Arabic OCR Segmented-based System

Hassanin M. Al-Barhamtoshy [1] and Mohsen A. Rashwan [2]

[1]Computing and Information Technology, King Abdulaziz University (KAU), Saudi Arabia
[2]Electronics and Communication Department, Cairo University
hassanin@kau.edu.sa, mrashwan@rdi-eg.com

Abstract: A new investigation in the Arabic OCR system has presented for the offline recognition of machine-printed cursive words. Therefore, a reliable transformation mechanism will be used to transform image text into free text (ASCII or Unicode Texts), that can be directly searched by a computer. Therefore, traditional preprocessing model (segmentation phase) will be included to extract each word from image text and divide it into segments. Then, recognition phase will take place, to find the most likelihoods of each possible text/character class given the segments. Accordingly, many classifiers can be used such as neural networks, Naïve Bayes, HMM classifiers. Such likelihoods are used to feed special algorithm as input in such ways to recognize the entire word. The whole process of the proposed framework includes three main stages: preparation, training, and testing. The data preparation aims at scanning, data image selection, alignment, identify text regions, and separate non text or image regions. Second, the training stage takes place, to extract features and build up the related language model; such features will be used in the third stage. Accordingly, at the first stage the paper focuses on the techniques used for font sizing, binarization, skewing, clearing (denoising), and segmentation before recognition takes place.
[Hassanin M. Al-Barhamtoshy and Mohsen A. Rashwan. **Arabic OCR Segmented-based System.** *Life Sci J* 2014;11(10):1273-1283]. (ISSN:1097-8135). http://www.lifesciencesite.com. 200

Keywords: Arabic; OCR; Segmented-based; System

1. Introduction

Method for Arabic feature selection was initiated in a handwritten OCR purpose [1] based on well-known common features extracted from the training patterns. Therefore, an algorithm was implemented in an OCR system for recognizing one of the biggest standard handwritten Farsi/ Arabic digit datasets.

A developed tool with proposed algorithm has been applied to find Table of Contents (ToC) pages in Urdu books without the use of OCR [2]. The proposed algorithm employed machine learning algorithm for segmenting the document image into digits and non-digits. So, vertical projection analysis is engaged to detect the column structure of a typical page.

Recurrent neural network (RNN) has been used for recognizing patterns of cursive handwritten documents [3]. The proposed solution had error rate of 13.6 % in case of shape variations and 5.15% in case of character level. Another technique has been engaged to fragment printed Arabic texts in order to split the Arabic characters and then extracting features for each to be recognized [4].

Another approach is anticipated and attempted to identify and separate handwritten from printed text using the Bag of Visual Words model (BoVW). Firstly, blocks of interest are detected in the document image, and then a descriptor is calculated based on the BoVW [5]. The last classification of the blocks can be characterized as Handwritten, Machine Printed, or Noise.

2. Characteristics of Arabic Characters

Arabic language is one of the most spoken languages in the world, 422 people around the world speak it, which considered being one of most considered languages around the globe [6]. Arabic speakers are increasing, therefore a number of Arabic documents and articles are increased. Arabic is also the language of the Qur'an, so Muslims of all nationalities, such as Indonesians, are familiar with it. This shows the importance of the Arabic Language in the world [6], [7], we can summarize several important differences:

- Arabic Alphabet consists of 28 consonants and 8 vowels/diphthongs. Short vowels are unimportant in Arabic, and indeed do not appear in writing.
- Arabic texts are read and written from right to left.
- Arabic texts are written in a cursive script, in which most characters are connected and their shapes vary according the position in the word.
- The presence of dots in Arabic letters (15 out of the 28). This leads to some characters being more prone to OCR errors than others are.
- The morphological and syntactic complexity of Arabic grammar, which results 60 in billion possible surface forms complicates error correction within dictionary-based solution.
- It is semi-cursive whether printed or handwritten. Each character has a connection point right and/or left linked on the baseline.

- The concept of uppercase and lowercase does not exist in Arabic script.

The Arabic word or Piece of Arabic Word (PAW) can be made up of one or more components (pseudo-word) and the characters of the same connected component can be ligatured horizontally and vertically dependent on the used font (Table 1) [8].

Table 1. The Piece of Arabic Word (PAW) or Arabic letters according to their position

Arabic letters.

Label	Isolated	Begin	Middle	End
Alif	ا		ـا	
Baa	ب	بـ	ـبـ	ـب
Taaa	ت	تـ	ـتـ	ـت
Thaa	ث	ثـ	ـثـ	ـث
Jiim	ج	جـ	ـجـ	ـج
Haaa	ح	حـ	ـحـ	ـح
Xaa	خ	خـ	ـخـ	ـخ
Daal	د		ـد	
Thaal	ذ		ـذ	
Raa	ر		ـر	
Zaay	ز		ـز	
Siin	س	سـ	ـسـ	ـس
Shiin	ش	شـ	ـشـ	ـش
Saad	ص	صـ	ـصـ	ـص
Daad	ض	ضـ	ـضـ	ـض
Thaaa	ط	طـ	ـطـ	ـط
Taa	ظ	ظـ	ـظـ	ـظ
Ayn	ع	عـ	ـعـ	ـع
Ghayn	غ	غـ	ـغـ	ـغ
Faa	ف	فـ	ـفـ	ـف
Gaaf	ق	قـ	ـقـ	ـق
Kaaf	ك	كـ	ـكـ	ـك
Laam	ل	لـ	ـلـ	ـل
Miim	م	مـ	ـمـ	ـم
Nuun	ن	نـ	ـنـ	ـن
Haa	ه	هـ	ـهـ	ـه
Waaw	و		ـو	
Yaa	ي	يـ	ـيـ	ـي

Table 2. The most used Fonts in modern Arabic

Simplified Arabic	العربيـــة النصـــوص علـى الآلـي التعـرف المطبوعـة
Traditional Arabic	العربيـــة النصـــوص علـى الآلـي التعـرف المطبوعـة
Microsoft San Serif	المطبوعة العربيـــة النصـــوص علـى الآلـي التعـرف
Andalus	العربيـــة النصـــوص علـى الآلـي التعـرف عةالمطبـو
Arabic Typesetting	العربيـــة النصـــوص علـى الآلـي التعـرف المطبوعـة
Arial Unicode MS	العربيـــة النصـــوص علـى الآلـي التعـرف المطبوعـة
Courier New	العربيـــة النصـــوص علـى الآلـي التعـرف المطبوعـة
Sakkal Majalla	العربيـــة النصـــوص علـى الآلـي التعـرف المطبوعـة
Segoe WP	العربيـــة النصـــوص علـى الآلـي التعـرف المطبوعـة
Tahoma	العربيـــة النصـــوص علـى الآلـي التعـرف المطبوعـة
Simplified Arabic Fixed	العربيـــة النصـــوص علـى الآلـي التعـرف المطبوعـة
Times New Roman	العربيـــة النصـــوص علـى الآلـي التعـرف المطبوعـة
Urdu Typesetting	العربيـــة النصـــوص علـى الآلـي التعـرف المطبوعـة

There are more than 450 Arabic fonts [8, 9] used somewhere in Arabic books and Arabic old documents. Figure 2 illustrates thirteen fonts of the most used in modern Arabic, today. Some fonts are simpler without overlaps and ligatures (Tahoma). Some other fonts are more difficult, richer in overlaps, ligatures and flourishes (Urdu Letter) [9].

Arabic graphemes connectivity, dotting, multiple graphemes for the same character at different positions, and composite ligatures are some of the extra challenging properties of Arabic script [10].

3. Proposed Framework Approach

A proposed framework system with a font and size recognition is described as a hybrid system. As presented in figure 2, it works using three stages: (1) data preparation and annotation module, (2) training module, and (3) recognition module. All of them share the same architecture and the same feature extraction.

Figure 1 shows some difficulties that are still behind due to a multitude of complexities.

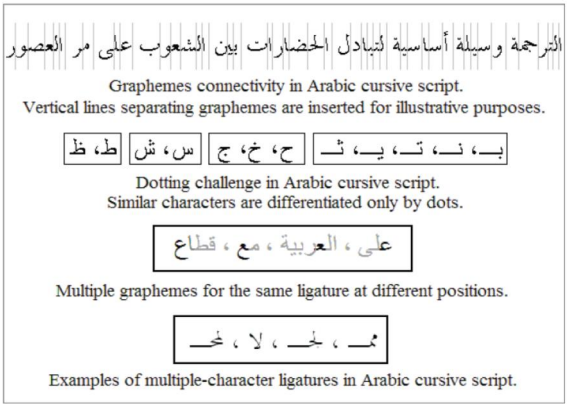

Fig. 1. Challenges of Arabic Cursive Script

The proposed framework is based text polarity to differentiate between binarized pixel of text and binarized pixel of image [11]. Thus, in our approach, overlay text binarization is used to obtain better result in segmentation. Consequently, k-means clustering with Otsu's global threshold and graph-cut method are used to get binarization result [11].

At the training and recognition phases a GUI tool uses sliding window for feature extraction. The obtained feature vectors are employed to train the font and size and generate language model using the expectation maximization likelihood algorithm. The recognition is performed through a simple score comparison of the trained language model.

Multi-layer neural networks are trained with back-propagation architecture algorithm. The proposed architecture are designed to recognize visual patterns from pixel images. Consequently, feature map is produced using convolution of pixel image with filter and function. Therefore, if we denote the k^{th} feature

map at specific layer as f_k, with weights w_k and bias b_k then the feature map f_k is defined as the following:

$f(k)_{ij} = (w_k * x)_{ij} + b_k$

Accordingly, hidden layers are represented as a set of multiple feature maps { f (k), k= 0… K }.

The Arabic OCR system (figure 2) shows a DFD of the proposed OCR system. In the training phase, a segmentation DNN model is trained from a collection of segments data set. Entirely there is an enhancement phase in order to generate enhanced dataset. Secondly, the recognition phase is used to recognize the estimated clean characters and words. A detailed description of the feature extraction module can be found in [10-12]. Also, in the training phase, word generation feature extraction and clustering process will be employed. The default concept is based on the property of the discrete cosine transform (DCT). A 2D-DCT transform coefficient has been selected as a word feature to generate a unique vector and minimize errors on the classification process.

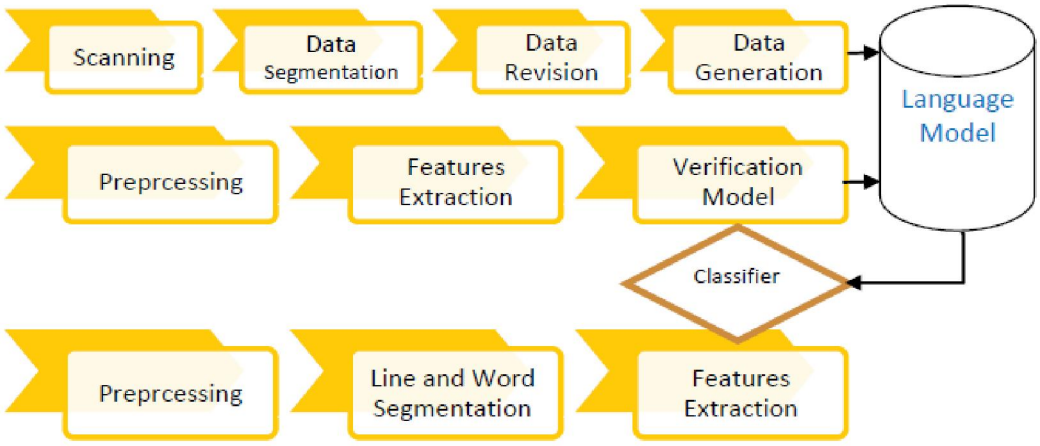

Fig. 2. DFD of the Proposed OCR System

3.1 Data Preparation

Data preparation is important during document understanding, document analysis and the OCR processes. In these processes, the data may contain unusable image formats, missing values, errors, and compressed format. Therefore, additional tool may be used at this level.

3.2 Data Set Generation

The used database in this work as references at training phase is prepared from a large dataset of more than 75 Arabic books, which are generated by computer in three fonts: Simplified Arabic, Traditional Arabic, and Arabic Transparent (Lotus Linotype in case of book test), in 300 dpi. Overall, our dataset has a good coverage for the Arabic Language. The corpus used to generate the 341 models is the news domain.

3.3 Segmentation and Annotation

The following steps are the required outputs associated with each image file:

1- Each image file forms the proposed dataset combined with a text transcription file for each image file. Text transcription files are required to be xml files that include all the details of each image (file name, font, size, quality, etc.) plus the full text transcription of the corresponding image for each line.

2- Each line in the image with full information about the starting/ending and height of each line in the image so that it can extract the line directly from the image such that there is no overlap between adjacent lines. This suggests approach either by using rectangular boxes or by using flexible contours. GUI tool is used to extract the line by specifying the page and the line number in the page.

3- The preprocessing module segments each line, by specifying word boundary segmentation for all the lines, such that there is no overlap between adjacent words. The segment will be done either by using rectangular boxes or by using flexible contours (polygon contour). The GUI's tool is used to extract the word by specifying the page, the line number in the page, and the word number in the line.

4- It is required to have character boundary segmentation for a 10% portion of the produced data. These boundary segmentations should extract the character (character or ligature) directly from the line/image/word such that there is no overlap between extracted characters. This can be done by using a flexible contour. The GUI proposed tool is also used to extract the character by specifying the page, the line number in the page, the word number in the line, and the character number in the word.

5- The transcription file should be hierarchical, in the sense that it specifies the page, and the number of lines in the page. Then, for each line, it specifies the number of words in the line. For each word, it

specifies the number of graphemes in the word. For each grapheme, it shows its number of characters, and then shows each character.

6- The above information should be included in the transcription files and it should allow to extract any line/word/character directly from the image by specifying the page, the line number, the word number, and the character number.

A. Font Resizing

One of the important step of Arabic OCR is concerning with font sizing/resizing. Usually, font sizing techniques involve the font sizing as well as font styling [13]. In some cases when the font size is less than 12 the binarized image looks very bad, and hard to be recognized, so a simple resizing algorithm was used. The proposed layout algorithm is as follows:

Resizing algorithm

1. Determine the font size of the given image, which is done by extracting some features from the image words i.e. average words width, average height, aspect ratio and compose feature vector.
2. Obtain the nearest font size (10, 12, 14, 16, 18, 20, 22).
3. If the detected font size is less than 12, resize the image with a factor equals to 1.5.

An overview of the key pages sizing and page frame content area are given in figure 3.

Fig. 3. Examples of page frame along with different font sizes (before and after)

Accordingly, a font of a particular design varies in feature-based parameters, such as; size, height, weight, width, slope, curvature, thickness, boldness, etc.

The font used in text may vary in design and shape from location to domain location. So, general conceptual form of font, font generation methods, and font and style strategies are discussed in [13].

As illustrated in [14] five generic font families are used in cascading style sheets (CSS) to adding styles to web documents. Such CSS includes fonts, colors, and spacing. All the font families consist of serif, sans serif, cursive, decorative fantasy, and fixed-width monotype fonts.

Accordingly, a number of alternative fonts can be reduced to single font, this cause to improve OCR accuracy.

(B) Text Orientation and Text Polarity

Text orientation deals with the direction of Arabic printed text, including; (1) Detection of printed text (portrait or landscape); (2) One column or two columns page, etc. An overview of the key pages orientation is illustrated in figure 4.

Detection of portrait or landscape is derived by using projection histograms or projection profiles, taken into consideration counting black to white transitions.

The projection profile algorithm is based on horizontal and vertical histograms. Consequently, the proposed algorithm takes the following rules:

$P_h >= a\ P_v$; i.e.; Text orientation is landscape

$P_h < b\ P_v$; i.e.; Text orientation is portrait

Many researchers use gray histogram for polarity judgment [11]. Also, Gaussian filter will be processed within the histogram. Then, the proposed images are fed to the Arabic OCR proposed system. As mentioned in many literatures, that the background corresponds the higher peak in the histogram, but in practice this is not true. Accordingly, skeleton based method is proposed to obtain binary images, the one that has fewer pixels represent text polarity [11].

(C) Binarization

Image binarization is defined as "the conversion of a color image or grayscale into binary image" [14]. The majority of image processing and image analysis systems have been implemented to work with binarization [15]. A short description of binarization and related methods are described in the following subsections.

Several technical terms are also used such as text separation, threshold, and text segmentation. The binarization approach can be categorized into global, local, and hybrid methods. The global method uses single threshold value T for the entire document, so,

the resulting binary document B(x, y) is defined as the following:

$$B(x, y) = \begin{cases} 1, if\ D(x,y) \le T \\ 0, if\ D(x,y) > T \end{cases}$$

However, in local method, threshold value is employed for each pixel (local) in the image. In the third method; hybrid, the binarization uses both global and local information.

Fig. 4. An Arabic Page Orientation Detection (a) Portrait (b) Portrait 2 Columns (c) Landscape

1. Global Method

Classifying of the pixels can be pertained into two categories foreground text C_0 and background text C_1, according to the good threshold value; as the following equation.

$$\eta = \frac{\sigma B}{\sigma T}$$

where η is a separability measure, σ_B and σ_T represent the variances of between-class and total class respectively. The good global threshold is defined as:

$$t = \arg\max \eta;\ \text{where } t \in \{0..255\}$$

2. Local Method

Rectangular window is shifted across the image to calculate a pixel wise threshold. The threshold T of such window is calculated using the mean m and the variance v of the gray values in the window using:

$$T = m + c\ v;$$

where c is constant set to 0.2 [15].

An adaptive proposed method is used to compute the threshold as follows:

$$T = m + (1- c (1 – v / s))$$

where s represents standard deviation (fixed to 128); and c is positive constant value set to 0.5.

(D) Noise Cleaning (Filtering)

Many types of noise presented a challenging problem in image analysis. Several of these noises, mainly due to document ageing (historical document), human manipulations during scanning, environmental conditions as well as document usage. Consequently, it is important to have an efficient document enhancement process, in order to recover the good quality of the image. These noises can be classified into four broad classes:

1. Low Contrast.
2. Shadow effect.
3. Noisy background and damaged character
4. Borders with/without neighboring page.

***1.* Low Contrast**

Sometimes, text region is not clear due to low contrast between background and foreground text. So, reducing background noise and increasing readability of foreground text have been proved by using a low-pass Weiner filter [14].

Fig. 5. Image examples (a) Background Illumination (b) Good Background contrast (after processing)

***2.* Shadow Effect**

Shadow is the result of diffusion or transparency of printing ink from one side to other of a page, so, page interferes with the data contained on the two sides. Most of these defects are removed using threshold like methods [15] [16].

***3.* Noisy background and damaged character**

This noise occurs due to poor quality of printed characters, merged characters, characters with holes or noise may appear at the background, see figure 3. An overview techniques in document analysis have been illustrated in [14] with image enhancement methods.

***4.* Borders with/without neighboring page**

Noisy black border is a result during document image capturing. Also, noisy text regions may be included from neighboring pages. Accordingly, additional overview techniques in document have been illustrated to enhancement image, taken into account, perform document cleaning by filtering out connected components (border removal). This method uses projection profile combined with connected components [14] [15].

(E) Image Orientation and Skewing / De-skewing

Image orientation plays an important role for document feeding, and heavy automatic document processing, when scanning and facsimileing documents being scanned in the wrong orientation. Accordingly, this wrong orientation is to be corrected in the right direction.

Skewing and de-skewing methods are considered with minimum and maximum rotation angles that can be detected and considered. The basic idea of the de-skewing algorithm is as the following steps:

1. Search about reference points, then lines in the document.
2. Compute the angle of the lines.

3. Calculate the skewing angle by using the following equation:

Skew-angle = average of all the computed angles

4. Rotate the angle by the value of the "skew angle"

Also, image can be rotated by specifying three points; upper left, upper right, and lower left corners, of the original image[http://www.msd.microsoft.com]

4. Binarization Overview

The overview of the proposed binarization is shown in figure 6. It includes two main steps; first, the pixels in the given text/ paragraph region are grouped into cluster using K-means. Then, the polarity of the text will be determined into one of the following two actions:

(1) Light cluster on dark background; or

(2) Black cluster on light background.

Consequently, text cluster with polarity action are what we mean by binarization.

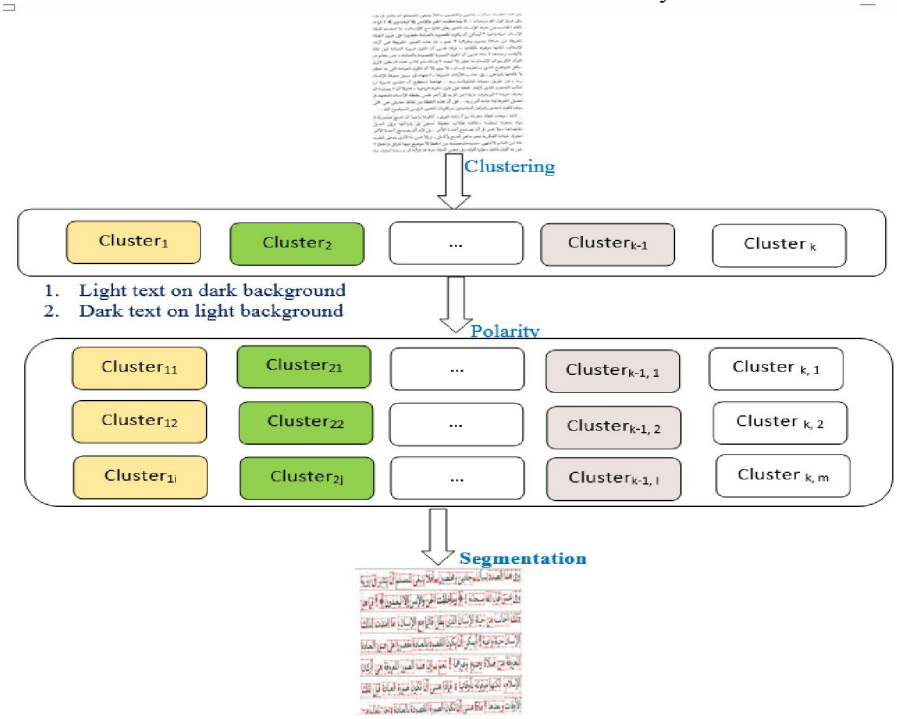

Fig. 6. The Overview of Binarization

(A). Clustering

K-means clustering will be used in this phase, taken into consideration the RGB space to group the pixels of the scanned image into K cluster [11]. Accordingly, for each region or segment, the following formula will be used [11] to calculate the score of each segment or region.

$$\text{Score (i)} = \alpha \frac{|If \cap Ci|}{|Ci|} + \beta (1 - N(v_i)) + \gamma (1 - N(B_i))$$

Where; i is the index of the classes, I_f represents the foreground pixels in image I. C_i is the set of pixels in cluster i. $N(v_i)$ represents the normalization computation of variance (v_i) and boundary pixels (b_i). Also, the three parameters; α (Alpha), β (Pita), and γ (Gamma) are used to weight the three elements in the score equation.

(B) K-Nearest Neighbor Classifier (KNN)

During the training phase, features are extracted from the training set by performing algorithm to build feature vectors database for the training set (356092 words) using three font's type, Simplified Arabic, Traditional Arabic, and Arabic Transparent with sizes 12, 14 and 16 respectively.

In order to classify a word image, the feature vector of word image is compared against each training feature vector by computed the Euclidian distance to measure the similarity between the two vectors. Then, prediction class of the testing image is found based on the minimum difference measured by the Euclidean distance; between the testing word image and the training samples.

In this work the classifier identifies the top k closest vectors (classes), which have the minimum Euclidean distance with the test image vector, and then arrange the distances in ascending order represent the candidate words that chosen as entries to the next stage. The Euclidean distance $d(w_c, w_t)$ for each candidate word w_c can be calculated as squared Euclidean distance or as absolute Euclidean distance.

(C) Dataset Management and Annotation

Annotation tasks range from simple data document labeling to text extent tagging and tag linking. Therefore, more information is needed to be contained within the annotation; category labeling. However, multiple categories use labels for an image, e.g. file name, font, size, quality, etc.), plus the full text transcription. Other labeling uses boundary segmentation for all the lines (starting/ending and height of each line in the image). Classification and identification tasks are two annotation labels that refer to the entirety of document images. Therefore, many annotation tasks required in this approach, where segmentations are applied to each word, rather than XM generation tool. There are other types of tasks such as ligature segmentation, and event verification tools.

The metadata-type tags used for the document image task could contain start and end indicators or could leave them out. However, with stand-off annotation it is required that locational indicators are present in each tag. Character clustering is a methodology to organize collections of dataset, and it is used with other fields such as information retrieval (IR) and topic identification. Also, it goals to partition a given documents into meaningful classes. The quality of clusters can be evaluated using entropy [12]. Therefore, the entropy of a cluster C_r using size n_r is calculated as:

$$E(Cr) = -\frac{1}{Log\ q}\sum_{i=1}^{q}\left(\frac{Nri}{Nr}Log\frac{Nri}{Nr}\right)$$

Where q is the number of clusters that are considered correct for evaluation, and $N_{r\neg i}$ is the number of documents for cluster i found in cluster r. The overall entropy is defined as follows:

$$Entropy = \sum_{r=1}^{k}\left(\frac{Nr}{N}E(Cr)\right)$$

The better clustering classification is the smaller values in the range of 0 to 1.

The manager consists of three manager modules named as Loader module, Feature Extraction module, and Classifier module, as shown in figure 7. The loader loads subsequent features from the database of the language model and placed in the warehouse. Feature Extraction module defines a set of properties, which are ranked based upon the role of the character features and determines such properties for each character/word. Classifier Module gets the result from Feature Extraction module and classifies it by using classifier (KNN, HMM, DCT, or Support Vector Machine (SVM)). At last, the proposed manager works with the characters and words validation and interpretation of collected data through an access to the classified documents, and decides the behavior or the attitude of the classification. Consequently, the practical applications of each of these methods use PAWs mechanism as a case study. PAW aims at marking up what probably think of as proper characters- words in the real world that have specific designators, not just generic labels. Therefore, the character can be assigned by numbers to the words. Simply number every token in order, starting at 1 and going until there are no more tokens left. Another way is to assign numbers to each sentence, and identify each token by sentence number and its place in that sentence.

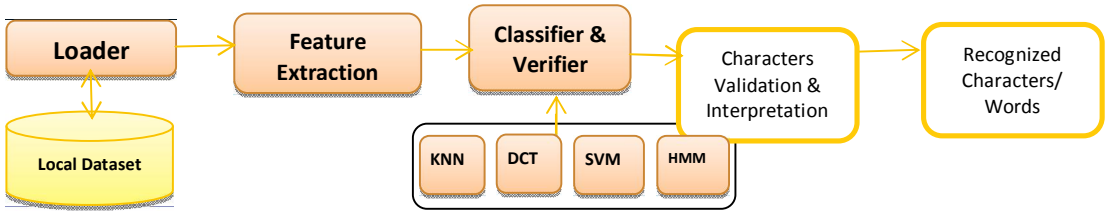

Fig. 7. Proposed Recognition Processing

(D) Verification Model

Verification model attempts to compute the probability P(W|F) of retrieving a word (W) given a feature (F) from the language model. Also, this model attempts to estimate similarity between a word on a page as follow [12]:

Sim(W|F) = P(R|w) / P(!R|w)

where R is the set of relevant words to a language model (features). The model tries to compute the probability that the word is relevant and the probability of the same word is not relevant [12].

5. Experimental Results

This section presents the experimental results of systems that have been described. It deals with the results of the proposed Arabic OCR System. These results are based on the assumption that the test data will be performed with perfect word segmentation. This type of data test is composed of 75 pages taken from books, historical book, thesis, and newspaper articles that have been printed in three fonts and four different sizes in two types of qualities clean and copy1. Figure 8 shows what it might look like to scan, segment, recognize and verification for the scanned image using the proposed tool.

Fig. 8. An Example of Scanning, Segmentation and Recognition Processes

At the verification phase, if there is difference between the number of lines in the segmented image and the recognized text, the verification tool focus on such differences, see figure 9.

Certainly, more identifying features could be added, such as paragraph number, document number, and so on. The advantage of having additional information used to identify words is the information that can be used later to help define features for the machine learning algorithms. Figure 10 illustrates screen shot of the Arabic batch of images after annotation model taking into consideration Arabic characteristics and peculiarities.

The two files (scanned image file and the recognized file "text") are not equivalent.
Scanned image's number of lines = 42
Recognized text's number of lines = 41

Fig. 9. Difference between the Segmentation and Recognition at the Verification Phase

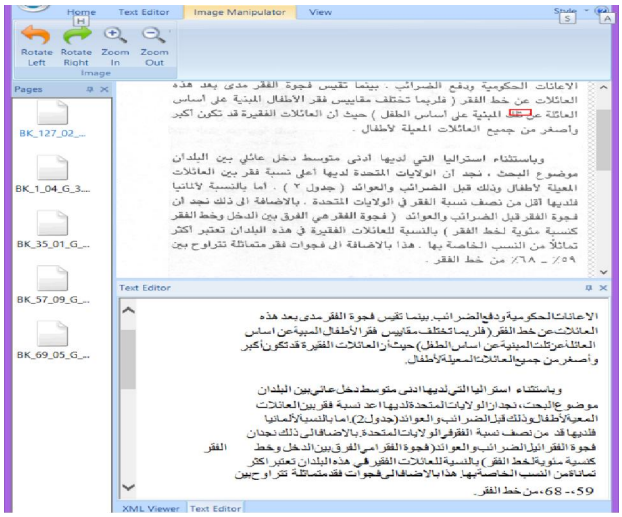

Fig. 10. Arabic batch Recognition of Arabic Images

5.1 Laser Scanned Text

In this part, the proposed system has been tested using (27624 single word images) in different size (12, 14, 16 and 20), fonts ("Arabic Transparent", "Simplified Arabic" and "Traditional Arabic"), which are the most common fonts, and different quality (clean and copy1). The main characteristic of which is that the system is font size independent, and the large vocabulary more than 356 thousand words, cover more than 96% of Arabic printed words used in the news domain.

Table 3 shows the simplified Arabic single-font experimental top-1 hypothesis results of this font and the corresponding Word Segmentation Rate (WSRs). This system achieves the WRR in Top-1 (as the first choice) 99.94 %. It is noticeable that beyond the top-1, the WSR is high and accurate, especially when top-n will be employed.

5.2 Overall Evaluation

Finally, let us have a method that classifies every document in the test directory and prints out the percent accuracy of this method. Accordingly, the proposed system decides the value of the accuracy; such value is taking into consideration: (1) Correct characters/words; and (2) Wrong characters/words. Table (4) illustrates the evaluation results for the proposed system relative to the competitive systems taken into consideration human experts' judgment for the five collection dataset types.

Figure 11 shows the comparison of the accuracy on the first datasets (books, journal and printed thesis) that is scanned and selected in the Arabic documents. As shown in table 4, the accuracy of the pre-processing was 91.54 (for proposed system), 92.00 (for proposed system with Nuance pre-processing support) and 81.3 (for Sakhr system), taken into consideration segmentation phase. The second row includes speed of the processing in second. The last two rows of the table include old or historical books (200 pages) with relative speed. Such figure shows the accuracy plot obtained by the experimental testing of the current test. It has been 10% better accuracy between the proposed system (with Nuance preprocessing) and Sakhr.

6. Conclusion

A new investigation in the Arabic OCR system has presented for the offline recognition of machine-printed cursive words. Various technical methods in pre-processing for data preparation of the Arabic OCR system have presented including the stages of resizing font, binarization, deskewing, denoising, and segmentation. Experimental results showed that proposed system achieves a good performance with less complexity. The proposed system has been implemented using C++ language and running in windows environment.

In future work, in order to improve the performance of the recognition engine further investigation will be conducted. This would include: larger vocabulary, system training using large number of fonts, and adding additional techniques like wavelet, Hough and Zernike.

Table 3. Simplified Arabic WSRs for Multi Size Fonts

Test type	Testing Fonts Size	Top 1
Clean	12	98.03%
	14	99.55%
	16	99.94%
	20	99.15%
	Ave.	99.40%
One Copy	12	98.55%
	14	98.49%
	16	98.43%
	20	98.72%
	Ave.	98.17%
Average	12	98.34%
	14	98.43%
	16	98.41%
	20	98.59%
	Ave.	98.32%

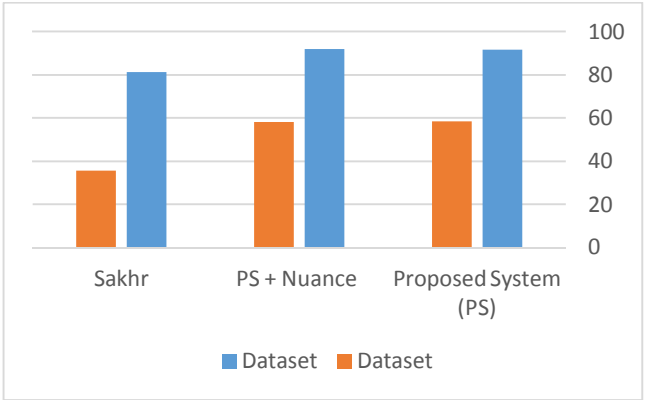

Fig. 11. Segmentation Accuracy Comparison

Table 4: Overall Evaluation Accuracy

Dataset Type	Pre Processing and Segmentation									
	No. of pages	Proposed System (PS)						PS + Nuance			Sakhr
Books(75), Journals (4), Thesis (45)	124	91.54%						92.00%			81.30%
Time (sec/page)		Fix font size	Deskewing	Binarization	Denoising	Segmentation	Total time	Segmentation	Preprocessing only	Total time	5.3
		12.15	6.47	1.35	0.01	16.26	39.00	17.97	0.96	20.50	
Historical Books	200	58.43%						58.00%072%			35.70%
Time (sec/page)		Fix font size	Deskewing	Binarization	Denoising	Segmentation	Total time	Segmentation	Preprocessing only	Total time	30.00
		42.06	29.81	2.67	0.03	57.64	137.86	51.00	2.6	55.15	

Acknowledgment

The teamwork of the ***"Arabic Printed OCR System"*** project was funded and supported; by the NSTIP strategic technologies program in the Kingdom of Saudi Arabia- project no. (11-INF-1997-03). In addition, the authors acknowledge with thanks Science and Technology Unit, King Abdulaziz University for technical support.

References

1. Mohammad Amin Shayegan and Chee Seng Chan, "A New Approach to Feature Selection in Handwritten Farsi/Arabic Character Recognition," 2012, International Conference on Advanced Computer Science Applications and Technologies, pp. 506-511.
2. Adnan Ul-Hasan, Syed Saqib Bukhari, Faisal Shafait, and Thomas M. Breuel, "OCR-Free Table of Contents Detection in Urdu Books," 2012, 10th IAPR International Workshop on Document Analysis Systems, pp. 404 – 408.
3. Adnan Ul-Hasan, Saad Bin Ahmed, Sheikh Faisal Rashid, Faisal Shafait and Thomas M. Breuel. Offline Printed Urdu Nastaleeq Script Recognition with Bidirectional LSTM Networks, 2013 12th International Conference on Document Analysis and Recognition, pp. 1061-1065.
4. Safwa Taha, Yusra Babiker, and Mohamed Abbas, Optical Character Recognition of Arabic Printed Text, Research and Development (SCOReD), 2012 IEEE Student Conference, pp. 235- 240.
5. Konstantinos Zagoris, Ioannis Pratikakis, Apostolos Antonacopoulos, Basilis Gatos, Nikos Papamarkos, Distinction between handwritten and machine-printed text based on the bag of visual words model, Pattern Recognition 47 (2014) 1051–1062.
6. Hassanin M. Al-Barhamtoshy and Fatimah M. Mujallid, Designing and Implementing Bi-Lingual Mobile Dictionary to be used in Machine Translation, International Journal of Digital Information and Wireless Communications (IJDIWC), March 2014.
7. Hassanin M. Al-Barhamtoshy, and Fatimah M. Mujallid, Building Mobile Dictionary System, The International Conference on Digital Information Processing, E-Business and Cloud Computing (DIPECC 2013), The society of Digital Information and Wireless Communication (SDIWC), October 23-25, 2013.
8. Fouad Slimane, Slim Kanoun, Jean Hennebert, Adel M. Alimi, Rolf Ingold, "A study on font-family and font-size recognition applied to Arabic word images at ultra-low resolution", Pattern Recognition Letters 34 (2013) 209–21.
9. Ben Amara, Najoua Essoukri, Gazzah, Sami, 2004. Une approche d'identification des fontes arabes. In: Actes du 8ème Colloque Internat. Francophone sur l'Ecrit et le Document 8, pp. 21–25.
10. Attia, M., Arabic Orthography vs. Arabic OCR, Multilingual Computing & Technology magazine, USA, Dec. 2004.
11. Zhike Zhang, and Weiqiang Wang, A Novel Approach for Binarization of Overlay Text, IEEE International Conference on Systems, Man, and Cybernetics, 2013, pp. 4259-4264.
12. Imed Zitouni (Editor), (2014). Natural Language Processing of Simitic Languages, Springer. Chapter 10: Kareem Darwish, Information Retrieval, pp. 299-334.
13. Umapula Pal and Niladri Sekhar Dash, (2014). Language, Script, and Font Generation, Handbook of Document Image Processing and Recognition, Springer Reference, Ch. 9, pp. 291-330, 2014.
14. http://www.ntchosting.com/multimedia/font.html.
15. Basilis G. Gatos, Imaging Techniques in Document Analysis Processes, David Doermann Karl Tombre Editors; Handbook of Document Image Processing and Recognition, Springer, Vol. 1, pp. 73-131.
16. Tseng YH, Lee HJ (2008). Document Image Binarization by Tow-Stage Block Extraction and Background Intensity Determination, Pattern Anal Appl 11,pp. 33-44.