..., h_T)$, $h_t \in R^m$:

$$i_t=\sigma(W_{xi}\, x_t+W_{hi}\, h_{t-1}+W_{Ci}\, C_{t-1}+b_i) \quad (3)$$

$$f_t=\sigma(W_{xf}\, x_t+W_{hf}\, h_{t-1}+W_{Cf}\, C_{t-1}+b_f) \quad (4)$$

$$C_t=f_t\, C_{t-1}+i_t\, tanh\,(W_{xc}\, x_t+W_{hc}\, h_{t-1}+b_c) \quad (4)$$

$$O_t=\sigma(W_{xo}\, x_t+W_{ho}\, h_{t-1}+W_{Co}\, C_{t-1}+b_o) \quad (5)$$

$$h_t=O_t\, tanh(C_t) \quad (6)$$

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 [26]. In addition to LSTM blocks, another method can be applied in order to prevent RNN from overfitting during training stage, called Dropout. It is a popular regularization technique for the feed-forward neural networks where some network units are randomly masked during training. This method was typically applied only at non-recurrent layers. Recently, it is applied at the recurrent layers [27], i.e. dropout is applied on the output at step *t* before it is used to compute the output at step *t+1*(Fig. 2).

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 (Section III.B) 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 (Fig. 5). 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.

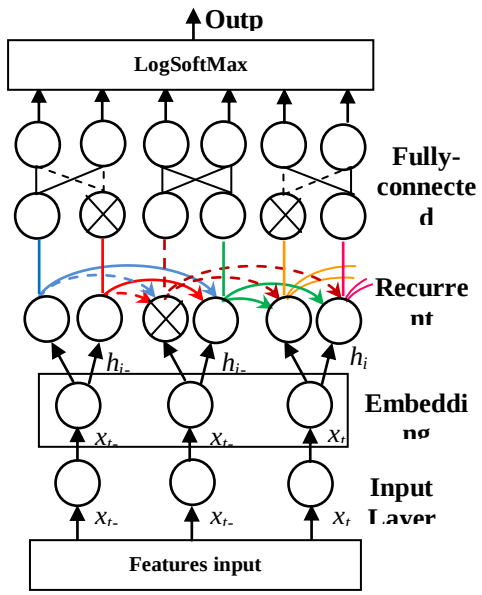


Fig. 5. Script identification Architecture

IV. EXPERIMENTS AND RESULTS

We performed the experiments on three public online handwriting datasets: ADAB, UNIPEN and PENDIGIT containing handwritten Arabic words, Latin words and digits, respectively. Each word from database is then subdivided into its pseudo-words (section III). Moreover, databases are then split into disjoint subsets to train and evaluate our model.

TABLE II. DATADASES DESCRIPTION

	ADAB (Arabic script)		***UNIPEN (Latin Script)***		***PENDIGIT (digit script)***	
	words	***Pseudo***	***words***	***pseudo***	***Digits***	***pseudo***
Training	15716	76616	10383	44876	7494	9378
Test	7859	38308	2736	11965	3498	4219
Total	**21575**	**114924**	**13119**	**56841**	**10992**	**13597**

TABLE III. WORD/PSEUDO-WORD DATADASE 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 in Arabic, 26 in Latin scripts and 10 in digits), 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 number of 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*. Experiments results showed superior performances with an identification rate of 100% after 7 epochs when applying the dropout technique. It took less than 13 minutes.

TABLE IV. EXPERIMETS RESULTS

epoch	Co-occurrence matrix	Rate (%)	epoch	Co-occurrence matrix	Rate (%)
1	1856 927 5076 280 314 2142 1 0 3497	**45.5**	2	1210 4414 2235 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**	6	7786 73 0 0 2736 0 1 0 3497	**99.5**
7	7859 0 0 0 2736 0 0 0 3498	**100**	8	7859 0 0 0 2736 0 0 0 3498	**100**

The obtained results prove the effectiveness of the proposed system which is tested on 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 improves the script identification rate by 1.3%.

TABLE V. DROPOUT IMPROVEMENT

	Script identification rate	
Epoch	***no dropout***	***with dropout***
7	98.7%	100%

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. The following table presents a comparison with some existing systems.

TABLE VI. COMPARISON RESULTS

	System Description		
Authors	***Scripts***	***Method***	***Rate***
Tan et al. [15]	Arabic Latin Tamil	Information Retrieval	93.3%
Namboodiri et al. [16]	Arabic C1yrillic Devnagari Han Hebrew	Spapial and Temporal features with KNN	96%
Jlaiel et al. [28]	Arabic Latin	Geometric features	96%
Proposed	Arabic	Beta-elliptic	100%
system	Latin Digits	model with LSTM	

V. CONCLUSION

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, we will intend to extend the proposed system with a second stage for multi-language script recognition.

REFERENCES

[1] M. Kherallah, F. Bouri and A.M. Alimi, "On-line Arabic handwriting recognition system based on visual encoding and genetic algorithm", Engineering Applications of Artificial Intelligence Vol. 22, Issue 1, 2009, pp. 153-170

[2] R. Zouari, H. Boubaker and M. Kherallah, "A time delay neural network for online arabic handwriting recognition", In proceedings of the International Conference on Intelligent Systems Design and Applications, 2016, pp. 1005-1014

[3] R. Zouari, R. Mokni and M. Kherallah, "Identification and verification system of offline handwritten signature using fractal approach", in proceeding of first international conference on image processing applications and systems, 2014, pp. 1-4,IEEE.

[4] A. Chaabouni, H. Boubaker, M. Kherallah, H. El Abed and A.M. Alimi, "Static and dynamic features for writer identification based on multi-fractals". International Arab Journal on Information and Technology, 2014, pp. 416-424.

[5] A. Singh and C. Lawahar, "Can RNNs reliably separate script and language at word and Line Level?", in 13th International Conference on Document Analysis and Recognition (ICDAR), 2015, pp. 976-980, IEEE

[6] A. Hasan, M. Afzal, F. Shafait, M. Liwicki and T.M Breuel, " A sequence learning approach for multiple script identification", in 13th International Conference on Document Analysis and Recognition (ICDAR), 2015, pp. 1046-1050, IEEE.

[7] A. Kulkarni, P. Upparamani, R. Kadkol,and P. Tergundi, "Script Identification from Multilingual Text Documents", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 4, Issue 6, 2015, pp. 15-19.

[8] P. Singh, R. Sarkar, M. Nasipuri and D. Doermann, , "Word-level script identification for handwritten Indic scripts.". In 13th International Conference on Document Analysis and Recognition (ICDAR), 2015, pp. 1106-1110, IEEE.

[9] A. Saïdani, A. Kacem and A. Belaid, " Co-occurrence matrix of oriented gradients for word script and nature identification", In 13th International Conference on Document Analysis and Recognition (ICDAR), 2015, pp. 16-20, IEEE.

[10] A. Saïdani, and A. Echi, "Pyramid histogram of oriented gradient for machine-printed/handwritten and arabic/latin word discrimination". In 6th International Conference of Soft Computing and Pattern Recognition (SoCPaR), 2014, pp. 267-272, IEEE.

[11] A. Rouhou, Z. Abdelhedi and Y. Kessentini, "A HMM-Based Arabic/Latin Handwritten/Printed Identification System". In International Conference on Hybrid Intelligent Systems, 2016, pp. 298-307, Springer, Cham.

[12] A. Saïdani, A. Echi and A. Belaid, "Identification of machine-printed and handwritten words in Arabic and Latin scripts". In 12th International Conference on Document Analysis and Recognition (ICDAR), 2013, pp. 798-802, IEEE.

[13] A. Mezghani, F. Slimane, S. Kanoun and V. Märgner, "Identification of Arabic/French Handwritten/Printed Words using GMM-Based System", In CORIA-CIFED, 2014, pp. 371-374.

[14] K. Ubul, G. Tursun, A. Aysa, D. Impedovo, G. Pirlo and T. Yibulayin, " Script Identification of Multi-Script Documents: A Survey", IEEE Access, Vol.5, 2017, pp. 6546-6559.

[15] G. Tan, C. Viard-Gaudin and A. Kot, "Information retrieval model for online handwritten script identification", In 10th International Conference on Document Analysis and Recognition (ICDAR), 2009, pp. 336-340, IEEE.

[16] A. Namboodiri and A. Jain, " Online handwritten script recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence,Vol. 26, N°1, 2004, pp. 124-130.

[17] R. Zouari, H. Boubaker and M. Kherallah, "Hybrid TDNN-SVM algorithm for online arabic handwriting recognition", in proceedings of the 16th International Conference on Hybrid Intelligent Systems, 2016, pp. 113-123

[18] T. Dhieb, W. Ouarda, H. Boubaker, M. Halima and A. Alimi, "Online Arabic writer identification based on beta-elliptic model ", In 15th International Conference on Intelligent Systems Design and Applications (ISDA), 2015, pp. 74-79, IEEE.

[19] M. Kherallah, L. Haddad, A.M. Alimi and A. Mitiche, "On-line handwritten digit recognition based on trajectory and velocity modeling ", Pattern Recognition Letters, Vol. 29, N°5, 2008, pp. 580-594.

[20] A.M. Alimi, "Beta neuro-fuzzy systems", TASK Quarterly Journal, Special Issue on Neural Networks, Vol. 7, N°1, 2003, pp. 23-41.

[21] P. Assawinjaipetch, V. Sornlertlamvanich, K. Shirai and S. Marukata, " Recurrent Neural Network with Word Embedding for Complaint Classification", Proceedings of WLSI/OIAF4HLT, 2016, pp. 36-43.

[22] J. MacQueen, "Some methods for classification and analysis of multivariate observations", In Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, Vol.1, N°14, 1967, pp. 281-297

[23] G. Mesnil, X He, L. Deng and Y. Bengio, " Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding", In Interspeech, 2013, pp. 3771-3775.

[24] A. Graves, A. Mohamed and G. Hinton, " Speech recognition with deep recurrent neural networks", In international conference on Acoustics, speech and signal processing (icassp), 2013, pp. 6645-6649, IEEE.

[25] H. Sak, A. Senior and F. Beaufays. "Long short-term memory recurrent neural network architectures for large scale acoustic modeling ", In 15th Annual Conference of the International Speech Communication Association, arXiv preprint arXiv:1402.1128, 2014

[26] T. Moon, H. Choi, H. Lee and I. Song, " Rnndrop: A novel dropout for rnns in asr", In IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU), 2015, pp. 65-70, IEEE.

[27] V. Pham, T. Bluche and C. Kermorvant and J. Louradour, " Dropout improves recurrent neural networks for handwriting recognition", In 14th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2014, pp. 285-290, IEEE.

[28] M. Jlaiel, S. Kanoun, A.M. ALIMI, and R. Mullot, "Three decision levels strategy for Arabic and Latin texts differentiation in printed and handwritten natures", In 9th International Conference on Document Analysis and Recognition (ICDAR).Vol.2, 2007, pp. 1103-1107, IEEE.