International Journal of Engineering & Technology, 8 (1.11) (2019) 9-16

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

An OCR System for Arabic Calligraphy Documents

Hassanin Al-Barhamtoshy[1], Kamal Jambi[1], Hany Ahmed[2], Shaimaa Mohamed[3], Mohsen Rashwan[2], Sherif Abdou[3]

[1] *Computing and Information Technology, King Abdulaziz University, KAU, Jeddah, Saudi Arabia*
[2] *Faculty of Engineering, Cairo University, Egypt*
[3] *Computing and Information Science, Cairo University, Egypt*
**Corresponding author E-mail: hassanin@kau.edu.sa*

Abstract

Market of Arabic OCR for old documents is large and its deservers higher attention even more than the modern documents, which in many cases are already distributed in digitized format. Therefore, in this paper, we address the challenges of Arabic OCR for old and calligraphy documents.
This paper introduces an integrated development Arabic OCR system to get good accuracy for recognizing old and calligraphy documents. While our developed system has provided accurate results for modern Arabic documents, when we used that system for old Arabic documents, we got a steep degradation in performance, (around 25% accuracy compared with 85% for modern Arabic documents).
We made three main modifications for our Arabic OCR system. First, we eliminate the word segmentation step and run the OCR process on the complete line. With this modification, we managed to avoid large number of segmentation errors on the word level but had to change our recognition approach to be a dictionary based. The second modification, we changed the used features to histogram gravity based one. This type of feature provided much better performance especially in the challenging cases of old documents such as low-quality printing effects, wavy baselines and heavy noisy documents. Third, we used a hybrid model that integrate Neural Networks with HMM to provide a better discrimination between the shapes of Arabic ligatures. The paper starts with a fast review for the baseline system and the added enhancements of such OCR. Then introduces the new developed OCR efforts for Arabic old and calligraphy documents.
In other words, this paper describes a proposed approach based on language model with ligature and overlap characters for the proposed Arabic OCR. Therefore, a posterior word-based approach is used with tri-gram model to recognize the Arabic text. Features are extracted from images of words and generated pattern using the proposed solution. We test our proposed OCR system in different categories of Arabic documents: early printed or typewritten, printed, historical and calligraphy documents. The test bed of our system gives 12.5%-character error rate compared to the best OCR of other systems.

Keywords: *ARABIC; OCR; SEGMENTATION; RECOGNITION; HMM, NN; DNN.*

1. Introduction

Today, OCR systems achieve important role in document analysis and content retrieving with high accuracy. However, Arabic document analysis and Arabic retrieving systems have more challenges (Stahlber et al, 2016).
Arabic OCR is a very challenging application, due to writing nature: letter connectivity, cursiveness, context shape, ligatures, diacritics, dots, and it has many varieties in font names and styles (Abdelaziz et al, 2015; Elad et al, 2006). Moreover, the Arabic alphabet contains 28 letters. Each letter can be written between two and four shapes, depending on the position of that letter in the word. Most of Arabic fonts include complex ligatures shapes that consist of "consecutive letters of variable lengths as well as inter- and intra-word spaces". The "problem of diacritical points can change the meaning of the word or sub-word" (Abdel Azeem & Ahmed, 2013). Figure 1 shows sample of the challenges in Arabic scripts. Due to these properties the current performance of state of art Arabic OCR systems is much lagging compared with the performance of other Latin-based OCR systems. In this paper, we targeted the advancing of Arabic OCR systems to achieve a practical level of accuracy (in the range 90-95%) with Omni-font capabilities to process unseen fonts.
Due to characteristics of the Arabic writing, Arabic letters are written in many forms. Each Arabic letter has up to 4 or 5 forms, according to its location in the Arabic word. It may be at the starting of a word, middle, ending, or separate (Althobaiti H. and Lu C., 2017.
Location of ligatures can be defined by vertical histogram during text segmentation (Althobaiti H. and Lu C., 2017), but this process is not sufficient because of multi-ligature overlapping. Text segmented lines were determined into connected components (Lehal G. S., 2013) through heuristic approach with 99.02% accuracy. This in addition to a study segments lines into connected components through two steps; primary and secondary with accuracy 92.5% (Din I. U. et al, 2016).
An "Arabic documents layout analysis" have been introduced. In this paper, the document is segmented into set of zones using morphological operations, and such zones are classified using SVM into text and non-text zones (Hesham A. et al, 2017). Another works sending the segmented words from the segmented zone to an OCR's recognition phase to recognize the text, the results illustrates the accuracy rate is around 93.2% for printed documents, (Hesham A. et al, 2017).

Copyright © 2019 Authors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The remainder of the paper is systematized as the following. Section2 introduces to the new "proposed model for Arabic OCR system", in addition to, pertinent description of the algorithms that handle the old and calligraphy documents using hybrid HMM and DNN models are described in detail. Therefore, the implementation and related criteria within the hybrid HMM and DNN are illustrated in Section 3. Experimental testing and results will be pronounced in Section 4. Section 5 includes concluded remarks and future work prospective.



Fig. 1: Arabic Script OCR Challenges.

2. Proposed Model for Arabic OCR

Some of the Arabic OCR systems, the page is segmented into lines; each line is segmented to words. Each word is modeled as a sequence of ligature HMM models. The core system is built around the idea of the apparent analogy between the Arabic font/type-written OCR problem and the Automatic Speech Recognition (ASR) problem as systems use HMM to simultaneously segment and recognize the connected phonemes in continuous speech (Attia et al, 2015; Rashwan et al, 2012; Al-Barhamtoshy & Rashwan, 2014). Our OCR is using HMM to simultaneously segment/recognize the connected graphemes in the Arabic old and calligraphy scripts. Figure 2 illustrates this idea.

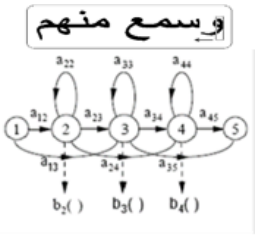


Fig. 2: The HMM model and state sequence

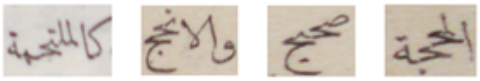


Fig. 3: Sample Arabic Ligatures

As many of the used fonts for cursive Arabic scripts, extensively use ligatures where two or more letters are joined as a single glyph, which complicates the character level segmentation. Figure 3 shows some samples of Arabic compound ligatures. Therefore, a ligature data-driven approach model is proposed with a language model taken into consideration, the Arabic written peculiarities.

The paper handles many problems related to the OCR two main processes: segmentation and recognition phases. These problems include (1) old and calligraphy Arabic documents cursive nature, (2) Arabic characters variation sizes, (3) ligatures combinations with vertical overlapping, diacritics and similar characters shape with changing in the number of dots or diacritics.

Many algorithms are used for documents segmentation and recognition. But actually, the evaluated performance were used in most are Arabic scripts without old or calligraphy documents. They gave poor accuracy results with maximum 71.5% (Hesham A. et al, 2017). In contrast, we are handling with complete line segment instead of single word segment at both levels of document analysis (segmentation and recognition).

The developed OCR system consists of two modules. The first one is the training module where the holistic features are extracted from the training set of the word images. The extracted features are used to build set of clusters of the similar word shapes. The generated words' clusters and their extracted features represent the knowledge base that is used in the recognition phase. The second module is the recognition module. In that module, after applying the preprocessing operations on the input image, the detected text blocks are segmented to lines and words. The features are extracted for each word image. Then the word cluster, or best-n clusters that have the minimum Euclidean distance with the test image vector (Al-Barhamtoshy, 2016), is assigned.

The recognition phase deals with the whole word as a particular inseparable component. An overall feature vector is extracted from the input word, which is then used to classify it against a stored dictionary of words (Srimany et al, 2014). The main challenge is to develop an Arabic OCR system that is considered to be efficient for a large vocabulary size according to Arabic language varieties and peculiarities (as illustrated in section 1). To achieve this objective, we developed a lexicon reduction approach that cluster the similar shape words to reduce the word recognition time to make that approach a practical one. That system used a hybrid of several holistic features that combine global word level Discrete Cosine Transform (DCT) based features and locale block based 4B-DCT features. Using this type of features, the system managed to achieve Omni-font performance with font and size independence. Figure 4 shows the architecture of the developed Holistic Arabic OCR system.

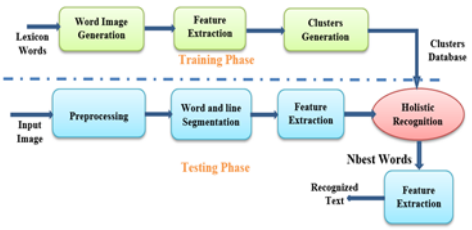


Fig. 4: Block Diagram of the Arabic OCR System

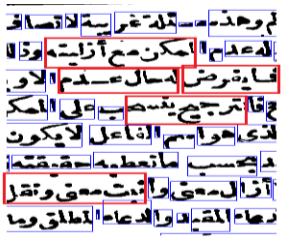


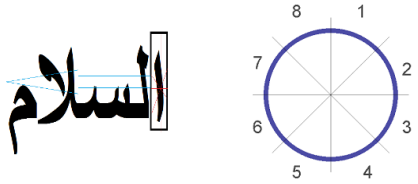
Fig. 5: Sample word segmentation errors

2.1. Complete Line Recognition System

With our first approach, the process is to segment the line images into words then use such words for recognition. The main advantage of this approach is its efficiency, since the recognition process is localized on word level. In addition, it provides an open vocabulary system since the system can compose any word using its ligature sequence (Abdelaziz et al, 2016). The disadvantage of such approach is the word segmentation errors that can happen due to the overlapping that occurs in many Arabic fonts, which is the hardest challenge in cursive languages. Figure 5 shows samples of segmentation errors from the collected data sets.
To avoid such type of errors we complete line recognition approach. This change lead to some major modifications in our decoding techniques.
When we evaluated the performance of our developed Arabic OCR features, on old Arabic documents, calligraphy documents, early printed documents or modern Arabic documents with high level of noise we got large degradation in the system performance. The error analysis for the OCR results of these documents revealed several reasons for this steep degradation in performance. Such reasons are:

- The regularity in the shape evolution among the segments that represent the different states of an Arabic ligature is not always sustained in old documents.
- The wavy like baselines, which is common in the old Arabic documents would result in the deviation of the features values from there expected ones even after applying some remediation using the de-skewing techniques (Liu et al, 2008).
- The noisy images would result in complete damage in the features values.

To recognize different categories of Arabic texts, especially calligraphy documents, we introduce a new method based on "Freeman Chain Code". Such chain code changing from zero to seven depends on connectivity mechanism that describe the direction of each pixel (clockwise or anticlockwise). Accordingly, for these reasons, we started to investigate the usage of histogram-based features as shown in Figure 6.



(a) Histogram **(b) Polar segment**

Fig. 6: *Histogram based Features with polar segments*

To estimate these features vector, we use the following algorithm:

- Apply a sliding window of 11 pixels in width along the writing direction.
- Calculate the Center of Gravity "CoG" ; Figure 8 (a).
- Divide the frame in 8 equally segments from CoG.
- Count the number of foreground pixels in each segment (0-45, 45-90, 90-135, 135-180, 180-235, 235-270, 270-325, 325-360) as shown in Figure 8(b).
- Shifting 1-pixel along the writing direction.
- Normalize the count by the total number of pixels in the whole frame which provides continues features in the range between zero and one.

2.2. Curved Segmentation Algorithm

The current proposed Arabic OCR system includes an algorithm for Arabic text segmentation, especially for the Arabic calligraphy and handwritten documents. In addition, this algorithm used for any Arabic document categories (early printed or typewritten, printed, and historical). The proposed curved/waved algorithm depends on projection profiles and connected component information. Accordingly, the locations of diacritics and dots (over or below the characters) are addressed and employed. Therefore, the segmentation algorithm finds out a separation/demarcation curved line between every two successive words/text lines. It navigates on horizontal track from right to left, if it finds text pixels; with traversing vertically up or down. Samples of the output results are presented in figure 7.



Before Segmentation **After Segmentation**

Fig. 7: Arabic Calligraphy (Before and After Segmentation)

The detail of the curved/waved segmentation algorithm is described in the following steps.

- Comparing the color of the current pixel with color of the foreground object.

- If match found, then the coordinates of the current pixel is selected as a starting pixel and direction is set to zero.
- Obtaining the chain code of the next pixel according to the following criteria (Ahmad et al, 2017):

$$ND = (PD + 5) \bmod 8 \quad (1)$$

Where ND represents next direction and PD is the previous direction.

Algorithm curved/waved segmentation
Input: Arabic document images
Output: Segmented text lines or segmented words
Begin
Predict text baselines. Any baseline is identified as a max projection profiles (intensity pixels) between two successive split lines.
For each initial split line Do
I=line number; j= 1;
caught =0; /* number of change in j value */
resolve =0; /* number of change in j value */
Store (i , j) as initial coordinates for demarcation segmented line;
Move up or down When find "text pixels"
While (j<= page column width) **And** (baseline$_1$ < i > baseline$_2$) **Do**
Move forward on ith line by increasing j until find "text pixels"
Declare (i, j-1) as demarcation coordinate line;
Store (i, j-1)
Move through (j-1) until a "text pixels"
Store (i/2, j-1) as demarcation coordinate line;
If previous value of j = j-1 Then
caught = caught +1
previous value of j = j-1
If caught (No change in value of y) for 4 times and resolve < 4 Then
Go back 3 pair values in coordinates for demarcation line;
If current row > i in coordinates for demarcation line Then
Set direction = down
Else
Set direction = up
Set i as a previous row coordinates for demarcation line;
Set j = j of previous column coordinates for demarcation line;
previous value of j = j;
Set resolve = resolve + 1 And caught = 0;
If caught (no change in value of j)=4 And resolve=4 Then
Move over info pixels on i row And increase j until information pixels are over.
Set resolve = 0 And caught = 0;
previous value of j = j-1;
j = j +1;
Return by coordinates for demarcation/ split lines
End

3. The Proposed Integrated Model

The characteristic of Arabic old documents presents a higher requirement in the used modeling. Models with high capacity are needed to model the diversity in the types of documents. Recently Deep Belief Networks (DBNs) with several hidden layers were proposed for acoustic modeling in speech recognition because they have a higher modeling capacity per parameter than GMMs (Kirti & Patil, 2013). The recent surge of interest with these effective models have resulted from the introduction of a pre-training stage, which is an unsupervised technique that can lead for an effective initialization of the DNN network parameters. Then a following step of supervised learning tunes the network parameters to optimize its discrimination capability. One of the effective approaches is the combination of the discrimination power of DNN and the dynamic timing modeling power of HMM in a hybrid DNN-HMM model. The context-dependent (CD)-DNN-HMM model that has been proposed for Large Vocabulary Automatic Speech Recognition (LVASR) has resulted in cutting word error rates by up to one third on the interesting conversational speech transcript jobs when compared to the state of art discriminatively trained conventional CD-GMM-HMM systems (Li & Sim 2013; Siniscalchi et al, 2013). A DNN is simply a multi-layer perceptron with large number of hidden layers, that how it got the name of deep networks. The main challenge in training this type of networks is finding efficient training strategies that can avoid falling in poor local optimum that usually results with the complicated nonlinear error surface due to the increased number of hidden layers. A common training to deal with this local optimal challenge is to initialize the DNN parameters greedily and generatively. This can be achieved by treating each pair of layers in the network as a Restricted Boltzmann Machine (RBM) as a pre-training step before doing the whole network training of all the layers. This learning strategy enables the DNN training to start with well initialized weights and makes the global network training a feasible process. An RBM is bipartite graph with two layers, the first one is considered the visible layer and second one is the hidden layer. The elements in the visible layer are only connected to the elements in the hidden layer. The visible layer elements are typically represented using Bernoulli or using Gaussian distributions while the hidden layer elements are commonly represented using Bernoulli distributions. This type of Gaussian–Bernoulli RBMs converts the real-valued stochastic variables (such as the OCR histogram features) to binary stochastic variables that can be processed further using the Bernoulli–Bernoulli RBMs (Siniscalchi et al, 2013). Assumed the typical parameters y, the combined distribution $p(v, h, \theta)$ over the visible elements v and hidden units h in the RBMs can be defined as:

$$p\,(v,\ h\,,\ \theta) = \frac{\exp(-E(v,h,\theta))}{Z(\theta)} \quad (2)$$

where E(v; h; θ) is an energy function and Z(θ) is the normalizing term. For Bernoulli RBMs, we have:

$$E(v,h;\theta) = -\sum_{i=1}^{D}\sum_{j=1}^{F} W_{ij} v_i h_j - \sum_{i=1}^{D} b_i v_i - \sum_{j=1}^{F} a_j h_j \quad (3)$$

$$Z(\theta) = \sum_V \sum_h \exp(-E(v,h;\theta)) \quad (4)$$

The parameters θ = {W; a; b} include the symmetric interaction between the units (W_{ij}) and the bias terms (a_j , b_i). On the other hand, for Gaussian-Bernoulli RBMs, we have:

$$E(v,h;\theta) = -\sum_{i=1}^{D} \frac{(v_i - b_i)^2}{2\sigma_i^2} - \sum_{i=1}^{D}\sum_{j=1}^{F} W_{ij} h_j \frac{v_i}{\sigma_i} - \sum_{j=1}^{F} a_j h_j \quad (5)$$

$$Z(\theta) = \int_V \sum_h \exp(-E(v,h;\theta)) \quad (6)$$

The pre-training step targets the maximization of the log-likelihood of the data, log P(v;θ). Differentiating this log-likelihood with respect to the parameters θ results in a form that consists of the expectation over the data distribution minus the expectation over the model distribution. The exact computation of the first part, which is the expectation over the model, is intractable. A contrastive divergence-based approach such as a one-step Gibbs sampling can be used to approximate the computation of the gradient of the log-likelihood probability. To build a DBN we stack several Bernoulli type RBMs on top of a one layer of type Gaussian-Bernoulli RBM. To learn the whole structure in a layer-by-layer manner the hidden activities of one RBM can be considered as the input data to a higher level RBM.
After the pre-training step of the DBN, a SoftMax layer is added on top of it and is trained using the classical back propagation approach. Given the model parameters θ fine-tuned over a pre-defined label set V = {l_1; l_2;…; l_V}, the DBN posterior gram for a feature vector frame x_i can be computed as:

$$DBN_{P_{x_i}} = [p(l_1|x_i;\theta), p(l_2|x_i;\theta), \ldots p(l_V|x_i;\theta)] \quad (7)$$

Where $\sum_j P(l_j|x_i;\theta) = 1$
Figure 8 illustrates the architecture of the used DNN-HMMs model for our Arabic OCR system.

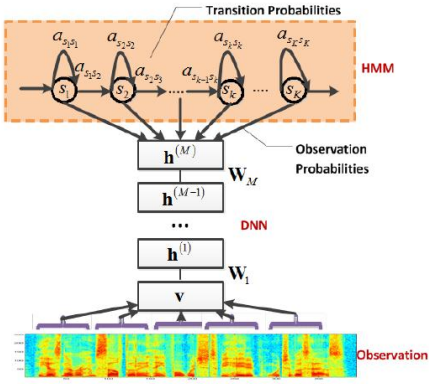
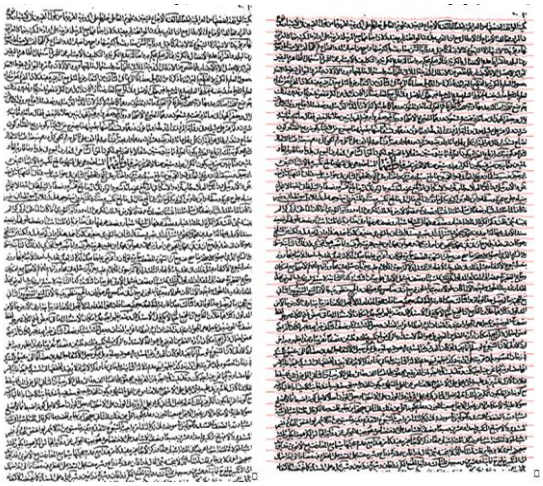


Fig. 8: *HMM-DNN Model*



Before Segmentation **After Segmentation**
Fig. 9: Arabic Calligraphy (Before and After Segmentation)

The model uses the HMM structure to represent the state transitions and DNN to represent the state emissions. Two types of models where investigated in our system, the context independent ligatures based DNN-HMM and a context-dependent tri-ligature DNN-HMM model. For context, independent based DNN-HMM we used DNN with output layer consisting from 963 nodes, that represents the 3 HMM states for each one of the used 321 ligatures. For the context, dependent based DNN-HMM we used DNN with output layer consisting from 3000 nodes that represents the clusters of the tied tri-ligature states. We used a clustering decision tree based on linguistic questions to cluster the Tri-ligature HMM model states. For the hidden layers, we experimented several DNN structures that ranged from 3 to 5 layers where each layer consists from large number of nodes in the range 2000 to 4000 nodes.

The impeded Viterbi algorithm is used to train the DNN-HMM model. A ligature level decoder was integrated in the training process. The objective function is to find the determinant of the ligature sequence P_h that maximize:

$$\widehat{Ph} = \underset{Ph}{\operatorname{argmax}}\, p(Ph|x) = \underset{Ph}{\operatorname{argmax}}\, p(x|Ph)p(Ph)/p(x) \quad (8)$$

Where $p(P_h)$ is the ligature level Language Model (LM) and $p(x|Ph)$ is the OCR model probability estimated by:

$$p(x|Ph) = \sum_q p(x,q|Ph)p(q|Ph) \quad \cong \max \pi(q_0) \prod_{t=1}^{T} a_{q_{t-1}q_t} \prod_{t=0}^{T} p(x_t|q_t) \quad (9)$$

Where $a_{q_{t-1}q_t}$ is the HMM sate transition probability and $p(x_t|q_t)$ is the observation probability that can estimated by:

$$p(x_t|q_t) = p(q_t|x_t)/p(x_t)/p(q_t) \quad (10)$$

Where $p(q_t|x_t)$ is the state posterior probability that is estimated from the DNN, $p(q_t)$ is the prior probability of each state estimated from the training data, and $p(x_t)$ is independent from the ligature sequence and can be ignored.
The main steps of the training procedure for the DNN-HMM model are summarized as following:
1- Train the best HMM model (either CI or CD) using the full training dataset.
2- Convert that HMM model to DNN-HMM model by borrowing the state transitions information (and state tying for the CD model).
3- Pre-train each layer in the DNN bottom-up layer by layer using the unlabelled training data
4- Use the trained HMM to generate the required labels for the training data frames up to the state level.
5- Use these acquired labels to train the last layer of the DNN using the back-propagation approach and the impeded Viterbi approach.
6- Use the trained DNN-HMM to generate new labels for the training data.
7- Repeat steps 5-6 for several iterations until the model converge.

4. Experimental Testing and Results

We compared the performance of first year of the Arabic OCR system (Al-Barhamtoshy et al, 2014, 2016, 2018; Nashwan 2018: Version 1), the 2-D HMM based system as reported in, and the second-year system (Version 2), which is a hybrid HMM-DNN with histogram features and use a complete line recognition with 3-gram language model as descried in this paper.
We used two test sets; the first one is 50 pages selected randomly from modern Arabic documents. The second test set was 50 pages selected from old and historical Arabic documents.

4.1 Line, Curved/Waved Segmentation

The segmentation approach that we developed in this manuscript is an integration between the "connected components" based approach and the "Seam Carving" based approach. This integrated approach takes advantage of the "seam carving" to segment noisy documents and makes better linking of the dots and the other Arabic diacritics. Table 1 shows the segmentation results for sample pages of Arabic old documents (Typewritten, Historical, and Calligraphy) of our dataset. In addition, table 2 shows the impact of this developed segmentation algorithm on the OCR result.

Table 1: Average line precision

Algorithm type	Line precision
Seam Carving	89.11%
Connected Component	93.08%
Proposed Algorithm (Integrated Approach)	95.46%

Table 2: Average recognition results of Arabic OCR engine

Algorithm type	Recognition accuracy
Seam carving	65.40%
Proposed Algorithm	67.22%

All the 40 calligraphy and historical documents include frames; the solution algorithm shows the result of frame extraction with average time three minutes. In case of the 100 of the early printed, there exist only 40 images with frames. The solution uses template matching and connected component algorithm with certain threshold value. The accuracy of frame extraction is 100% for early print, and it was 97.5% for historical images, (Table 3).

Table 3: The Frame Extraction on both Copies.

	Early printed		Calligraphy & Historical
	Without frames	*With frames*	
No. of pages	60	40	40
Correct pages	-	40	39
Run time (Sec)/page	0.45	4.89	5.75

The proposed line segmentation algorithm takes its input as early printed and historical images and produces segmented text waved line (curve-linear) images, as illustrated in Figure 9. The proposed algorithm is tested on 40 different images with 1480 lines (36-40 text lines for each image), which are taken from 40 different historical calligraphy books.
Table 4 summarizes the detail results of lines segmentation with the historical obtained results. Note that, when applying the proposed hybrid algorithms (HMM and DNN), not only the features are reduced, but also, the recognition accuracy is improved, and, the performance is improved. When we are comparing the DNN hidden layers, we find that, the three hidden layers achieves improved accuracy as illustrated in Table 5.

Table 4: *The Segmentation accuracy of the historical books*.

Images	No. of segmented lines	Accuracy
1	37	99.94
2	36	99.95
3	38	100.00
..	..	..
39	39	99.93
40	36	99.92
Total	1480	99.95

Table 5: *Average recognition results relative to number of hidden layers*

Number of Layers	Recognition accuracy
1 Hidden Layer	67.12%
3 Hidden Layers	68.22%

4.2 Modern Documents Results

Table 6 illustrates that the new version of the system provides absolute 4% gain over the previous version. The gain is contributed to the new histogram features, the enhanced HMM-DNN modeling and the complete line recognition to avoid word level segmentation errors. This accuracy improvement was on the price of the system speed. The complete line recognition has increased the search space greatly compared to the single word recognition in the old version. This change in recognition approach has slowed the recognition rate form 1

second / page in the old version to 20 seconds / page in the new version. With recent availability of large cloud computing facilities, the processing time for OCR systems is considered a secondary issue, as you always can add hardware to compensate for the page recognition time delay, in comparison to the system accuracy, which is the more critical issue. Especially for Arabic OCR systems that have been lagging for a long time the performance of equivalent systems of Latin-based languages. Therefore, any gain in the accuracy for an Arabic OCR system is for sure the most critical issue.

Table 7 displays samples of recognition results for modern Arabic documents using the old and new Arabic OCR systems. The reference text is displayed to the right and the recognition results are displayed to the left. The highlighted lines in yellow color are the ones that include errors. The error words are highlighted in grey. The white lines are the ones that are recognized correctly.

Table 7: Systems evaluation using modern Arabic documents

Image	Arabic OCR System Accuracy	
	OCR V.1	*OCR V.2*
Original	86.43%	94.97%
		

Table 6: Systems evaluation using modern Arabic documents

Arabic OCR Accuracy		
OCR V.1	***OCR V.2***	***Gain***
84.94%	90.99%	4.05%

4.3 Calligraphy/ Historical Documents Results

The results in table 8 shows that the new version of the system provides absolute 62% absolute gain over the previous version. This was a huge gain, and it provides a practical level of performance. As seen in these results, the previous version of the system failed to achieve any practical performance for the old documents. The same degraded performance was achieved also for the most common commercial Arabic OCR systems such as the two version (V.1 and V.2).

Table 8: *Systems evaluation using modern OCR (V.2) and (V.1) For Calligraphy Documents*

Arabic OCR Accuracy		
OCR V.1	***OCR V.2***	***Gain***
10.2%	71.87%	61.67%

Table 9: *Comparisons of WER between the two OCR systems*

Arabic OCR WER		
OCR V.1	***OCR V.2***	***Gain***
46.7 %	34.4 %	12.4 %

Table 9 illustrates Word Error Rate (WER) of the two versions of our OCR systems: (a) the OCR (V.1) that uses HMM without language model, (b) the modern OCR (V.2) which use DNN+HMM and language model. Therefore, the new OCR (V.2) demonstrates using tri language model and DNN had lower WER than the OCR (V.1). In addition, OCR (V.2) achieves best performance by 4.5 %.

The large improvement of the new system can be contributed to three main reasons (Arabic OCR V.2):

- Using a complete line segmentation, eliminated most of segmentation errors, especially with the words overlap, which is very frequent in old Arabic documents.
- The used histogram-based features which proved to be robust against the low printing quality, the dominant noise effect and the wide variance in ligature shapes that represent the main features of old Arabic documents.
- The discrimination power of the HMM-DNN models, which provided better modeling for the large number of Arabic ligatures that are common phenomena in Arabic old documents.

Table 10 displays the recognition results for the old and new versions of the system for some old Arabic documents. Table 11 displays average recognition accuracy as a result between "the proposed solution: Arabic OCR system", Sakhr OCR and ABBY OCR using noisy images.

Table 10: Average recognition results of Arabic OCR engines

OCR System	Recognition Accuracy
Proposed	67.50 %
Sakhr	36.00 %
ABBY	57.00 %

Table 11: Systems evaluation using Arabic Calligraphy documents

Image	Arabic OCR Accuracy	
	OCR V.1	*Arabic OCR V.2*
Original	74.00 %	90.07 %
		

5. Conclusion and Future Work

Despite the many efforts that have been devoted in the last two decades towards achieving Arabic OCR technologies with practical level of performance, the current state of art of Arabic OCR system is far lagging the equivalent technologies for Latin languages. In this paper we targeted the research and development of Arabic OCR that can provide practical performance for digitizing Arabic documents. Also, we addressed the challenges of old and calligraphy Arabic documents to improve the OCR accuracy of such type of documents above the level to make them searchable documents.

Therefore, we worked on enhancing the performance of all the components of Arabic OCR system starting with document preprocessing and denoising, text detection, line segmentation, features extraction and the recognition engine. We introduced the integrated approach of connected component and projection profiles to segment and recognize the entire documents.

Hybrid combination between dynamic timing power of HMM and multi-layer perception with large number of hidden layers (deep neural network (DNN)) are used to implement the Arabic OCR system. We evaluated the accuracy of the developed system against the best performing Arabic OCR systems that are commercially available and our system showed significantly improve in OCR accuracy, around 10% absolute raise in the recognition accuracy. Therefore, a curved/waved line segmentation algorithm is presented and implemented with connected components and ligature segmentation. The proposed segmentation algorithm has accuracy 99.99%, which is better than previous contributions. The experimental results of our system gave 14.5% character error rate compared to the best OCR of other systems.

In future work, extended work may include other approaches and algorithms to segment and recognize other categories of Arabic documents such as handwritten and historical. In addition, the language model and the statistical lexicon/corpus will be extended with probabilistic and statistics rules. The lexicon/corpus of the language model will include different types, categories and domains of Arabic documents and scripts. Therefore, this extension may improve recognition results significantly. Also, we plan to implement our Arabic OCR prototype system as a final product. This are, in addition to, Arabic OCR web services will be produced.

Acknowledgement

"This project was funded by the National Plan for Science, Technology and Innovation (MAARIFAH) – King Abdulaziz City for Science and Technology -the Kingdom of Saudi Arabia– award number (11-INF-1997-03). The authors also, thanks Science and Technology Unit, King Abdulaziz University for technical support".

References

[1] Abdel-Azeem S., and Ahmed H., (2013). "Effective technique for the recognition of offline Arabic handwritten words using hidden Markov models", International Journal on Document Analysis and Recognition (IJDAR), December 2013, 16 (4), 399–412.

[2] Attia M., El-Mahallawy M. S., Rashwan M., Nazih W., Al-Badrashiny M., (2015). Omni font text recognition of printed cursive scripts via HMMs, compact lossless features, and soft data clustering, Pattern Analysis and Applications, August 2015, 18(3), 507–521.

[3] Rashwan A., Rashwan M., Abdel-Hameed A., Abdou S., Khalil A., (2012). A robust omni-font open-vocabulary Arabic OCR system using pseudo-2D-HMM, Proc. SPIE 8297, Document Recognition and Retrieval XIX, 829707 (January 23, 2012); doi:10.1117/12.910390.

[4] Srimany A., Chowdhuri S., Bhattacharya U., Parui S. K., (2014). Holistic Recognition of Online Handwritten Words Based on an Ensemble of SVM Classifiers, Document Analysis Systems (DAS), 2014 11th IAPR International Workshop, IEEEXplore, 86-90.

[5] Al-Barhamtoshy H. (2016). "Towards Large Scale Image Similarity Discovery Model". 2nd International Conference on Advanced Technologies for Signal& Image Processing ATSIP'2016, March 21-24, Monastir Tunisia, IEEE Xplore, 1-9.

[6] Liu H., Zha H., Liu X. (2008). Skew detection for complex document images using robust borderlines in both text and non-text regions. Pattern Recognition Letters, 29(13), 1893-1900.

[7] Al-Barhamtoshy H., & Rashwan M. (2014). Arabic OCR Segmented-based System, Life Science Journal, 11 (10), 1273-1283.

[8] Abdelaziz I., Abdou S., & Al-Barhamtoshy H., A large vocabulary system for Arabic online handwriting recognition. Pattern Analysis & Applications, Springer, Dec. 2015, 19(4), 1129–1141.

[9] Elad M., & Aharon M. (2006). Image denoising via sparse and redundant representations over learned dictionaries. IEEE Transactions on Image processing, 15(12), 3736-3745.

[10] Kirti Y., & Patil M. (2013). Confidance Calibration Mesures to Improve Speech Recognition. International conference on Communication and Signal Processing, April 3-5, 2013, India, 826- 829.

[11] Li Bo, & Sim K. C. (2013). Noise adaptive front-end normalization based on Vector Taylor Series for Deep Neural Networks in robust speech recognition, ICASSP 2013, 7408- 7412.

[12] Al-Barhamtoshy H., Abdou S., & Rashwan M. (2014). Mobile Technology for Illiterate Education. Life Science Journal, 11(9), 242-248.

[13] Siniscalchi S., Yu D., Deng Li, and Lee C., (2013). Speech Recognition Using Long-Span Temporal Patterns in a Deep Network Model, IEEE Signal Processing Letters, March 2013, 20(3), 201-204.

[14] Stahlberg F., Vogel S., (2016). QATIP- An Arabic Character Recognition System for Arabic Heritage Collections in Libraries, 12th IAPR Workshop on Document Analysis Systems, 2016.

[15] Althobaiti H. and Lu C., (2017). A Survey on Arabic Optical Character Recognition and an Isolated Handwritten Arabic Character Recognition Algorithm using Encoded Freeman Chain Code, 51st Annual Conference on Information Sciences and Systems (CISS), IEEE Conference 2017.

[16] Ahmad I., Wang X., Li R., Ahmad M., and Ullah R. (2017). Line and Ligature Segmentation of Urdu Nastaleeq Text, IEEE Open Access Journal, Vol (5), pp. 10924-10940, 2017.

[17] Lehal G. S., (2013). ''Ligature segmentation for Urdu OCR,'' in Proc. 12th Int. Conf. Document Analysis Recognition. (ICDAR), Aug. 2013, pp. 1130–1134.

[18] Din I. U., Malik Z., Siddiqi I., and Khalid S., (2016). ''Line and ligature segmentation in printed Urdu document images,'' J. Appl. Environ. Biol. Sci, 6(3), pp. 114–120, 2016.

[19] Hesham A., Rashwan M., Al-Barhamtoshy H., Abdou S., Badr A., Farag I., (2017). "Arabic document layout analysis", Pattern Anal Applic, Springer.

[20] Hesham A., Rashwan M., Al-Barhamtoshy H., Abdou S., Badr A., (2017). "Posteriori word-based approach for Arabic Documents Font recognition", ICDAR 2017, (Under reviewing).

[21] Al-Barhamtoshy H., Abdou S., Rashwan M., Jambi K., An OCR System for Arabic Calligraphy Documents, International Conference on Communication, Management and Information Technology ICCMIT'17, 3-5 April 2017, University of Warsaw, Warsaw, Poland, http://www.iccmit.net/

[22] Nashwan F., Rashwan M., Abdou S., Al-Barhamtoshy H., and Moussa A., (2018). A Holistic Technique for an Arabic OCR System, J. Imaging 2018, 4(1), 6; https://doi.org/10.3390/jimaging4010006

[23] Al-Barhamtoshy H., Abdou S., (2018). Arabic OCR Metrics-based Evaluation Model, Journal of Engineering Technology, Vol 6(1), Jan., pp. 479-495.