Historical Arabic Manuscripts Text Recognition Using Convolutional Neural Network

Najla Alsaedi
Department of Computer Science
Taibah University
Al-Madinah, KSA
nsm1416@gmail.com

Bodour Alrehali
Department of Computer Science
Taibah University
Al-Madinah, KSA
bodour__@hotmail.com

Hanan Alahmadi
Department of Computer Science
Taibah University
Al-Madinah, KSA
Hananaa79@gmail.com

Nahla Abid
Department of Computer Science
Taibah University
Al-Madinah, KSA
nabd@taibahu.edu.sa

***Abstract*— The Islamic heritage is rich of Arabic manuscripts that contain valuable knowledge of Islamic Sciences, such as Hadeethe, Tafseer and Akhidah. However, these manuscripts are hard to read and there is a need to convert them into a publishable form. Therefore, this paper proposes a method for recognizing the text in the images of these manuscripts and convert it into a readable text that can be copied and saved for further usage in other researches. The main steps of our algorithm are as follow: 1) enhancing the image (preprocessing); 2) dividing the manuscript image into lines and characters (segmentation);3) building the dataset of Arabic characters;4) recognizing the text (classification). In the classification stage, we apply Convolutional Neural Network CNN on three created datasets, and it provides an accuracy that ranges between 74.29% to 88.20%.**

Keywords*— *Optical Character Recognition; Arabic manuscripts; Text recognition; CNN.

I. Introduction

Arabic text recognition has attracted the attention of many researchers in the last decades. However, Arabic text recognition is a challenging task due to the complexity of Arabic writing. This complexity comes from the cursive nature of Arabic language and the addition of diacritics to distinguish similar letters [1].

Several approaches have been proposed to extract Arabic text that is written by hand [1, 2, 4, 13-16] or generated by a computer [1, 2, 6-12]. However, very limited work has been done to recognize Historical Arabic Manuscripts Text [13-17]. Here, we propose a method for Historical Arabic Manuscripts Text Recognition. In the experiment presented here, we use old Arabic manuscripts that written by scientists of Islam in the period of 7-8 Hijri centuries.

The purpose of our work is to recognize the text from the manuscript image which evolves a series of operations. These operations include preprocessing the image, dividing the image to sub parts (e.g. lines or characters), and identify the text in these sub parts that include characters.

The main contribution of our work is as twofold:

- Creating three datasets of Arabic letters that extracted from the manuscripts of the period of 7-8 Hijri centuries.
- Applying Convolutional Neural Network CNN on the three created datasets. The produced accuracy ranges between 74.29% to 88.20%.

II. Related Work

Text recognition has been a topic of interest for many researchers. English text recognition is done for printed and handwritten text [5, 6]. Many papers target text recognition of different languages such as Malayalam, the language of south India [7], Chinese [8, 9] and Urdu [10]. Recognition of Arabic text has been a growth interest topic for researchers in the last decades [3] for both printed [11, 12] and hand written text[1, 2, 4, 13-16]. Howver, a few papers draw the attention to historical manuscripts [17-21] and they are characterized by their bad quality because of materials used for their storage. Kefala et al. work on enhancing the quality of the damaged manuscript image [21]. They have provided a comparative study of different binarization algorithms for the images of historical Arabic manuscripts.

On the other hand, Boulid et al. have focused on the detection of Arabic text lines [1]. This process can be categorized as a segmentation process, where the connected components that belong to the same line are detected. Our work moves beyond detecting the text lines only and target the problem of text extraction.

Aouadi and Echi proposed an approach for preprocessing manuscripts by recognize the separator words and separate them from neighboring words [20].

Arabic manuscripts written in different ages and different handwritten styles. Al-Aziz developed a method to classify the Arabic manuscripts based on their ages [19]. Spatial Gray Level Dependence SGLD was applied to distinguish between manuscripts in different writing styles to three different ages: Contemporary (Modern) Age, Ottoman Age and Mamluk Age. Moreover, Adam et al. have recognized and classified old Arabic manuscripts based on handwritten styles [18]. The handwritten styles that recognized was Diwani, Kufic, Naskh, Farsi, Ruq'ah and Thuluth. Unlike the work done in [19] and [18], our work focuses on recognizing the text from Arabic historical manuscripts rather than classifying them based on age or writing style.

In Arabic manuscripts text recognition filed, Khorsheed provided a method of recognizing handwritten Arabic manuscripts using Hidden Markov Model HMM [17]. This method extracted the structural features from the manuscript word and trained the HMM with these features. We provide a compression between our work and their work in section VII.

As stated above, there is minimal work on extracting text from Arabic manuscript and existing papers usually deals with classifying them based on the writing style [18]. Here, we experiment the use of Applying CNN to extract text from Arabic manuscript. In recent years, CNN has proved its success in image classification tasks [22]. Therefore, we decide to use CNN in our approach to classify Arabic character images.

978-1-7281-2746-0/20/$31.00 ©2020 IEEE
DOI 10.1109/CDMA47397.2020.00012

III. Arabic Language Characteristics

In the following section, we describe the characteristics of Arabic language and their letters. This critical to identify before the process of segmentation.

A. The cursive nature of the language writing.

Letters of the same word are connected to each other. Each Letter is connected to the letter before and after in a word but six letters, ا ، د ، ذ ، ر ، ز ، و, cannot be connected to the letter after even if they are in the same word [3].

B. Arabic language contains 28 characters.

Each character has between 2 to 4 shapes based on the position of the letter in the word (end, middle, start) or standing alone [23]. Fig. 1 shows the different shapes of the letter "ص".

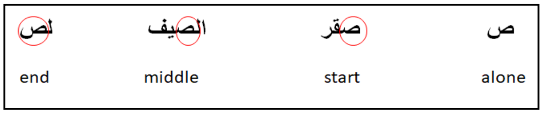


Fig. 1. The different shapes of the letter "ص" based on the position of it in the word.

C. Some letters have the same main structure.

They only differ from each other by the presence, or absence, the position and the number of dots putted on the letter [24]. An example of these letters shown in Fig. 2.

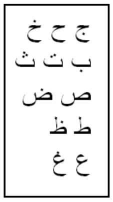


Fig. 2. Example of letters have the same main structure but differ in the dots putted on them.

These characteristics are taken into consideration during the segmentation, dividing words into letters. For example, for letters that have different shapes based on the location, we include all different shapes in our dataset associated with that letter as shown in TABLE I.

IV. Data acquisition

All images of the manuscripts in this project are acquired from Dr. Abdulbari Alansari, who is an associate professor at Hadeeth Science at Islamic University in Madinah. He provided us with a collection of images of the Arabic manuscripts for many scientists of Islam in the period of 7-8 Hijri centuries. Fig. 3 shows samples of these manuscripts.

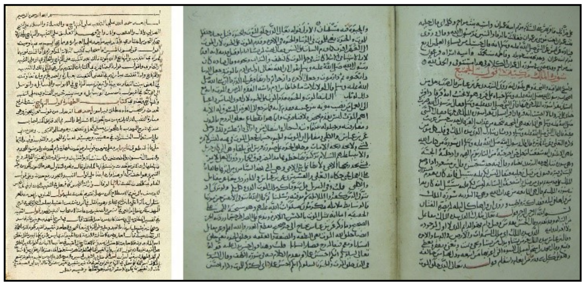


Fig. 3. Two different manuscripts for different scientists from our collection of manuscripts.

V. Proposed methodology

There are five main stages of the proposed approach. The input of our system is an image of one page of the manuscript. This image goes through the following steps:

1) *Preprocessing:* The image is transformed to the binary representation.

2) *Line segmentation:* The binary image is segmented to lines. A line is a set of Arabic words written in a single horizontal line.

3) *Character segmentation:* Each line from step two is segmented to characters of Arabic letter.

4) *Dataset:* Our dataset of Arabic letters is created.

5) *Classification:* The dataset is used to train and validate CNN classifier.

Fig. 4 shows the flow diagram of our proposed methodology.

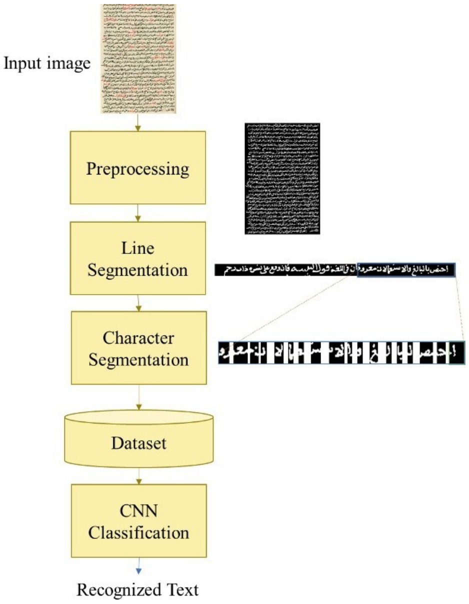


Fig. 4. The flow diagram of our proposed methodology.

A. Preprocessing

The main goal of preprocessing step is to improve the image and to facilitate the recognition stage. It starts by converting the whole color image into grayscale image. Then, the manuscript image is entered to the next step to generate the binary image and represent it as a single matrix. In the generated matrix, the white pixels are form the foreground (e.g. the text) and black pixels are form the background. Fig. 5 illustrates the preprocessing stages.

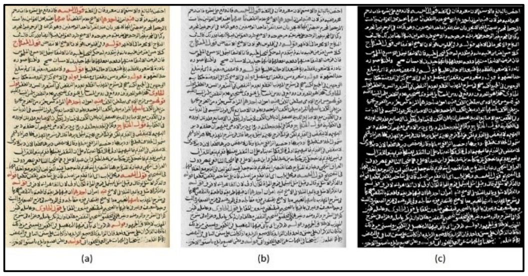


Fig. 5. (a) Original image. (b) Grayscale image. (c) Binary image.

The following step is segmentation that partitions each entity in the image to detect letters in the Arabic language. Here, we perform segmentation at two levels: line level and character level. The output of the segmentation is a set of images each of which represents a single character.

B. Line segmentation

Line segmentation is achieved via horizontal Projection Profile (PP) for the image. Horizontal PP is a method used to convert the image from 2D to 1D by calculating the densities of each row and draw the histogram of the image [22]. We use this method to crop out each line at the points of lower density. Fig. 6 shows the horizontal PP for the whole page, which is used to cut out each line.

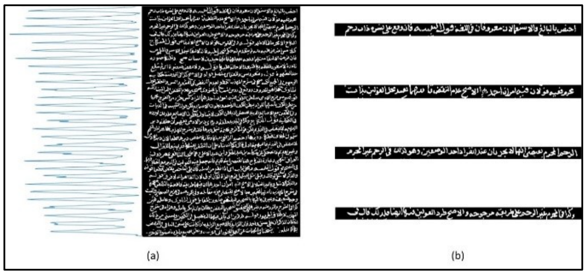


Fig. 6. (a) Horizontal Projection Profile for the whole page image. (b) The segmentation of the first four lines.

C. Character segmentation

Character segmentation is performed using vertical PP. Vertical PP is the same as horizontal PP, but the direction of computing densities is different. In horizontal PP, we calculate the densities for each row. However, in vertical PP the calculation of densities is done for each column. Then, this method is used to cut each character in the line at lower densities of the horizontal PP. This method is used by Sarafraz et.al. [12]. Fig. 7 shows how vertical PP is used to separate characters of each line in a sentence.

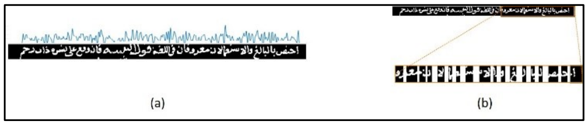


Fig. 7. (a) : Vertical PP for a line. (b) : Character segmentation for the first characters in the line.

D. Dataset

We have created three datasets in order evaluate the classifier. These datasets contain image for Arabic letters at different states (e.g. start, middle, end and standalone), as shown in TABLE I.

In addition, it contains images of two/three connected letters. We add these letters because these connected letters are hard to break down into two elements (one letter is written on top another letter). Additionally, these letters are repeated in several pages of the manuscript. Therefore, we choose to include them in the created dataset. TABLE II shows these connected letters.

TABLE I. SAMPLE OF THE FIRST DATASET. IT SHOWS IMAGES FOR EACH CHARACTER AT DIFFERENT STATES.

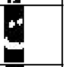
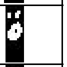
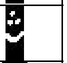
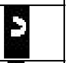
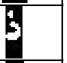
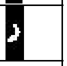
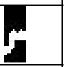
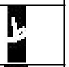
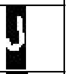
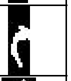
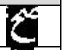
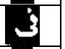
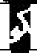
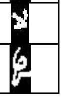
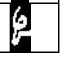
Arabic Letters	start	middle	end	alone
ا		-------		-------
ب			-------	
ت		-------	-------	
ة	-------	-------		
ث	-------	-------	-------	
ج		-------	-------	
ح		-------		
خ		-------	-------	-------
د	-------	-------		
ذ	-------	-------	-------	
ر	-------	-------	-------	
ز	-------	-------	-------	-------
س		-------	-------	
ش	-------	-------	-------	-------
ص		-------	-------	-------
ض		-------	-------	-------
ط	-------	-------	-------	
ظ	-------	-------	-------	
ع				-------
غ	-------		-------	-------
ف			-------	-------
ق			-------	
ك		-------	-------	-------
ل			-------	
م		-------	-------	
ن			-------	
هـ				-------
و	-------	-------	-------	
ي		-------	-------	

TABLE II. CONNECTED LETTERS THAT USED IN OUR DATASET.

Connected letters	image
صح	
على	
في	
كو	
لا	
ها	

First Dataset

The first dataset has all 28 Arabic letters with average of 40 images per letter. Total number of images is 2240 images.

Second Dataset

The second dataset has partial letters from the first one. It includes 10 letters. These letters are shown in TABLE III. However, we increase the number of images per letter. On average, there are 100 images per letter.

Third Dataset

The third dataset has the same letters presented in the second dataset with average of 200 images per letter.

TABLE IV shows total number of images in each dataset. We created the second and third datasets in order to test our model using higher number of images per class.

The data within dataset is randomly divided into training dataset and validation dataset. 85% of the data is used for training and 15% of the data is used for validation. This division is applied to each one of the three created datasets.

TABLE III. THE 10 CLASSES THAT USED IN THE SECOND AND THIRD DATASETS.

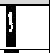
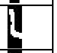
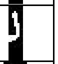
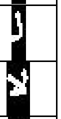
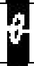
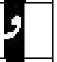
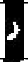
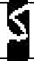
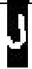
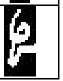
Arabic character	image
ا	
ـا	
لـ	
لا	
ـهـ	
و	
ر	
ک	
ل	
ها	

TABLE IV. THE TOTAL NUMBER OF ELEMENTS IN EACH DATASET.

	Number of classes	Number of samples per class	Total number of samples
Dataset1	56	40	2240
Dataset2	10	100	1000
Dataset3	10	200	2000

E. CNN Classification

Convolutional Neural Networks (CNNs) are feedforward networks. In particular, the data flows within CNN in one direction only. In general, CNN consists of one or more convolutional and pooling layers followed by one or more fully connected layers. Fig. 8 shows the general CNN architecture for an image classification task [22].

The input to the network is the image itself, and it is followed by several layers of convolution and pooling. The convolution layers act as features extractors. The output of the convolution layer is a set of feature maps. Each neuron in the convolution layer is connected to a set of neurons from the previous layer via a set of weights. Weights are adjusted during learning by a nonlinear activation function. In the past, sigmoid and hyperbolic tangent functions were used. Recently, the most popular activation function is Rectified Linear Unit (ReLU) [22].

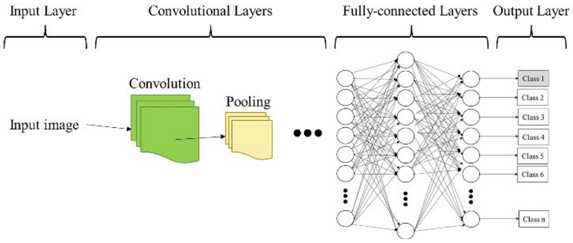


Fig. 8. The general CNN architecture for an image classification task. Adapted from [22].

The spatial resolution of the feature maps is reduced by the pooling layers. There are two types of pooling: average pooling and maximum pooling. In average pooling, the average of the current field is propagated to the next layer while in max pooling the maximum value within the current field is propagated to the next layer [22].

The output of the last convolution and pooling is feed to one or more fully connected layers. The fully connected layers are followed by an activation function that normalizes the output and produces positive numbers that sum to one, which can be used as classification probabilities to identify the class label. The most common activation function in this context is Softmax function [22].

We define the CNN architecture for Arabic characters images classification as follow:

1) Input Layer: The input is an image that represents a single character from the historical Arabic manuscripts. We specify the image size to be 30-by-30-by-1 for all input images. The height and the width for the input image is equal to 30 pixels. As our images are grayscale images, the channel size is 1.

2) Convolutional Layers: In our architecture, we have three convolutional layers. Each layer is followed by a nonlinear activation function, which is ReLU. The number of neurons varies in each convolutional layer. Particularly, the first layer has 8 neurons, the second layer has 16 neurons, and the third layer has 32 neurons.

3) Max Pooling Layers: Each convolutional layer is followed by a down-sampling operation that reduces the spatial resolution of the feature map. One way of down-sampling is to use max pooling, which returns the maximum values of rectangular fields of inputs. The two arguments that specify max pooling operation are pool size and step size. Pool size is the size of the rectangular field of the input. Step size specifies the size that the training function takes as it scans along the input. We make both pool size and step size to equal 2.

4) Fully Connected Layer: In this layer, the neurons connect to all neurons in the previous layer in order to combine all features learned by the previous layers to identify class label. Therefore, the number of neurons in this layer is equal to the number of classes in the target dataset.

5) Output Layer: This layer uses the probabilities returned by the Softmax activation function for each input and assign the input to one of output classes.

VI. EVALUATION METRIC

In order to compare between the character generated by such a classifier and the actual character of the image, we use the accuracy score. Accuracy is calculated by equation (1).

$$Accuracy = (TP+TN)/(TP+TN+FP+FN) \times 100 \quad (1)$$

TP, TN, FP and FN are true positive, true negative, false positive and false negative, respectively.

VII. RESULT AND DISCUSSION

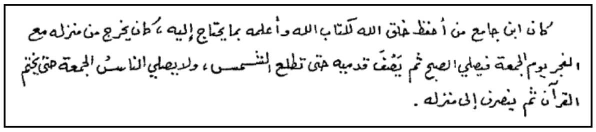
We apply CNN classification on the three datasets and compute the accuracy resulted from applying CNN on each dataset. Dataset1, which contains 56 classes with 40 images per class, gives an accuracy of 74.29%. On the other hand, Dataset2 and Dataset3 give higher accuracies: 84.67% and 88.20%, respectively.

In these two datasets, we aim to decrease the number of classes and increase number of images within the class to check classification accuracy. As we expect, as increasing the number of samples per class, accuracy increases. TABLE V shows the testing results for the CNN with the three datasets. Using the first dataset that has all Arabic letters, our system achieve accuracy as good as 74.29%. This result is very promising. In the second and third dataset, the result has improved as the model is trained with higher number of images per class.

TABLE V. THE TESTING RESULTS FOR THE CNN WITH THE THREE DATASETS.

Dataset ID	Accuracy
Dataset1	74.29%
Dataset2	84.67%
Dataset3	88.20%

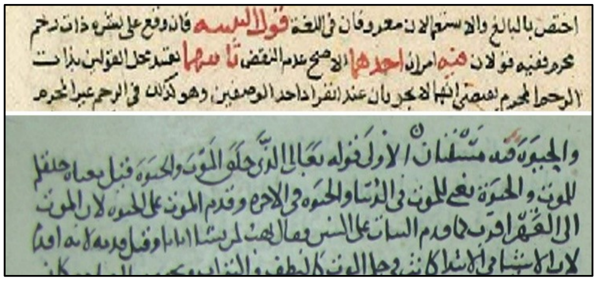
As we illustrated previously, Khorsheed has provided a method for recognizing handwritten Arabic manuscripts using HMM [17]. This method extracted the structural features from the manuscript word and trained the HMM with these features. The accuracy achieved was 97%. However, this paper [17] used only one manuscript which is a sample of it shown in Fig. 9. This sample is very similar to normal hand writing.



كان ابن جامع من أحفظ خلق الله لكتاب الله وأعلمه بما يحتاج إليه، كان يخرج من منزله مع
الفجر يوم الجمعة فيصلي الصبح ثم يَصُفّ قدميه حتى تطلع الشمس، ولا يصلي الناس الجمعة حتى يختم
القرآن ثم ينصرف إلى منزله .

Fig. 9. A sample of the manuscript that has used in [17].

In our paper, we aim to recognize Arabic characters from historical Arabic manuscripts using CNN. We consider each character at different locations in the word. Moreover, we use a collection of manuscripts for two different scientists. Two samples from this collection are shown in Fig. 10.



اختص بالبالغ والاستعمالان معروفان في اللغة قول المسه فان وقع على بشره ذات رحم
محرم ففيه قولان فيه امران احدها الاصح عدم النقض ثانيها نقض لحمل القولين بذات
الرحم المحرم يقتضي انها لا يجري ان عند انفراد احد الوصفين وهو كذلك في الرحم غير المحرم

والحيوة فيه مسئلتان / الاولى قوله تعالى الذي خلق الموت والحيوة قيل معناه خلق
للموت والحيوة نفع للموت في الدنيا والحيوة في الاخرة وقدم الموت على الحيوة لان الموت
الى القهر اقرب [illegible]
[illegible]

Fig. 10. Tow samples for tow different scientists from our collection.

VIII. LIMITATION

We assume the manuscripts written on a baseline in straight away. Therefore, any manuscript its phrases written randomly, or some words lagged of the baseline, our system may not be able to recognize it.

If the damage is great, distorts the words of the manuscript made it unreadable at all. Preprocessing step may not be able to handle this type of damage.

IX. CONCLUSION AND FUTURE WORK

It is known that historical Arabic manuscripts are complex to read. This complexity comes from the addition of diacritics to distinguish similar letters also from the cursive nature of Arabic language. This paper proposes a method for recognizing the text from manuscript image using Convolutional Neural Networks (CNNs). This includes preprocessing of the manuscript's image, segmentation of the lines and characters in the manuscript, building a dataset that contains images for each character at different shapes and classification of each character using CNN. The accuracy we gained from CNN on the three created datasets range between 74.29%-88.20%.

At this point, our approach works at the character level. Actual word is not being identified. As future work, we plan to enhance our work to identify words in the processed image. Additionally, we plan to enhance our dataset and has larger number of images per class. Finally, we plan to enhance the recognition by using other segmentation techniques such as skeletonization.

REFERENCES

[1] Y. Boulid, A. Souhar, and M.Y. Elkettani, “Detection of Text Lines of Handwritten Arabic Manuscripts using Markov Decision Processes,” International Journal of Interactive Multimedia and Artificial Intelligence, 2016, pp. 31-36.

[2] R. Mouhcinea, A. Mustaphaa, and M. Zouhirb, “Recognition of cursive Arabic handwritten text using embedded training based on HMMs,” Journal of Electrical Systems and Information Technology, 2018, pp. 245-251.

[3] B. Qacimy, A. Hammouch, and M.A. kerroum, “A Review of Feature Extraction Techniques for Handwritten Arabic Text Recognition,” in 1st International Conference on Electrical and Information Technologies, 2015, IEEE, pp. 241-245.

[4] R. Haraty, and C. Ghaddar, “Arabic Text Recognition,” The International Arab Journal of Information Technology, 2004, pp. 156-163.

[5] S. S. Dutt, and J.D. Amin, “Printed English Character Recognition using Feature based Matching and Error Correction: A Survey,” International Journal Of Innovative Research In Technology, 2015, pp. 508-516.

[6] A. Yuan, et al., “Offline handwritten English character recognition based on convolutional neural network,” in 10th IAPR International Workshop on Document Analysis Systems, 2012, IEEE: Gold Cost, QLD, Australia.

[7] J. John, P.K. V, and K. Balakrishnan, “Handwritten Character Recognition of South Indian Scripts: A Review,” National Conference on Indian Language Computing, 2011, pp. 2-6.

[8] R. Messina, and J. Louradour, “Segmentation-free handwritten Chinese text recognition with LSTM-RNN,” in 13th International Conference on Document Analysis and Recognition (ICDAR), 2015, IEEE: Tunis, Tunisia.

[9] Q. Wang, F. Yin, and C.-L. Liu, “Handwritten Chinese Text Recognition by Integrating Multiple Contexts,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, pp. 1469 - 1481.

[10] S. Naz, et al., “Zoning Features and 2DLSTM for Urdu Text-line Recognition,” Procedia Computer Science, 2016, pp. 16-22.

[11] Al-Muhtaseb, et al., “Recognition of off-line printed Arabic text using Hidden Markov Models,” Signal Processing, 2008, pp. 2902-2912.

[12] M. Sarfraz, S.N. Nawaz, and A. Al-Khuraidly, “Offline Arabic Text Recognition system,” in International Conference on Geometric Modeling and Graphics. 2003.

[13] S. Al-Ma’adeed, D. Elliman, and C. Higgins, “A Data Base for Arabic Handwritten Text Recognition Research,” The International Arab Journal of Information Technology, 2004, pp. 117-121.

[14] A. AL-Shatnawi and K. Omar, “Methods of Arabic Language Baseline Detection – The State of Art,” IJCSNS International Journal of Computer Science and Network Security, 2008, pp. 137-143.

[15] H. Hassen and S. Al-Maadeed, “Arabic Handwriting Recognition using Sequential Minimal Optimization,” International Workshop on Arabic Script Analysis and Recognition, 2017, pp. 79-84.

[16] M. Khorsheed, “Automatic Recognition Of words In Arabic Manuscripts,” 2000.

[17] M. Khorsheed, “Recognising handwritten Arabic manuscripts using a single hidden Markov model,” Pattern Recognition Letters, 2003, pp. 2235–2242.

[18] K. Adam, S. Al-Maadeed, and A. Bouridane, “Letter-based classification of Arabic scripts style in ancient Arabic manuscripts,” International Workshop on Arabic Script Analysis and Recognition (ASAR), 2017, pp. 95-98.

[19] A. Al-Aziz, M. Gheith, and A.F. Sayed, “Recognition for old Arabic manuscripts using spatial gray level dependence (SGLD),” Egyptian Informatics Journal, 2011, pp. 37-43.

[20] N. Aouadi and A.K. Echi, “Prior Segmentation of Old Arabic Manuscripts by Separator Word Spotting,” in International Conference of Soft Computing and Pattern Recognition. 2014, pp. 31-36.

[21] A. Kefali, T. Sari, and M. Sellami, “Evaluation of several binarization techniques for old Arabic documents images,” 2010.

[22] W. Rawat and Z. Wang, “Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review,” Neural Computation, 2017, pp. 2352-2449

[23] A. Zoizou, A. Zarghili, and I. Chaker, “A new hybrid method for Arabic multi-font text segmentation, and a reference corpus construction,” Journal of King Saud University – Computer and Information Sciences, 2018

[24] M. Y. Potrus, U.K. Ngahb, and B.S. Ahmed, “An evolutionary harmony search algorithm with dominant point detection for recognition-based segmentation of online Arabic text recognition,” Ain Shams Engineering Journal, 2014, pp. 1129–1139.