Arabic Handwriting Word Recognition Based on Convolutional Recurrent Neural Network

Manal Boualam, Youssef Elfakir, Ghizlane Khaissidi, and Mostafa Mrabti

Abstract The success of any words-characters recognition system depends on board parameters such as the language (Arabic, Latin, Indi …), the document type (writing or typing), based or free-segmentation, pretreatment, features extraction and classification approaches. Within these fields, Building a robust and viable recognition system for Arabic handwritten has always been a challenging task since a long time. In this study, we propose an end-to-end system based on deep Convolutional Recurrent Neural Network CNN/RNN; we trained our system on IFN/ENIT extended database in order to improve our results.

Keywords Arabic handwriting recognition · Convolutional neural network · Recurrent neural network · IFN/ENIT database

1 Introduction

Optical Character recognition OCR system for Arabic used to solve several problems in different areas: analyze humans sentiment [1, 2], writer identification [3]. Disease detection such as Parkinson's [4], speech recognition [5], etc.

Usually, the process of word recognition achieved through four steps: preprocessing, segmentation, features extraction and classification. In this model, each step input depends on the output of the previous one (pipeline), which could increase the error rate. In the recent studies, researchers used deep learning in order to realize

M. Boualam (✉) · Y. Elfakir · G. Khaissidi · M. Mrabti
Laboratory of Information and Interdisciplinary Physics, ENS, University Sidi Mohamed Ben Abdellah, Fes, Morocco
e-mail: Boualam.manal@gmail.com

Y. Elfakir
e-mail: Elfakir.youssef11@gmail.com

G. Khaissidi
e-mail: Ghizlane.derkaoui1@hotmail.com

M. Mrabti
e-mail: Mostafa.mrabti@yahoo.fr

S. Bennani et al. (eds.), *WITS 2020*, Lecture Notes in Electrical Engineering 745,
https://doi.org/10.1007/978-981-33-6893-4_79

the system. Which is a free-segmented model starts by preprocessing then features extraction and classification. Our proposed model inspired by [6] constructed of Convolutional Neural Network (CNN) layers, Recurrent Neural Network (RNN) layers and Connectionist Temporal Classification (CTC) layer.

Elbashir [7] proposed a Convolutional neural network (CNN) model for off-line Arabic handwritten, they used Sudan University of Science and Technology dataset (SUST ALT), the model accuracy is 93.5%. In [8] Najib Tagougui et al. built a hybrid model NN/HMM for Arabic handwritten recognition (AHR), they trained the segments on a multi-layer perceptron Neural Network (MLPNNs) to extract characters probabilities. A Hidden Markov Model (HMM) used to decode the outputs of the first system, the evaluation of their model done on ADAB database; it achieved 96.4%.

Younis [9] used a deep neural network to build an Arabic handwritten recognition system, they started by a Batch Normalization to improve the speed and the accuracy of their system, the accuracy is 94.8% on AIA9k [10] dataset and 97.6% on AHDC dataset. El-Sawy et al. [11] model a deep learning architecture based on CNN to recognize Arabic Handwritten characters, with two convolutional layers and two pooling layers, they created their own dataset in order to train and test the proposed system, the dataset is composed of 16,800 character with 94.9% of accuracy.

Sahlol et al. [12] proposed a novel method for Arabic characters recognition, starting with a new preprocessing step based on noise removal and different kind of features, then they trained the system on CENPRMI database [13] to feed a neural network, it gives an accuracy = 88%. Abdalkafor et al. [14] used the same database to build their system based on novel features extraction techniques and MLP NN, the system accuracy reached up to 94.75%.

In [15] Mohammed Ali Mudhsh et al. presented an Alphanumeric VGGNet for Arabic character recognition, their system is developed by thirteen convolutional layers, three fully connected layers and two max-pooling layers, the system shows a high performance with an accuracy equal to 99.66% after training it on ADBBase [16] database and 97.32% for HACDB Dataset [17]. In [18] Ahmed El-Sawy et al. provided a deep learning technique for Arabic Digits Recognition, five CNN layers trained on MADBase with 88% of accuracy.

Few researchers focused on Arabic word/text recognition. In [19, 20] Mohamed Elleuch et al. integrated two classifiers: Convolutional Neural Network (CNN) and Support Vector Machine (SVM) in order to recognize Arabic words Handwritten, the training and testing is done on IFN/ENIT [21] and HACDB [22] databases, their model compared to other Optical Character Recognition systems (OCR) gives better results. Shi et al. [6] developed a novel End-to-End Convolutional Recurrent Neural Network (CRNN) for scene Text recognition; their system handles the variation of length and height of words in scenes, the experiments is evaluated on the IIIT-5 K [23], Street View Text [24] and ICDAR [25] datasets it shows a competitive performance. Alaa Alsaeedi et al. [26] adopted their system to use it in smartphones; they used a CNN for characters recognition and transparent neural Network (TNN) for printed words recognition, the recognition rate of their system is 98%.

Or, text recognition in video is a very interesting topic for OCR community, Yousfi et al. [27] improved Bidirectional Long Short-Term Memory-Connectionist Temporal Classification (BLSTM-CTC) segmentation-free model for Arabic video text recognition based on a joint learning of Maximum entropy and RNN models, they judged that their model outperform the classical BLSTM-CTC model by 36%. In [28] Ali Mohcine et al. proposed a model, which convert an Arabic line into characters, and then feed into a neural network, their model achieved an accuracy up to 83%.

This paper arranged as follow: Sect. 2 describes the existing OCR systems based on neural networks with a brief discussion, Sect. 3 presents the proposed approach, used techniques and results discussion. The paper ends with conclusion and perspectives for future work.

2 Material and Methods

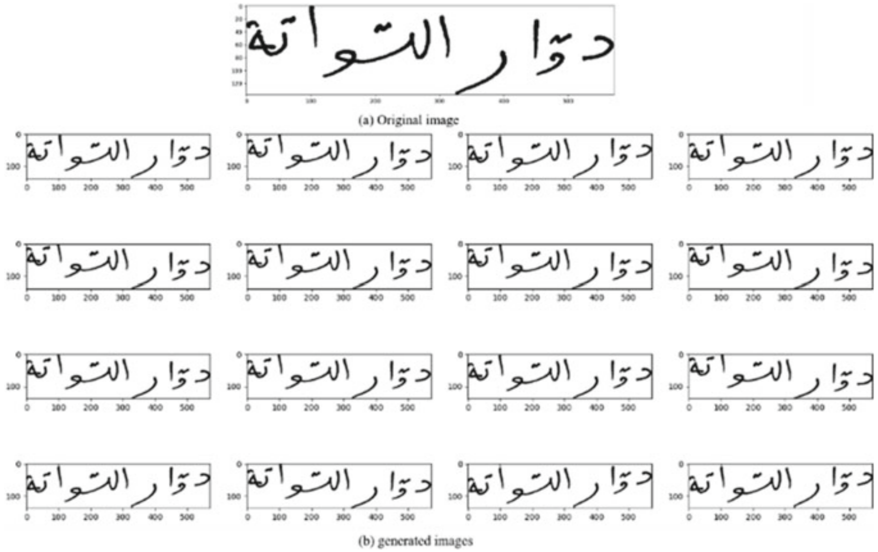
2.1 Database

As input of our model, we used 946 name of Tunisian town/village written by more than 1000 writers. The original database is composed of 32,492 handwritten name, to improve the performance/ability and prevent over fitting of our model a large dataset is crucial, we used Keras deep learning neural network to fit our model using image data augmentation via the ImageDataGenerator, by applying domain-specific techniques to base images in order to create transformed versions of images. The main challenge is the selection of techniques used for data expend to not lose information or generate damaged/unreadable images.

In our model, we used a simple and powerful library for data preparation, the library provides class that define the configuration for image data preparation and augmentation, instead of choose specific arguments for each image, the function fits different arguments to model then randomly transform the original image to generate the desired augmented images. In our model, we used a rotation up to 2°, width and height shift up to 2 and shearing up to 4, in order to generate 15 examples for each image in IFN/ENIT database, as result we fit 487,350 training sample and (~3,876,648 characters) to our CNN–RNN model (Fig. 1).

2.2 Methodology

The main purpose of this work is to propose a robust approach for Arabic word recognition; the model achieved in four steps: preprocessing in order to normalize the input data, five CNN layers to extract relevant features from image, and then two RNN layers for sequence modeling, at the end CTC layer for transcription (Fig. 2).


(a) Original image

(b) generated images

Fig. 1 Image augmentation using image data generator from Keras library of Tensorflow, **a** original image, **b** fifteen randomly generated images: rotation up to 2°, width and hight shift up to 2, shearing up to 4

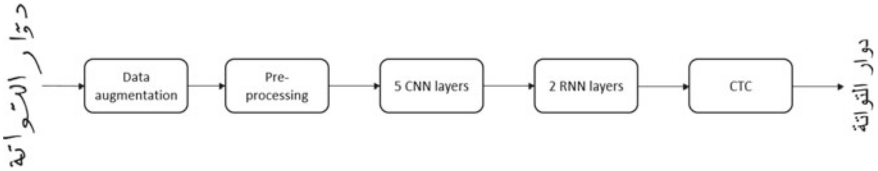


Fig. 2 Overview of the end-to-end proposed model

The proposed model gave significant results for Latin language in different research [6].

Preprocessing. The first fundamental step in an OCR model is preprocessing, the aim is to improve the quality of image data to suppress unwilling distortions or enhances in order to get better results. Even if IFN/ENIT images are already extracted and binarized, we used preprocessing to resize images to 128 × 32 without distortion by copying it to a white target image of same size, and convert our text corpus into UTF-8 to be readable.

CNN Model. A convolutional Neural Network (CNN) used in different fields (image processing, audio processing, etc.), it is the method of loosely simulate the neural human brain network, it define features in the image in two parts: the *convolution layers* that extract features from input images, *Fully connected layers* that used data from the previous part to generate output. In the training step, two importing processes are involved: *forward propagation* process the input data and generate the output, *Backward Propagation* calculate error and update parameters (weight and bias). The CNN that we created contain five layers to identify multiple features, each layer composed of three operations: filter Kernel of size 5 × 5 in the first two layers and 3 × 3 in the last three layers. Then, all negative outputs of the previous layer converted into zero by RELU function (1), in this step the image shape is not transform. The last layer "Pooling" reduce the image dimensionality in order to run the algorithm at a decent speed, the pooling technique used in our model is max-pooling (2 × 2 squares).

$$\{F(x) = \max(0, x)\} \tag{1}$$

RNN Model. RNN is a very powerful neural network, it captures sequential information present in the input data, the output at each step depends on the current word and the previous one, since it is equipped with a mechanism of recurrent feedback that have a memory which captures information about the calculations done previously. RNN suffers from vanishing and exploiting gradient problem (with a large number of time steps), this problem was explored in depth by [29, 30]. In [31] S. Hochreiter and J. Schmidhuber designed LSTMs to avoid the previous problem:

$$h_t = f(W_{xh}x_t + W_{hh}x_{t-1} + b_h) \tag{2}$$

$$y_t = W_{hy}h_t + b_y \tag{3}$$

where x: input sequence given as input, h: sequence of hidden vectors computed by hidden NN, y: output vector. W the matrices weight, b the bias vectors and f the activation function it could be Sigmoid or tanh function. Long Short-Term Memory (LSTM) have access to past but not to future, for this reason, a Bidirectional RNN is used; it feeds the learning algorithm with the original data from beginning to the end, and once from end to beginning using a forward recurrent component and a backward recurrent component. Our model composed of two RNN layer to propagate relevant information through a sequence that contain 256 features per time-step.

CTC function. The Connectionist Temporal Classification [32] CTC loss layer provide end-to-end training and free-segmentation transcription, it takes as input the output matrix of the last RNN layer and the ground truth text (GT), then computes the loss value and infer/decode the matrix to get the text represented in the image. The loss calculation done by summing up all scores of all possible alignments of GT text. Alternatively, decoding done in two steps: First takes the most likely character per time-step and calculates the best path. Second, removes duplicate characters and blanks from path to represent the recognized text.

Hybrid CNN-RNN model. CNN-RNN hybrid model proved excellent results in different fields such as visual description [33], video emotion recognition [33], etc. In our model as described in Fig. 2, we performed feature extraction using CNN,

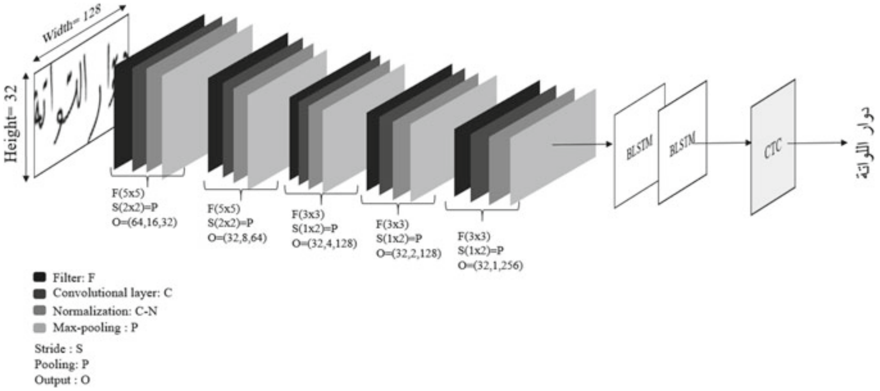


Fig. 3 Proposed end-to-end hybrid architecture based on CNN-RNN-CTC

which acts like encoder, then generate a language model and provide sequence-using RNN, which acts like decoder. Otherwise, the RNN is language based model use the output features of CNN layers and translate it into natural language sentence. Our end-to-end hybrid CNN-RNN model generally used to map static spatial inputs (images) to sequence outputs (text), the proposed model can perform classification of image and text inputs by taking advantages of CNN and RNN (Fig. 3). Our model classify the word containing in the provided image, and recognize it character per character. This allow the model to recognize words even if they are not present in the database (if they are well classified).

3 Results and Discussion

To construct our model we used python language, which have lot of advantages comparing to other languages such as MATLAB widely used for machine learning, C, etc.: it is free and open source, interpreted language so it is ported in different operating systems, it is easier for programing, it contains a wide choice of libraries for learning machine, etc. In [34] Colliau Taylor et al. presented a comparative study between Matlab, Python and R. We implemented our model using TensorFlow [35] library. In the proposed approach, we used 25,000 sample/epoch; 50 sample/batch, the model proved a high performances comparing to similar models using the same dataset IFN/ENIT. Our database split to training and testing images, we wanted to prove that the size of training and testing database affects the results, we used different scales of inputs for training and testing respectively (95–5%, 80–20%, 70–30%), we notice that the results of training dataset between (80–95%) gave the best results with minimal difference. We trained our network until our character error rate (CER) on the development set did not improve for at least 8 epochs, the results are shown in Table 1.

Table 1 CNN-RNN-CTC CER results for different training and testing dataset sizes

Train-test size	CER (%)	WER (%)
95–5%	2.07	91.49
80–20%	2.10	91.79
70–30%	3.03	88.49

Table 2 Performance comparisons with other methods

References	Techniques	CER (%)
Elleuch et al. [19]	SVM + CNN	7.05
Yan et al. [36]	LSTM	6.91
Awni et al. [37]	CNN	6.63
Maalej [38]	Maxout into MDLSTM	10.11
Present work	CNN + RNN + CTC	2.1

To evaluate our system we used character error rate (CER) and word error rate (WER), which represent the number of correct detected words from testing data set, we obtained significant results compared to previous studies using same database (IFN/ENIT). The resulting WER, CER respectively were above 91.79 and 2.10% (Table 2).

4 Conclusion and Future Work

We have presented an offline Arabic end-to-end hybrid recognition system; based on convolutional neural network and recurrent neural network specially bidirectional LSTM method and connectionist temporal classification CTC, we used image data augmentation in order to increase training images.

For perspectives, to improve our model we will focus on training-test data improvement (data preparation), as it is the first source of recognition errors, by eliminating damaged images, and improve the quality of training dataset. The next step will be improving the CNN-RNN by adding more layers, we will try to increase image size to use complex text lines/paragraphs, focusing on two main issues: touching and overlapping.

References

1. Al-Kabi MN, Al-Ayyoub M, Wahsheh HA, Alsmadi I (2016) A prototype for a standard Arabic sentiment analysis corpus. Int Arab J Inf Technol 13(1A):163–170
2. Alayba AM, Palade V, England M, Iqbal R (2017) Arabic language sentiment analysis on health services. https://arxiv.org/abs/1702.03197

3. Akram B (2018) Clonal selection classification algorithm applied to Arabic writer identification. In: 8th international conference on off-line handwriting based writer identification. https://doi.org/10.1145/3200842.3208087
4. Ibtissame A, Ammour A, Khaissidi G, Belahsen F, Mrabti M, Aboulem G (2020) A novel approach combining temporal and spectral features of Arabic online handwriting for Parkinson's disease prediction. J Neurosci Methods. https://doi.org/10.1016/j.jneumeth.2020.108727
5. El Choubassi MM, El Khoury HE, Alagha CEJ, Skaf JA, Al-Alaoui MA (2003) Arabic speech recognition using recurrent neural networks. In: Proceedings of the 3rd IEEE international symposium on signal processing and information technology, pp 543–547. https://doi.org/10.1109/ISSPIT.2003.1341178
6. Shi B, Bai X, Yao C (2016) An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. IEEE Trans Pattern Anal Mach Intell 39:2298–2304. https://doi.org/10.1109/TPAMI.2016.2646371
7. Elbashir MK, Mustafa ME (2018) Convolutional neural network model for Arabic handwritten characters recognition. Int J Adv Res Comput Commun Eng 7(11). https://doi.org/10.17148/IJARCCE.2018.7111
8. Tagougui N, Boubaker H, Kherallah M, Alimi AM (2014) A hybrid NN/HMM modeling technique for online Arabic handwriting recognition. CoRR (Online) https://arxiv.org/abs/1401.0486
9. Younis KS (2017) Arabic handwritten character recognition based on deep convolutional neural networks. Jordanian J Comput Inf Technol 3(3). https://doi.org/10.5455/jjcit.71-1498142206
10. Torki M et al (2014) Window-based descriptors for Arabic handwritten alphabet recognition: a comparative study on a novel dataset. arXiv preprint arXiv:1411.3519
11. El-Sawy A, Loey M, EL-Bakry H (2017) Arabic handwritten characters recognition using convolutional neural network. WSEAS Trans Comput Res 5:11–19
12. Sahlol A, Suen C (2013) A novel method for the recognition of isolated handwritten Arabic characters. Technical report. Cornell University
13. Alamri H, Sadri J, Suen CY, Nobile N (2008) A novel comprehensive database for arabic off-line handwriting recognition. In: 11th international conference on frontiers in handwriting recognition 2008. Montreal
14. Abdalkafor AS, Sadeq A (2016) Arabic offline handwritten isolated character recognition system using neural network. Int J Bus ICT 2:41–50
15. Mudhsh M, Almodfer R (2017) Arabic handwritten alphanumeric character recognition using very deep neural network. Information 8(3):105. https://doi.org/10.3390/info8030105
16. Abdelazeem S, El-Sherif E (2017) The Arabic handwritten digits databases ADBase & MADBase. Available online: https://datacenter.aucegypt.edu/shazeem/. Accessed on 24 Aug 2017
17. Lawgali A, Angelova M, Bouridane A (2013) HACDB: handwritten Arabic characters database for automatic character recognition. In: 4th European workshop on visual information processing (EUVIP), pp 255–259. https://doi.org/10.4108/eai.18-7-2019.2287842
18. El-Sawy A, EL-Bakry H, Loey M (2016) CNN for handwritten Arabic digits recognition based on LeNet-5. In: Proceedings of the international conference on advanced intelligent systems and informatics 2016, pp 566–575. https://doi.org/10.1007/978-3-319-48308-5
19. Elleuch M, Maalej R, Kherallah M (2016) A new design based-SVM of the CNN classifier architecture with dropout for offline Arabic handwritten recognition. Proc Comput Sci 80:1712–1723. https://doi.org/10.1016/j.procs.2016.05.512
20. Elleuch M, Tagougui N, Kherallah M (2016) A novel architecture of CNN based on SVM classifier for recognizing Arabic handwritten script. Int J Intell Syst Technol Appl 15(4):323–340. https://doi.org/10.1504/IJISTA.2016.10000779
21. Yin F, Wang QF, Zhang XY, Liu CL (2013) ICDAR Chinese handwriting recognition competition. In: 2013 12th ICDAR on document analysis and recognition
22. Lawgali A, Angelova M, Bouridane A (2013) HACDB: handwritten Arabic characters database for automatic character recognition. In: 2013 4th European workshop on visual information processing (EUVIP), pp 255–259. https://doi.org/10.4108/eai.18-7-2019.2287842

23. Mishra A, Alahari K, Jawahar CV (2012) Scene text recognition using higher order language priors. BMVC. https://doi.org/10.5244/C.26.127
24. Wang K, Babenko B, Belongie S (2011) End-to-end scene text recognition. In: ICCV, 2011. https://doi.org/10.1109/ICCV.2011.6126402
25. Lucas SM, Panaretos A, Sosa L, Tang A, Wong S, Young R, Ashida K, Nagai H, Okamoto M, Yamamoto H, Miyao H, Zhu J, Ou W, Wolf C, Jolion J (2003) ICDAR 2003 robust reading competitions: entries, results, and future directions. IJDAR. https://doi.org/10.1109/ICDAR.2003.1227749
26. Alsaeedi A, Al Mutawa H, Natheer S, Al Subhi W, Snoussi S, Omri K (2018) Arabic words recognition using CNN and TNN on a smartphone. In: IEEE 2nd international workshop on Arabic and derived script analysis and recognition, pp 57–61. https://doi.org/10.1109/ASAR.2018.8480267
27. Yousfi B (2016) Contribution of recurrent connectionist language models in improving lstm-based Arabic text recognition in videos. Pattern Recogn. https://doi.org/10.1016/j.patcog.2016.11.011
28. Mohsin A (2020) Developing an Arabic handwritten recognition system by means of artificial neural network. J Eng Appl Sci. https://doi.org/10.36478/jeasci.2020.1.3
29. Schäfer AM, Udluft S, Zimmermann HG (2006) Learning long term dependencies with recurrent neural networks. In: Proceedings of the 16th international conference on artificial neural networks (ICANN 2006), vol 4131. https://doi.org/10.1007/11840817_8
30. Bengio Y, Simard PY, Frasconi P (1994) Learning long-term dependencies with gradient descent is difficult. IEEE Trans Neural Networks 5(2):157–166. https://doi.org/10.1109/72.279181
31. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735
32. Graves A, Fernández S, Gomez F, Schmidhuber J (2006) Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In: Proceedings of the 23rd international conference on machine learning, vols 12, 13, 16 and 20. ACM, pp 369–376. https://doi.org/10.1145/1143844.1143891
33. Schuster M, Paliwal K (1997) Bidirectional recurrent neural networks. IEEE Trans Signal Process 45:2673–2681
34. Taylor C, Rogers G, Hughes Z, Ceyhun O, MatLab vs. Python vs. R (2017) Business faculty publications. 51. https://scholar.valpo.edu/cba_fac_pub/51
35. Abadi M, Agarwal A (2016) TensorFlow: large-scale machine learning on heterogeneous distributed systems. arXiv:1603.04467v2
36. Yan R, Peng L, Xiao S, Johnson MT, Wang S (2019) Dynamic temporal residual network for sequence modeling. Int J Doc Anal Recogn (IJDAR). https://doi.org/10.1007/s10032-019-00328-x
37. Awni M, Khalil MI, Abbas HM (2019) Deep-learning ensemble for offline Arabic handwritten words recognition. In: 2019 14th international conference on computer engineering and systems (ICCES). https://doi.org/10.1109/icces48960.2019.9068184
38. Maalej R, Kherallah M (2019) Maxout into MDLSTM for offline Arabic handwriting recognition. In: Gedeon T, Wong K, Lee M (eds) Neural information processing. ICONIP 2019. Lecture notes in computer science. Cham. https://doi.org/10.1007/978-3-030-36718-3_45