# Printed Arabic Script Recognition: A Survey

Mansoor Alghamdi
Department of Computer Science
Community College
University of Tabuk
Tabuk, Saudi Arabia

William Teahan
School of Computer Science
Bangor University
United Kingdom

***Abstract*—Optical character recognition (OCR) is essential in various real-world applications, such as digitizing learning resources to assist visually impaired people and transforming printed resources into electronic media. However, the development of OCR for printed Arabic script is a challenging task. These challenges are due to the specific characteristics of Arabic script. Therefore, different methods have been proposed for developing Arabic OCR systems, and this paper aims to provide a comprehensive review of these methods. This paper also discusses relevant issues of printed Arabic OCR including the challenges of printed Arabic script and performance evaluation. It concludes with a discussion of the current status of printed Arabic OCR, analyzing the remaining problems in the field of printed Arabic OCR and providing several directions for future research.**

***Keywords*—Optical character recognition; arabic printed OCR; arabic text recognition; arabic OCR survey; feature extraction; segmentation; classification**

## I. Introduction

Optical Character Recognition (OCR) is a technique that transforms a printed or handwritten text image into an electronic format. OCR development is considered a challenging task in the field of pattern recognition. Many OCR approaches have been proposed for Latin and non-Latin scripts. However, printed Arabic OCR still poses great challenges because of the special characteristics of Arabic script [1].

Arabic OCR is highly desirable in various real-world applications, such as digitising learning resources to assist visually impaired people, bank cheque processing and mail sorting[2], [3]. Furthermore, there are many initiatives for Arabic digital content enrichment [4]. One of these initiatives is King Abdullah’s Initiative for Arabic Content. Therefore, a robust and efficient Arabic OCR is required to support this initiative by increasing Arabic content on the Internet.

Numerous methods have been proposed for recognising printed Arabic script from an image, yet we are unaware of comprehensive surveys of printed Arabic OCR during the last fifteen years. Two surveys have been conducted on printed Arabic OCR [2], [5]. However, these reviews do not reflect the current progress in printed Arabic OCR. Therefore, establishing a guide and baseline for future directions remains important for Arabic OCR researchers.

This work will establish this guide and baseline for Arabic OCR researchers by providing a comprehensive literature review of printed Arabic text recognition research. It reviews techniques that have been utilized for developing printed Arabic OCR with emphasis on the issues related to Arabic script. It also highlights the current status of printed Arabic OCR and provides several directions for future research.

This paper is organised as follows. In section 2, Arabic script characteristics and challenges are discussed. Section 3 presents the methodologies of printed Arabic OCR, with subsections that review the five stages of the development of printed Arabic OCR: preprocessing, segmentation, feature extraction, classification and post-processing. Section 4 discuses performance evaluation issues of printed Arabic OCR. Section 5 concludes with a discussion about open problems and future directions.

## II. Arabic Script Characteristics and Challenges

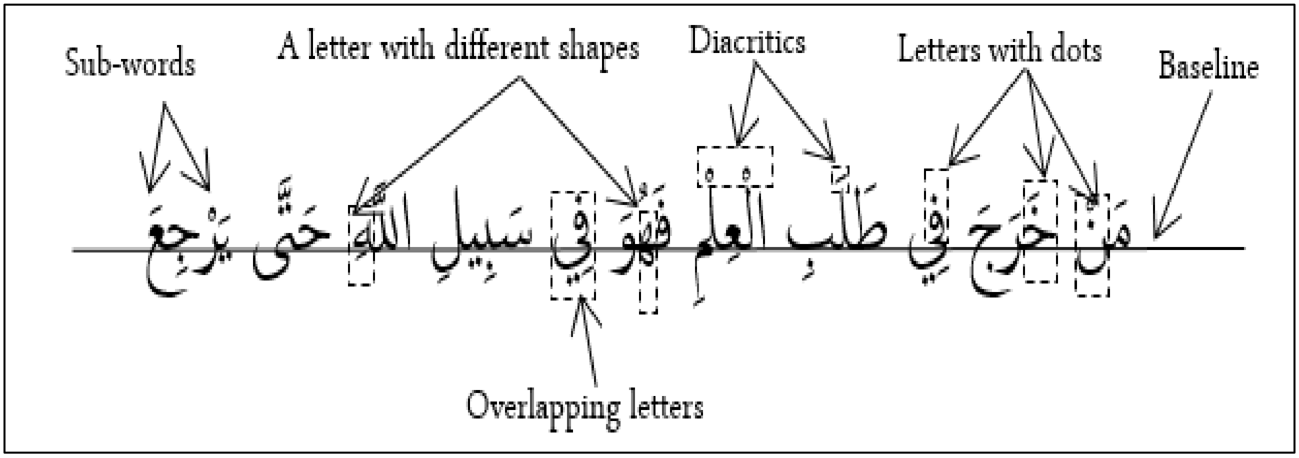

Sub-words
A letter with different shapes
Diacritics
Letters with dots
Baseline
مَنْ خَرَجَ فِي طَلَبِ الْعِلْمِ فَهُوَ فِي سَبِيلِ اللهِ حَتَّى يَرْجِعَ
Overlapping letters

Fig. 1. Arabic script characteristics

TABLE I. ARABIC CHARACTERS WITH DIFFERENT POSITIONS AND SHAPES

| Isolated | Initial | Middle | End |
|---|---|---|---|
| ا | ا | ـا | ـا |
| ب | بـ | ـبـ | ـب |
| ت | تـ | ـتـ | ـت |
| ث | ثـ | ـثـ | ـث |
| ج | جـ | ـجـ | ـج |
| ح | حـ | ـحـ | ـح |
| خ | خـ | ـخـ | ـخ |
| د | د | ـد | ـد |
| ذ | ذ | ـذ | ـذ |
| ر | ر | ـر | ـر |
| ز | ز | ـز | ـز |
| س | سـ | ـسـ | ـس |
| ش | شـ | ـشـ | ـش |
| ص | صـ | ـصـ | ـص |
| ض | ضـ | ـضـ | ـض |
| ط | ط | ـط | ـط |
| ظ | ظ | ـظ | ـظ |
| ع | عـ | ـعـ | ـع |
| غ | غـ | ـغـ | ـغ |
| ف | فـ | ـفـ | ـف |
| ق | قـ | ـقـ | ـق |
| ك | كـ | ـكـ | ـك |
| ل | لـ | ـلـ | ـل |
| م | مـ | ـمـ | ـم |
| ن | نـ | ـنـ | ـن |
| ه | هـ | ـهـ | ـه |
| و | و | ـو | ـو |
| ي | يـ | ـيـ | ـي |

There is no doubt that printed Arabic OCR faces a number of challenges and there is still an intensive need for more research [6]. However, most challenges facing the development of Arabic OCR are due to the characteristics of Arabic script. Arabic script has some features that distinguish it from other languages. Compared to English, the most obvious feature of Arabic script is that it is written cursively from right to left in both printed and handwritten. The greatest challenges are due to the more complex characteristics of Arabic script. In the following section, the characteristics of Arabic script that may complicate recognition will be discussed:

### A. *Shapes and Positions*

The Arabic alphabet has 28 basic letters (see Table 1). However, an Arabic letter may contain four dissimilar shapes in relation to its location inside a word: whether it is an isolated letter, an initial letter (in which a letter is inked from the right side, an ending letter (in which a letter is linked form the left side) or a middle letter (in which a letter is linked from the right and left sides). Thus, the number of letters to be recognized will increase from 28 letters to 125 letters.

### B. *Overlapping characters and Ligatures*

Characters in an Arabic word might be overlapped vertically with or without touching each other (see Figure 1). In particular, some characters are combined and written as a ligatures such as (لا) which is a combination of two letters Lam (ل) and Alf (ا). However, ligatures occurs in Arabic script depending on the type of fonts being used. For instance, in Traditional Arabic font, there are about 220 ligatures whereas Simplified Arabic incorporates about 150 ligatures, [7].

### C. *Diacritics*

Characters in an Arabic word can exist with diacritics or short vowels such as Fat-hah, Dhammah, Mada'ah, Kasrah and Sukkun, as illustrated in Figure 1. These can be placed either over or below the letters as strokes. In addition, Tanwen is considered as a diacritic which is indicated by double Fat-hah, double Dhammah and double Kasrah. One more diacritic that Arabic script has is Shaddah which is similar to the number 3 as it is rotated 90° clockwise.

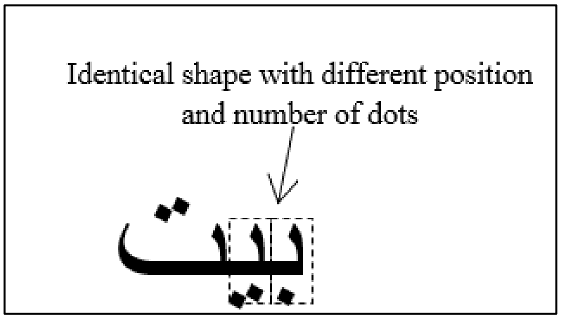


Fig. 2. Two characters (*Ba* and *Ya*) with an identical shape and a different number of dots.

### D. *Cursive*

As mentioned above, Arabic script is a cursive script which means that a word is composed of connected characters. However, six characters (ا ,د ,ذ ,ر ,ز ,و) of the Arabic alphabet are not linked with succeeding letters. This can present a challenge because these characters can divide a word into one or more units as sub-words (see Figure 1).

### E. *Presence of dots*

The Arabic alphabet relies on number and position of dots in order to differentiate between similar letters (see Figure 2). Fifteen characters in the Arabic alphabet have dots. They can be placed below the character, above it or in the middle. Ten of these characters are dotless, three have two dots and two have three dots, as shown in Table 1.

## III. GENERAL ARABIC OCR METHODOLOGY (MODEL)

This section will focus on the methodologies used by printed Arabic OCR systems. Published approaches and systems for Arabic OCR indicate that the process of implementing Arabic OCR consists of five phases: (1) pre-processing; (2) segmentation; (3) feature extraction; (4) classification and (5) post-processing, see figure 3.

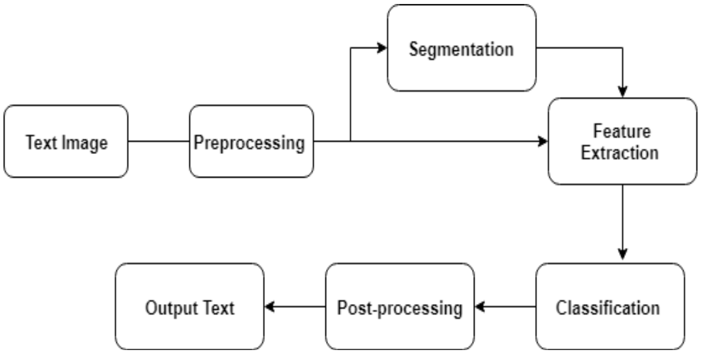


Fig. 3. General printed Arabic OCR methodology.

## A. Preprocessing Phase

This is the first phase of OCR methodology which is responsible for enhancing the readability of the input image. Preprocessing is a combination of algorithms that are applied to the input image in order to reduce noise and alterations, thus simplifying the subsequent phases of OCR methodology [2]. There are various factors that affect the quality of the input image. A study lists the history of image, the printing process, the kind of font, the quality of paper, the condition of the image and the image acquisition as the vital factors that influence the input image quality [2].

Researchers emphasize that the downstream OCR accuracy relies on the quality of the input image [8]. Furthermore, a study states that OCR systems, which report high recognition accuracy on some input images, will report less recognition accuracy on input images that are poor in quality [9]. Thus, the preprocessing phase is a critical stage in OCR development that simplifies the data for the subsequent phases to operate accurately. Generally, several preprocessing operations are employed on the input image: binarization, layout analysis, thinning, smoothing and filtering, size and slant normalization, slant detection, skew detection and baseline detection. However, the selection of these operations, to be applied in the preprocessing, relies upon the conditions of the input image, such as the amount of noise and skew in the input image [10]. In the following section, the preprocessing techniques which are applied in Arabic OCR, will be clarified.

### 1) Binarization

For character recognition, the binarization (sometimes called thresholding) process involves converting an input gray scale image into a binary image, in which a pixel has only two values 0 and 1. The binary image has the critical information, such as the shape of characters. It has been found that increasing processing speed and reducing storage capacity are the key benefits of binarization technique [7], [11]. Researchers suggest selecting the most appropriate method for binarization might separate connected objects or joining isolated objects [12]. A number of studies have confirmed the efficiency of computing the histogram of the gray scale of an image and then detecting a cut-off point as the binarization method [13] [12]. However, some researchers work on recognition without applying binarization methods, such as [14], [15].

### 2) Size Normalization

Since Arabic characters differ in size, as described earlier, size normalization is commonly applied to characters or words by scaling the characters or the words to an adjusted size. This process is crucial for the recognition or classification phase, since some recognition methods are sensitive to dissimilarity in size and position, such as template matching and correlation approaches [16]. A study classified normalization methods into two approaches: moment-based normalization; and nonlinear normalization [17]. It is argued that normally a character is normalized to a standard size for classification [18]. However, in terms of word normalization, applying normalization to a word instead of a character will result in losing critical information [18], [19].

### 3) De-noising

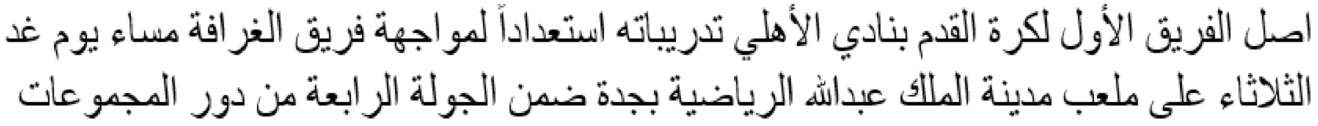
Noise may be presented during the acquisition process via scanners which results in distortions and variations in the input text image. Besides this, very small items in the text image can be reflected as noise [11], which are byproducts of image scanning or binarization and which are not parts of the text. Such noise may has a major impact on the performance of OCR systems [20]. Noise removal is an operation for enhancing the visual quality of the input image [21].

As a solution, several techniques have been introduced that are considered as noise removal methods [22]. These methods include filtering and morphological operations (smoothing) which are conditioning processes in terms of OCR development [2], [23]; for instance, dilation algorithms, which are applied to broken letters, and erosion algorithms which are applied to text images with touching letters [17]. In addition, the median filter approach is commonly used in both printed text images and handwritten text images. For example, a study apply this approach for removing noise in printed Arabic text images [24]. Another example of a study that applies a median filter algorithm in handwritten Arabic text images is in [25].other researchers [26] applied a morphological noise removal method for Arabic printed OCR proposed in [27]. However, a study discovered that letter holes could be filled while applying this method, with lower thresholds, to Arabic text images [26].

In fact, the review suffers from the fact that some printed Arabic OCR studies applied noise removal algorithms without providing information of the applied algorithm, for instance, in [28]. Such approaches, however, should be selected carefully when considering OCR systems. That is, because of the similarity between Arabic letters, any alteration of a letter might change it to another letter. Thus, a perfect noise removal method is able to eliminate noise while preserving the shape of the character [20].

### 4) Skew Detection and Correction

Initially, a text image has zero rotation, yet when physically scanning the image manually, rotation of images up to 20$^{o}$ might occur [5]. This rotation is called skew which results in non-zero skew text images (see Figure 4). The skew can lead to incorrect recognition and baseline detection [29]. It is impossible to segment a text if the text is rotated [30]. As a result, detecting and correcting the skew is critical to OCR applications that rely on segmentation approaches to recognize characters.



اصل الفريق الأول لكرة القدم بنادي الأهلي تدريباته استعداداً لمواجهة فريق الغرافة مساء يوم غد
الثلاثاء على ملعب مدينة الملك عبدالله الرياضية بجدة ضمن الجولة الرابعة من دور المجموعات

Fig. 4. An example of Arabic text skew.

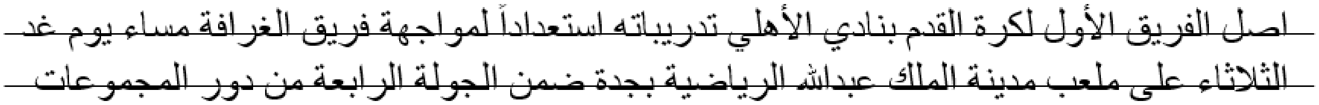

اصل الفريق الأول لكرة القدم بنادي الأهلي تدريباته استعداداً لمواجهة فريق الغرافة مساء يوم غد
الثلاثاء على ملعب مدينة الملك عبدالله الرياضية بجدة ضمن الجولة الرابعة من دور المجموعات

Fig. 5. Baseline detection of a printed Arabic text image.

The process of estimating the skew angle is known as skew detection, whereas the process of rotating the image with the purpose of correcting the skew is called skew correction. A wide variety of skew detection and correction methods have been proposed. A study groups these methods into five groups: projection profile, Hough transform, Fourier transform, nearest neighbor clustering and correlation [31]. The Hough transform is the standard approach for detecting the skew [32]. A method based on the projection profile was introduced in [33]. A researcher has provided a comprehensive review of twenty–five skew detection and correction approaches [34]. The author concludes that further work on more sophisticated methods is still required. The Radon transform method has shown its efficiency for skew correction [35]. Some methods are designed for specific applications and image type. For example, a new method has emerged for Arabic text images in [36]. One study concludes that selecting a skew detection and correction method relies on the image type [37].

### 5) Baseline Detection

As described in the previous section, Arabic characters are joined through a horizontal line called the baseline (see Figure 5). Graphically, the baseline can be described as the line which has the maximal amount of black pixels [38]. This line contains critical information about the text, such as text orientation and position of connection points between Arabic letters [2]. Thus, detecting the baseline is beneficial for many OCR stages, for instance, skew normalization [39], segmentation [40], [41] and structural features extraction such as the character's dots [42].

It has been reported that most Arabic OCR has applied baseline detection methods as a preprocessing step [25] .The baseline detection techniques for Arabic script has been classified into four groups in [36]; namely, horizontal projection methods, the word skeleton method, contour tracing and principle component analysis. Among these, the horizontal projection technique is widely implemented for determine the baseline in Arabic OCR, such as in [43], [2]. Several studies implement a horizontal projection approach in OCR systems for detecting the baseline, such as in [26], [42], [44], [45]. It has been emphasized that the horizontal projection method is simple and efficient for Arabic printed text [43], [46]. However, this method is applicable only for noise-free images, as it fails for unclean images [47].

Another baseline detection approach is the x-y cut proposed in [48] which is based on a horizontal projection method. This method works well for Arabic noisy images, though it fails in the presence of large amounts of noise and skew [47]. Consequently, researchers proposed using a ridge-based text line detecting approach for Arabic text [47]. The former method's efficiency has been tested and recommended for different types of Arabic text images, since it was found to achieve above 96% text line detection accuracy [47].

Researchers summarize the state of the art of baseline detection methods in Arabic script [49]. In summary, for printed Arabic text, the standard horizontal projection method is sufficient for detecting the baseline, since the baseline in printed text is straight. Whereas for handwriting, the baseline is not straight, thus more sophisticated approaches should be considered [50].

### 6) Thinning and Skeletonization

Thinning " skeletonization" can be defined as "the process of peeling off a pattern as many pixels as possible without affecting the general shape of the pattern" [18]. In other words, it involves operations that can be implemented in order to produce the skeleton of text images. Thinning is a crucial processing step for text recognition, in particular for such OCR applications in which extracting the skeleton of a character is essential [2], [32]. However, in terms of the obtaining the skeleton, it must be as thin as possible, connected, and centered [18]. Thinning simplifies the process of the segmentation, future extraction and classification phases as a result of reducing the amount of data that needs to be considered in the input image [25].

Most of the existing thinning algorithms have been designed for general purpose or other text languages [51]–[53]. However, when applying thinning algorithms to Arabic scripts, various obstacles are encountered [49]. One problem is the reduction in the number of dots in some Arabic characters as a result of the thinning process for which the number of dots is a crucial aspect in differentiating between these characters [54].

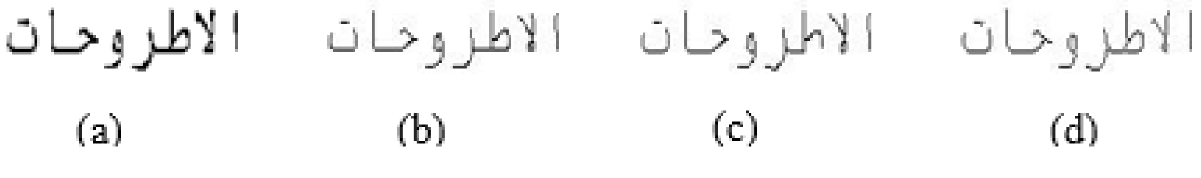


Fig. 6. Example results of different thinning algorithms: (a) original word, (b), (c) and (d) thinned word.

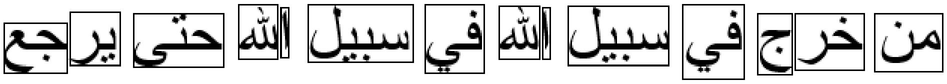


Fig. 7. An example of line segmentation.

Also, dots in Arabic characters are likely to be vulnerable to noise. However, some researchers extract dots of Arabic characters before applying thinning algorithms, in order to overcome this problem [55], [12]. Another problem of thinning algorithms when considering Arabic script concerns preserving the connectedness of Arabic text. Some thinning approaches may not cope well with Arabic text due to its connectivity characteristic [49]. Thus, this should be taken into consideration, when selecting thinning algorithms for Arabic text. Also, since Arabic characters consist of different shapes such as loops and lines, the selected thinning algorithm must be capable of preserving these different shapes.

Therefore as a consequence of specific characteristics of Arabic script detailed above, direct adoption of thinning algorithms, which have been developed for other languages, may not be as effective [24]. As a result of these difficulties, there is comparatively little published work on developing thinning algorithms for Arabic [25], [24].

Some studies introduce thinning algorithms for Arabic letters [56], [57]. However, the proposed algorithms can only deal with isolated Arabic characters. One study provide a thinning algorithm which is designed specifically for printed Arabic script recognition to overcome dis-connectivity and loss of information [58]. This algorithm is applied on Arabic text to illustrate the efficiency of reducing the outline of each word's characters (its number of pixels) thereby overcoming the challenges of Arabic script. Also, the authors propose an experimental framework with new performance measures for the evaluation of thinning algorithms. Figure 6 shows the output of three different thinning algorithms.

## B. *Segmentation Phase*

After the preprocessing phase, an enhanced text image in the sense of low noise and variation, and a necessary amount of character information [2], has been produced. During the segmentation phase, the text image is segmented into small components, with a page being segmented into lines, a line into words and a word into letters [59]–[61]. Segmentation is a crucial step in Arabic OCR system development because of the fact that it plays a vital role in ensuring the success of the subsequent feature extraction and classification stages [3], [46]. However, the author in [46] stresses that misrecognition can arise by applying a poor segmentation method. As a result, this stage will have a critical impact on the recognition rate of the text [7].

As explained previously, one of the main challenges facing Arabic OCR development is the cursiveness of Arabic script. Segmentation of Arabic text thus can be more difficult and time consuming for the development of Arabic OCR systems [3]. Correspondingly, segmentation has been considered as the main contribution for increasing the recognition error rate in Arabic OCR systems [46], [62], [63].

Generally, segmenting a text image can be graded into two types: external segmentation; and internal segmentation [64]. While the former type deals with the isolation of different writing objects such as, paragraphs, sentences and words, the latter deals with the isolation of characters [64], [65].

### 1) *External Segmentation*

External segmentation refers to the document layout analysis, in particular page decomposition. Document layout analysis is accomplished in order to identify the physical structure of a page [66]. As far as offline OCR development is concerned, page analysis is a basic step which segments the image into its different logical parts with the identical type of information, such as graphs, text and tables. Page layout analysis is performed in two approaches: structural analysis by which a page is decomposed into blocks of the page elements, such as paragraphs and words; and functional analysis by which a page is decomposed into functional elements such as title and abstract [41], [65], [66].

With respect to Arabic document processing, page decomposition refers to the isolation of text lines of a texture region and the segmentation of words and sub-words [5], [7], [67], since it is restricted to text images [5]. Applying a fixed threshold to Arabic text documents to determine text lines is the standard method [5], [68], [69]. However, this method fails with a skewed text image [40].

Methods based on histogram projection are considered as conventional approaches for isolating lines and words in Arabic text documents [68], [70]. Several studies have relied on horizontal projection techniques for segmenting Arabic text images into lines, such as in [71]–[76], [28]. [72] It is recommended horizontal projection be applied for text images because of its advantages in reducing computational load and its simplicity of implementation [71]. Moreover, horizontal projection is an appropriate method for locating text lines in Arabic printed text, since the text lines in printed text are straight [50].

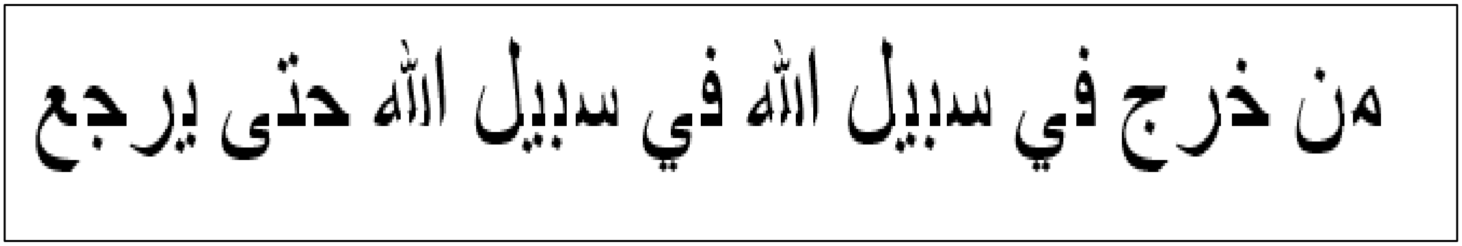


Fig. 8. Segmenting Arabic words into their characters.

For line segmentation, researchers in Arabic OCR determine words in a line of text by inspecting the vertical projection [14], [28]. (See Figure 7). This method depends on the estimation of the minimum space between words. However, it was pointed out in the Arabic script characteristics section above that some Arabic characters are not linked with succeeding letters, thus this results in a word having with one or more connected components (sub-words), as shown in

Figure 7. To overcome this issue, methods based on vertical projection consider that the width of spaces between sub words is smaller than the width of the spaces between words [14].

Generally, it is relatively easy to segment a text line into words in printed text images, compared to handwritten text images which involve overlapping and touching characters by using vertical projection histogram profiles [37], [59], [25]. However, some Arabic fonts contain characters that vertically overlap, such as the Traditional font type. Thus, Arabic script even in printed form can contain touching and overlapping characters, so algorithms that have been designed to overcome this challenge for handwritten script may be utilized for printed Arabic. For example, the authors in [77] have developed a method based on the connected components that analyses the distance between connected components in order to segment handwritten words.

#### 2) *Internal segmentation*

Internal segmentation deals with segmenting a word into characters. When reviewing segmentation methods in the literature, a major complication arises concerning the classification of word segmentation approaches. For instance,[5] a study classifies Arabic OCR systems based on word segmentation into ‘segmentation based systems’, which is based on analytical techniques where a word is segmented into characters, and ‘segmentation–free systems’, which is based on recognizing a word as a unit without segmentation [72]. Some researchers discuss word segmentation in terms of implicit and explicit segmentation [73], [78]. Others classify word segmentation in terms of techniques which have been applied to segmenting a word, such as [59], [46], [3]. Researchers organize segmentation methods for Arabic script into holistic approaches and analytical approaches [25].

Mostly, Arabic OCR systems have been developed by two main paradigms: holistic approaches (segmentation–free) which require a large lexicon of Arabic words, and analytical approaches (segmentation based) where a word is segmented into units and each unit is recognized separately.

### C. *Holistic Approach*

Segmentation-free or holistic Arabic OCR systems perform the recognition on the entire word as a unit without segmenting the word or recognizing characters separately [2]. Several studies have investigated the holistic approach for printed Arabic scrip OCR such as in [79], [16], [8]. OCR systems based on a holistic approach require tracing the feature of the entire word and dealing with words instead of characters. As a result, this approach is restricted to recognizing a word against a lexicon [2]. Moreover, this approach has the challenge of how to deal with the large lexicon size of Arabic words. It is claimed that systems based on this type of segmentation are not useful for general text recognition. A study suggests this approach for systems in which a lexicon is statically defined, such as bank cheque recognition where vocabulary is limited [80].

### D. *Analytical Approach*

For the analytical or segmentation based approach, Arabic OCR systems segment words into smaller units like characters (see Figure 8). In the typical Arabic OCR system, the analytical approach is divided into two approaches: explicit segmentation and implicit segmentation.

#### 1) *Explicit Segmentation*

The explicit segmentation approach, which is also called dissection segmentation, attempts to segment a word into smaller units. These units could be characters, strokes or loops. Researchers argue that there are two classes of explicit segmentation, which are: direct segmentation and indirect segmentation [81]. In the former, a word is directly segmented into characters exploiting a set of heuristics, while in the latter, a word is divided into smaller segments which can be characters or marks that over segmented characters, such as strokes.

Projection analysis is considered as one of the earliest applied dissection methods on Arabic character segmentation [46], [68], [70]. The projection method of the text image aims to reduce 2D information into 1D in order to simplify the character segmentation process. A method based on a modulated histogram of the image has been proposed in [82]. However, this method has been tested on specific Arabic fonts which do not contain overlapping and ligatures. Consequently, this method would not be appropriate for Arabic fonts that have ligatures, such as traditional Arabic font [3], [62].

Another histogram projection method is presented for printed Farsi word segmentation in [83] which is also applicable to Arabic script, as Arabic script is similar to Farsi script [3]. However, this method is font dependent and ineffective in segmenting small font sizes. Although many of the other techniques based on projection analysis have been devolved for Arabic script such as in [84], [62], [85], it seems that no projection based segmentation algorithm is accurate in segmenting Arabic text [50].

Instead of applying projection analysis methods, contour–based algorithms, which are used for dissection segmentation that rely on the skeleton or contour of Arabic words, are used to simplify the Arabic word segmentation such as in [78]. Other methods rely on white space and pitch finding techniques for segmenting Arabic words [46], [74]. However, a major criticism of the explicit approach is that it is expensive because of the requirement of finding the optimum word from the arrangement of segmented units [81]. The researchers in [71] conclude that an accurate segmentation may not be acquired by relying on dissection segmentation approaches.

#### 2) *Implicit Segmentation*

In OCR systems based on implicit segmentation, the segmentation phase and recognition phase are performed simultaneously [3]. In other words, a word is segmented into characters while being recognized without segmentation in advance [46]. Straight segmentation and recognition based segmentation are also referred to as implicit segmentation [46]. This segmentation approach searches the text image for components that match predefined classes. The principle of implicit segmentation is to utilize a sliding widow to segment the word image into frames of fixed width on which classification relies to make a decision [86]. Owing to challenges in segmentation of cursive scripts such as Arabic, researchers use the implicit segmentation approach in order to overcome the problems of word segmentation [80]. In principle, by applying this type of segmentation, there is no need for a specific dissection algorithm for Arabic script segmentation and the accuracy performance relates to the classification performance [87]. Thus, some researchers implement techniques based on implicit segmentation in order to improve recognition accuracy of Arabic OCR, such as in [88], [12], [89].

### E. Feature Extraction Phase

Once the text image is segmented into isolated regions (such as character, part of character), the next step is feature extraction which is the process of obtaining distinguishing attributes of the segmented character to be utilized by the next phase which is classification [90]. Feature extraction is the most significant level that heavily influences overall OCR performance [11], [25], [60]. The feature extraction stage is correlated with other OCR stages, such as preprocessing and classification stage. In other words, the authors in [91] point out that the selection of feature extraction methods depends on the output of the preprocessing stage. For instance, some techniques for feature extraction work on skeletons, whereas others work on grayscale images. Moreover, the set of features extracted must match the specification of the selected classifier [2].

In terms of OCR performance, feature extraction plays a critical role in achieving high accuracy performance [11], since the feature extraction stage has the contributes to the success of the classification step [60]. However, selection of feature types is a major issue in OCR development [92]. Researchers recommend that the feature extraction methods should be independent of scalable font characteristics such as font styles, font types, font sizes and should be able to describe and distinguish different patterns effectively [92], [93]. In other words, a study emphasizes that the key purpose of selecting good features is to maximize the effectiveness and the efficiency of the OCR system minimizing the complexity and processing time simultaneously [94].

Among OCR system development, researchers propose various types of features. Such features can be categorized into three groups: structural features; statistical features; and global transformation feature [5]. In the following, these features will be discussed in the context of recognizing Arabic script.

#### 1) Structural Features

Structural features illustrate a text image in terms of its topological and geometrical characteristics by using its local and global properties [2], [92]. In case of Arabic script, lines, dots, loops, holes, strokes and zigzags are some structural features [92]–[99]. Considering Arabic script characteristics, some characters have common primary shapes and they can only be differentiated by the number and location of their dots. Thus, the researchers in [59] claim that structural features have been commonly used for Arabic script in order to capture the dot information of characters explicitly.

On the other hand, A study argues that structural feature methods are not capable of discriminating between characters having similar shapes [100]. Similarly, a study reports that relying on the structural features of Arabic script may result in misrecognition, owing to the small difference between Arabic letters [92]. It is mentioned in [5] that extracting structural features of Arabic characters is a challenging task. Furthermore, it is claimed that Arabic OCR systems implementing structural feature methods are processed exhaustively [101]. Likewise, various studies have reported that another complication of applying structural features is that it involves expensive preprocessing techniques, such as skeletonization which may result in character shape distortion and loss of structural feature data [60], [59], [102]. Therefore, research on Arabic OCR has been carried out on other feature extraction approaches, as will be discussed below, that are effective in reducing process time and improving performance accuracy [101].

#### 2) Statistical Features

Statistical features are derived from statistical representation of patterns which provide a measurable event of interested patterns. Researchers in Arabic OCR systems adopt different approaches to produce statistical features. Some examples of the approaches, which have been applied for representing Arabic characters, are zoning, moments, characteristic loci, histograms and crossing [92], [5], [2], [25].

The zoning method divides the character image into serval overlapping and non-overlapping regions. Then, the density of each region pixel is analysesd and used as a feature [92], [103].

The moment method is a common statistical feature approach that has been applied in patter recognition applications [26]. Moments, including Legendre moments, Zernike moments, central moments, pseudo-Zernike moments and Hu moments, extract geometric features in an image, such as, the shape area of a pattern and the center of the mass [17], [104] and [105]. Several studies in printed Arabic script, such as, [106], [107]–[109], have applied moment invariants as a feature vector.

In short, it is claimed that statistical features for pattern representation are easy to extract [92], [2]. Moreover, such features can be effective in recognition systems and providing high speed and low complexity implementation [60], [110]. However, special attention to the prepossessing techniques should be given, since misrecognition may accrue due to poor prepossessing techniques [5]. Nevertheless, the fundamental issue is to determine a set of statistical features, which need to be the most representative data of a pattern, maximizing the performance accuracy and minimizing the processing time simultaneously.

As a result, researchers call for investigating other statistical features which maximize the performance accuracy and minimize the processing time [2], [64].

#### 3) Global Transformation Feature

The global transformation method is applied to convert a skeleton or contour of a pattern by a linear transform into a form that reflects the most relevant features of the transformed pattern [64]. Numerous global transformation methods have been used in developing Arabic OCR systems. An example of such methods is the Fourier descriptor which represents the characteristic of a pattern in a frequency domain [111]. The Fourier descriptor has been applied to Arabic script, such as in [8], [27]. Another method is the Hough transform which detects lines in binary images and then define the parameter of the lines [18]. Other, such as in [112], [113], utilized the Hough transform for extracting features from Arabic script. Also, some other global transformation methods that have been applied for Arabic OCR for feature extraction are the direction codes method such as Freeman's chain code in [28], Wavelets in [114] and Walsh transformation in [107].

Overall, it is claimed that global transformation feature techniques have several advantages over structural and statistical approaches. For example, they are applicable for new fonts and easily implemented. Another advantage is that they are robust to noise and variation. However, they might require the implementation of other features in order to obtain high accuracy performance.

In conclusion, the feature extraction stage plays a critical role in Arabic OCR development in which distinguishing attributes are extracted and it is clear that each Arabic OCR developer needs to apply different feature extraction approaches. Still, good features are required, which assist in distinguishing a character from other characters and maximize the accuracy performance simultaneously. Furthermore, these features must be selected specifically for a selected classifier. Some researchers apply different feature extraction methods in combination. However, this may cause extra complications for the implementation [8].

### F. Classification Phase

The classification phase has the responsibility for assigning a pattern into a pre-classified class based on the features of the pattern which have been extracted in the previous phase [18]. The pre-classified classes can be words, sub-words, characters or strokes, based on the OCR approach used [6]. There are a number of different classification approaches that have been applied for Arabic OCR, such as Hidden Markov Models (HMM), Support Vector Machines (SVM), K-nearest neighbour.

SVM, which is a binary classifier, has been used in the implementation of printed Arabic OCR systems [106], [95], [115]. (For a comprehensive review of applying SVM to Arabic OCR, refer to [116]). However, classifiers based on SVM are mostly applied to a small set of data due to the high complexity of training and processing time [117], [118]. Another classification technique that has been applied to printed Arabic OCR are Hidden Markov Model (HMM) based techniques. HMMs are statistical models that are considered as being one of the most efficient for recognition applications especially for speech recognition [17]. Therefore, researchers in OCR have implemented HMMs for OCR in order to obtain high performance OCR systems, such as in [72], [119]–[122].

### G. Post-processing Phase

Post-processing is the final stage of the development of Arabic OCR. The objective of this step is to enhance the recognition accuracy by detecting and correcting linguistic misspellings in the produced OCR text without human intervention. Research studies on Arabic OCR have implemented post-progressing methods in order to improve the output, such as [123], [124]. It is worth mentioning that three main elements should be considered in correcting OCR output: non-word errors correction; isolated word errors correction; and context–based word correction [50]. Generally, post-processing methods can be categorized into two main approaches: lexicon-based methods; and context-based (statistical) methods [125].

The typical technique for correcting the mistakes of Arabic OCR outputs is the lexicon-based method which requires the utilization of an Arabic dictionary, such as in [126], [127]. This technique corrects errors without considering any contextual information in which the errors appear. Therefore, a problem might occur with using this approach when a word is misrecognized by an OCR system and is also in the lexicon (these are called real-word errors) such as, *Fear* for *Tear*. This occurs in many languages such as Arabic in which a large fraction of three characters sequences are corrected words. Consequently, only non-word errors can be corrected, since this method is comparing the recognized words with the words that are in the dictionary. Also, this approach requires a wide-ranging lexicon that consists of all single words. However, the Arabic language has various dialects and it is also a triglossic language with three forms – modern standard Arabic and classical Arabic [128] and mixtures of the two. Therefore, this approach is less appropriate for Arabic language since building a single lexicon for Arabic language is more complicated.

On the other hand, context-based (statistical) methods take into account the contextual information in which the misrecognized words appear. A few studies have implemented statistical language models for improving the recognition accuracy of Arabic OCR systems, such as in [129], [130]. Using such methods will help overcome the problem of correcting real-word errors. Moreover, they are also useful in correcting word errors that might have several potential corrections, since these techniques can correct word errors based on grammatical concepts and semantic context [131].

Recently, there have been several attempts to provide systems for correcting Arabic OCR output. For instance, the authors in [123] propose a system for Arabic OCR output correction based on Google online suggestions within Microsoft Office Word. On the other hand, the authors in [131] describe a context-based technique for detecting and correcting Arabic OCR errors. Although there are some studies on applying context-based methods for correcting Arabic OCR output, more research is needed on investigating the use of Arabic contextual information for OCR output correction [132].

## IV. Performance Evaluation

OCR performance evaluation can be classified into two types: black-box evaluation and white-box evaluation. In the former, an entire OCR engine is treated as an indivisible unit, so the submodules of the OCR system are not known to the evaluator, whereas with the white-box evaluation, each submodule of the OCR system is evaluated if the submodules are accessible [133]. Performance evaluation of OCR systems is essential for monitoring progress of OCR systems development, assessing the effectiveness of OCR algorithms, identifying open areas for further research and providing scientific justification for the performance of OCR systems [134], [135]. Although the performance evaluation of OCR systems is important, there has been very little work focus on empirical evaluation of Arabic OCR systems, such as in [135], [136] and a recent study in [1]. Furthermore, these evaluation studies have conducted a black-box evaluation on Arabic OCR systems as the submodules are not accessible. Thus, only the overall performances of Arabic OCR have been reported.

Performance evaluation in research areas of pattern recognition is facing several obstacles [137]. For Arabic OCR, conducting performance evaluation is challenging as no standard dataset is available [138], [94]. Moreover, most Arabic OCR systems are evaluated in terms of character accuracy which is a general metric, such as performance results reported in [138], [135], [136]. This accuracy metric is insufficient to assess how Arabic OCR systems are overcoming the challenges of Arabic text. However, a study suggests a new set of objective performance metrics for evaluation Arabic OCR with respect to the challenges of Arabic script which are character accuracy based on character position, dot character accuracy, zigzag-shaped character accuracy, loop-shaped character accuracy and diacritics accuracy [139].

## V. Discussion and Future Directions

This paper has overviewed the main stages used in printed Arabic OCR. It main aim is to reveal the current status of printed Arabic OCR. Although there are various attempts to solve the problems of Arabic text recognition, there is still a crucial need for more research.

In an attempt to evaluate the status of printed Arabic OCR and support the claim that more research is needed in many areas, we used Google scholar to search for scientific research publications using phrases that are related to Arabic text recognition. The findings are summarized in Table 2. The table shows the search phrases used and the search results returned by Google Scholar. It is apparent from the table that there is a lack of Arabic OCR research as comparatively very little research has focused on Arabic OCR compared to studies in OCR for other languages. For example, there 322,000 results were returned for the more general search query ‘OCR’, whereas there were only 956 results returned for the more specific search query ‘Arabic OCR’.

In order to provide a measure of the coverage of research in a particular area, we can estimate the probability that a particular research paper will be in a more specific topic area compared to the more general topic area. For example, we may be interested in the general topic area “single font OCR” and wish to see how much research has been published in the more specific topic area “single font Arabic OCR” in comparison. We can estimate the probability $p$ that the more general topic will be concerned with the more specific topic as $p = s / g$ where $s$ is the count of the number of papers found for the specific topic compared to the count $g$ of the number of papers found for the more general topic. Then we can define the ‘Information Coverage’ $I$ associated with the specific topic in relation to the more general topic as $I = – \log p$. If this value is high compared to other specific vs. general comparisons for the same overall topic (e.g. in relation to Arabic OCR vs. OCR in general), then this reflects that research may under-represented in this area.

This analysis has been done using the values from Table 2 and graphed in Figure 9. In the Figure, we see that except for papers on Arabic OCR concerning easy fonts and diacritics, the remaining topics have higher Information Coverage values meaning that there have been less papers published in these areas proportionately compared to papers published in the more general (non-Arabic) areas. We can use Figure 9 to help gauge the present status of printed Arabic OCR research as it highlights some open areas which need more research. This is based on the number of publications for single, omni and multi font OCR concerning various elements that are related to text recognition concerning easy fonts, complicated fonts, diacritics, page layout, multi-language and noisy documents. In particular, for Arabic text images which contain complicated fonts, there are still many gaps in the research. Furthermore, for single, omni and multi font Arabic OCR on multi-language text images, intensive further research is needed.

Figure 10 plots the number of papers per 5-year period for the top 100 Google Scholar searches using the Arabic OCR related phrases. A number of striking results are apparent in Figure 10. For example, publications for Arabic OCR peak since 2005. Also, the numbers of papers for printed Arabic OCR decrease since 2005 (this could be because researchers have focussed on handwritten Arabic OCR). Furthermore, it is apparent that the smallest numbers of papers in the period of Arabic OCR research are papers related to noisy documents.

Note that the earliest research papers for the ‘Arabic OCR’ search query are from 1985, which is a result of the top-100 ranking returned by Google Scholar. If, however, we restrict the range of years for which we search, we find that the first papers returned by Google Scholar appear in the 1970 to 1980 period. In contrast, the author in [9] states that text recognition research first originated in 1940, and papers related to the ‘text recognition’ query appear in Google Scholar from the 1960s.

TABLE II. GOOGLE SCHOLAR SEARCH RESULTS FOR ARABIC OCR RELATED PHRASES

| Google Scholar search phrase | | Number of papers |
|---|---|---|
| 1 | +"Arabic OCR" | 956 |
| | +"OCR" | 322,000 |
| 2 | +"OCR" + "Arabic printed text" | 247 |
| | +"OCR" + "printed text" | 7,190 |
| 3 | +"Arabic OCR" + "diacritics" | 302 |
| | +"OCR" + "diacritics" | 1,800 |
| 4 | +"Arabic OCR" + "page layout" | 54 |
| | +"OCR" + "page layout" | 3,360 |
| 5 | +"Arabic OCR" + "multi-language " | 20 |
| | +"OCR" + "multi-language" | 866 |
| 6 | +"Arabic OCR" + "Omni font" | 60 |
| | +"OCR" + "Omni font" | 380 |
| 7 | +"Arabic OCR" + "single font" | 61 |
| | +"OCR" + "single font" | 717 |
| 8 | +"Arabic OCR" + "multi-font " | 149 |
| | +"OCR" + "multi-font" | 1270 |
| 9 | +"Arabic OCR" + "noisy document" | 12 |
| | +"OCR" + "noisy document" | 515 |
| 10 | +"Arabic OCR" + " Simplified Arabic " + "single font" | 20 |
| | +"Arabic OCR" + " Simplified Arabic " + "Omni font" | 22 |
| | +"Arabic OCR" + " Simplified Arabic " + "multi font" | 48 |
| 11 | +"Arabic OCR" + " Advertising Bold " + "single font" | 5 |
| | +"Arabic OCR" + " Advertising Bold " + "Omni font" | 5 |
| | +"Arabic OCR" + " Advertising Bold " + "multi font" | 15 |
| 12 | +"Arabic OCR" + "diacritics" + "single font" | 30 |
| | +"Arabic OCR" + "diacritics" + "Omni font" | 36 |
| | +"Arabic OCR" + "diacritics" + "multi font" | 62 |
| 13 | +"Arabic OCR" + " page layout " + "single font" | 8 |
| | +"Arabic OCR" + " page layout " + "Omni font" | 9 |
| | +"Arabic OCR" + " page layout " + "multi font" | 11 |
| 14 | +"Arabic OCR" + " multi-language " + "single font" | 17 |
| | +"Arabic OCR" + " multi-language " + "Omni font" | 9 |
| | +"Arabic OCR" + " multi-language " + "multi font" | 29 |
| 15 | +"Arabic OCR" + " noisy document " + "single font" | 1 |
| | +"Arabic OCR" + " noisy document " + "Omni font" | 2 |
| | +"Arabic OCR" + " noisy document " + "multi font" | 3 |

From the review of each stage used in Arabic OCR, the following observations have been noted for possible research directions:

- All the reviewed research of printed Arabic OCR have used the general OCR methodology which involves the five stages; pre-processing; feature extraction; segmentation; classification and post-processing. However, the following are still open questions: 'Is the current OCR methodology the most effective for designing Arabic OCR?' and 'Are there alternative methodologies that might yield better results for Arabic OCR'?

- From the review of each OCR stage, it is apparent that the most challenging task in the development of Arabic OCR is the segmentation task. Although previous studies have presented different segmentation techniques for Arabic OCR, these studies have not provided the accuracy performance of these techniques. Since only the overall OCR performances have been reported, it is difficult to gain an insight into which segmentation techniques perform better for printed Arabic OCR. A performance evaluation tool should be developed to assess the different segmentation techniques.

- The pre-processing stage review given in this study reached the conclusion that direct adoption of pre-processing methods which are designed for general purposes might be not applicable for Arabic script. Thus, developing pre-processing methods that consider the specific characteristics of Arabic script is needed.

- Most of the proposed methods for feature extraction in Arabic OCR have been adopted from methods that have been developed for other languages without considering the characteristics of Arabic script. Such methods may not be the most appropriate for accurate recognition. The characteristics of Arabic script should be taken into consideration when selecting a feature extraction method that is able to distinguish between Arabic characters.

- The studies on performance evaluation of printed Arabic OCR have used a black-box evaluation method which can only provide the overall performance of OCR systems. For more insight into which OCR stage is causing the most problems, a white-box evaluation, where each component of the system is accessible, is required.
- Performance evaluation of printed Arabic OCR suffers from the lack of public datasets. For objective performance evaluation, an accurate and free printed Arabic dataset is essential.
- Our investigation indicates the need for more research on Arabic OCR output correction with the use of Arabic contextual information.

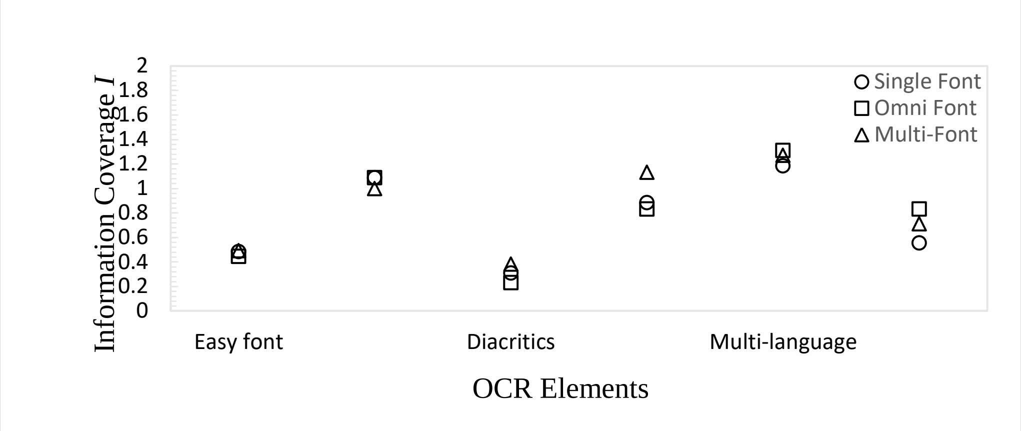


Fig. 9. The present status of printed Arabic OCR based on the number of publications for different OCR elements.

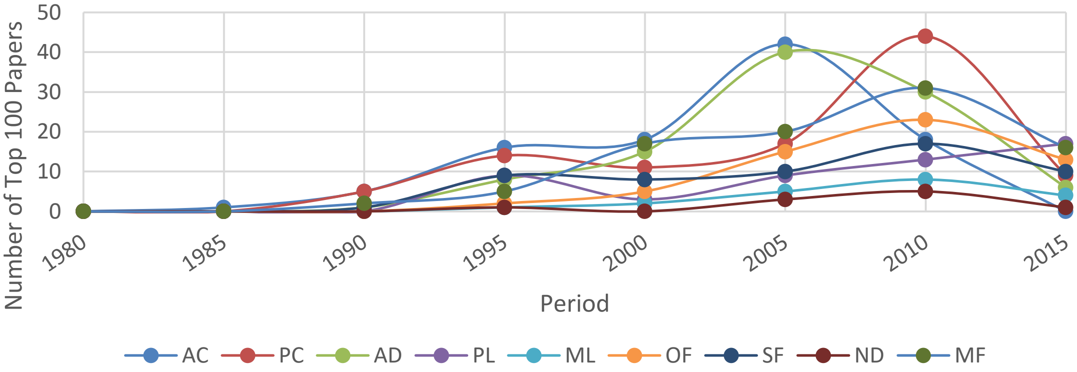


Fig. 10. Number of papers per 5-year period in the top 100 results returned by Google Scholar for different Arabic OCR related search phrases. AC = 'Arabic OCR'; PC = 'Arabic printed text'; AD = 'Arabic diacritics'; PL = 'Arabic OCR + page layout'; ML= 'Arabic OCR + multi-language'; OF = 'Arabic OCR + omni font'; SF = 'Arabic OCR + single font'; MF = 'Arabic OCR + multi font'; ND = 'Arabic OCR + noisy document'.

## VI. Conclusion

This paper has provided a comprehensive literature review of printed Arabic text recognition. At first, the specific characteristics of Arabic script that challenge the recognition process have been discussed. Then, the general methodology of printed Arabic OCR has been presented. This methodology was divided into five stages: preprocessing; segmentation; feature extraction; classification; and post-processing. Techniques applied at each stage of Arabic OCR have been discussed. Also, the issues related to the performance evaluation have been reviewed. Finally, we analyzed the remaining problems in the field of printed Arabic OCR and provide several direction for future research.

## References

[1] M. Alghamdi and W. Teahan, "Experimental evaluation of Arabic OCR systems," PSU Res. Rev., vol. 1, no. 3, pp. 229–241, 2017.

[2] B. Al-Badr and S. A. Mahmoud, "Survey and bibliography of Arabic optical text recognition," Signal Processing, vol. 41, no. 1, pp. 49–77, 1995.

[3] Y. M. Alginahi, "A survey on Arabic character segmentation," International Journal on Document Analysis and Recognition, vol. 16, no. 2. pp. 105–126, 2013.

[4] M. Saad and W. Ashour, "OSAC: Open Source Arabic Corpora," 6th Int. Conf. Electr. Comput. Syst. (EECS'10), Nov 25-26, 2010, Lefke, Cyprus., pp. 118–123, 2010.

[5] M. S. Khorsheed, "Off-line Arabic character recognition - A review," Pattern Analysis and Applications, vol. 5, no. 1. pp. 31–45, 2002.

[6] B. M. Al-Helali and S. A. Mahmoud, “Arabic Online Handwriting Recognition (AOHR): A Survey,” ACM Comput. Surv., vol. 50, no. 3, p. 33, 2017.

[7] M. S. M. El-Mahallawy, “A Large Scale HMM-Based Omni Font-Written OCR System for Cursive Scripts,” Faculty of Engineering, Cairo University Giza, Egypt, 2008.

[8] M. Khorsheed and W. Clocksin, “Multi-font Arabic word recognition using spectral features,” Pattern Recognition, 2000. …, no. 2, pp. 543–546, 2000.

[9] B. H. Al-Badr, “A Segmentation-free Approach to Text Recognition with Application to Arabic Text,” University of Washington, Seattle, WA, USA, 1995.

[10] R. J. Kannan and S. Subramanian, “An adaptive approach of Tamil character recognition using deep learning with big data-A survey,” Adv. Intell. Syst. Comput., vol. 337, pp. 557–567, 2015.

[11] A. Lawgali, “A Survey on Arabic Character Recognition,” vol. 8, no. 2, pp. 401–426, 2015.

[12] B. Al-Badr and R. M. Haralick, “Segmentation-free word recognition with application to Arabic,” Proc. 3rd Int. Conf. Doc. Anal. Recognit., vol. 1, pp. 355–359, 1995.

[13] K. Jumari and M. Ali, “A survey and comparative evaluation of selected off-line arabic handwritten character recognition systems,” J. Teknol., vol. 36, pp. 1–17, 2012.

[14] M. Sarfraz, S. N. Nawaz, and A. Al-Khuraidly, “Offline Arabic text recognition system,” 2003 International Conference on Geometric Modeling and Graphics 2003 Proceedings. pp. 30–35, 2003.

[15] T. Pavlidis, “Recognition of printed text under realistic conditions,” Pattern Recognit. Lett., vol. 14, no. 4, pp. 317–326, 1993.

[16] B. Al-Badr and R. M. Haralick, “A segmentation-free approach to text recognition with application to Arabic text,” Int. J. Doc. Anal. Recognit., vol. 1, no. 3, pp. 147–166, 1998.

[17] A. M. AbdelRaouf, “Offline printed Arabic character recognition,” 2012.

[18] M. Cheriet, N. Kharma, C. Liu, and C. Suen, Character recognition systems: a guide for students and practitioners. 2007.

[19] W. Cho, S. W. Lee, and J. H. Kim, “Modeling and recognition of cursive words with hidden Markov models,” Pattern Recognit., vol. 28, no. 12, pp. 1941–1953, 1995.

[20] F. Drira and F. Lebourgeois, “Denoising textual images using local/non-local smoothing filters : A comparative study,” Proc. - Int. Work. Front. Handwrit. Recognition, IWFHR, pp. 521–526, 2012.

[21] Z. Shi, S. Setlur, and V. Govindaraju, “Guide to OCR for Arabic Scripts,” V. Märgner and H. El Abed, Eds. London: Springer London, 2012, pp. 79–102.

[22] A. Buades, B. Coll, and J. Morel, “A Review of Image Denoising Algorithms, with a New One,” Multiscale Model. Simul., vol. 4, no. 2, pp. 490–530, 2005.

[23] R. Gonzalez and R. Woods, Digital image processing. 2002.

[24] H. Al-ani, N. Ban, and H. M. Abass, “Journal of Computing::Printed Arabic Character Recognition using Neural Network,” vol. 5, no. 1, pp. 64–66, 2014.

[25] A. M. Al-Shatnawi, F. H. Al-Zawaideh, S. Al-Salameh, and K. Omar, “Offline Arabic Text Recognition -- An Overview,” World Comput. Sci. Inf. Technol., vol. 1, no. 5, pp. 184–192, 2011.

[26] I. Ahmed, S. A. Mahmoud, and M. T. Parvez, “Guide to OCR for Arabic Scripts,” V. Märgner and H. El Abed, Eds. London: Springer London, 2012, pp. 147–168.

[27] S. A. Mahmoud, “Arabic character recognition using fourier descriptors and character contour encoding,” Pattern Recognit., vol. 27, no. 6, pp. 815–824, 1994.

[28] S. Taha, Y. Babiker, and M. Abbas, “Optical character recognition of arabic printed text,” SCOReD 2012 - 2012 IEEE Student Conf. Res. Dev., pp. 235–240, 2012.

[29] A. M. Al-Shatnawi and K. Omar, “Skew Detection and Correction Technique for Arabic Document Images Based on Centre of Gravity Atallah Mahmoud Al-Shatnawi and Khairuddin Omar Department of System Science and Management , Faculty of Information Science and Technology,” J. Comput. Sci., vol. 5, no. 5, pp. 363–368, 2009.

[30] I. S. I. Abuhaiba, “Skew Correction of Textural Documents,” J. King Saud Univ. - Comput. Inf. Sci., vol. 15, pp. 73–93, 2003.

[31] C. Sun and D. Si, “Skew and slant correction for document images using gradient direction,” in Proceedings of the Fourth International Conference on Document Analysis and Recognition, 1997, vol. 1, pp. 142–146.

[32] J. R. Parker, “Algorithms for Image Processing and Computer Vision,” Vasa, p. 504, 2011.

[33] H. S. Baird, “The Skew Angle of Printed Documents,” in Document Image Analysis, 1995, pp. 204–208.

[34] J. Hull, “Document image skew detection: Survey and annotated bibliography,” Ser. Mach. Percept. Artif. …, pp. 40–64, 1998.

[35] J. X. Dong, P. Dominique, A. Krzyzak, and C. Y. Suen, “Cursive word skew/slant corrections based on Radon transform,” Proc. Int. Conf. Doc. Anal. Recognition, ICDAR, vol. 2005, pp. 478–482, 2005.

[36] A. Al-Shatnawi and K. Omar, “Methods of Arabic Language Baseline Detection – The State of Art,” J. Comput. Sci., vol. 8, no. 10, pp. 137–143, 2008.

[37] Y. Alginahi, “Preprocessing Techniques in Character Recognition,” pp. 1–20, 2010.

[38] F. Shafait, Adnan-ul-Hasan, D. Keysers, and T. M. Breuel, “Layout analysis of urdu document images,” in 10th IEEE International Multitopic Conference 2006, INMIC, 2006, pp. 293–298.

[39] K. Bin Omar, R. Bin Mahmoud, M. N. Bin Sulaiman, and a. R. Bin Ramli, “The removal of secondaries of Jawi characters,” 2000 TENCON Proceedings. Intell. Syst. Technol. New Millenn. (Cat. No.00CH37119), vol. 2, pp. 149–152, 2000.

[40] A. Amin, “Off-line Arabic character recognition,” Pattern Recognit., vol. 31, no. 5, pp. 517–530, 1998.

[41] N. Arica and F. T. Yarman-Vural, “Optical character recognition for cursive handwriting,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 24, no. 6, pp. 801–813, 2002.

[42] R. El-Hajj, L. Likforman-Sulem, and C. Mokbel, “Arabic handwriting recognition using baseline dependant features and hidden Markov modeling,” in Proceedings of the International Conference on Document Analysis and Recognition, ICDAR, 2005, vol. 2005, pp. 893–897.

[43] M. Pechwitz and V. Margner, “Baseline estimation for Arabic handwritten words,” in Proceedings - International Workshop on Frontiers in Handwriting Recognition, IWFHR, 2002, pp. 479–484.

[44] A. Zidouri, M. Sarfraz, S. N. Nawaz, and M. J. Ahmad, “PC based offline Arabic text recognition system,” Seventh Int. Symp. Signal Process. Its Appl. 2003. Proceedings., vol. 2, 2003.

[45] H. Al-rashaideh, “Preprocessing phase for Arabic Word Handwritten Recognition,” Inf. Transm. Comput. Networks J., vol. 6, no. 1, pp. 11–19, 2006.

[46] A. M. Zeki, “The segmentation problem in arabic character recognition the state of the art,” in Proceedings of 1st International Conference on Information and Communication Technology, ICICT 2005, 2005, vol. 2005, pp. 11–26.

[47] S. S. Bukhari, F. Shafait, and T. M. Breuel, “High performance layout analysis of Arabic and Urdu document images,” in Proceedings of the International Conference on Document Analysis and Recognition, ICDAR, 2011, pp. 1275–1279.

[48] G. Nagy, S. Seth, and M. Viswanathan, “A prototype document image analysis system for technical journals,” Computer (Long. Beach. Calif)., vol. 25, no. 7, pp. 10–22, 1992.

[49] A. M. Al-shatnawi and K. Omar, “The Thinning Problem in Arabic Text Recognition - A Comprehensive Review,” Int. J. Comput. Appl., vol. 103, no. 3, pp. 35–42, 2014.

[50] M. T. Parvez and S. a. Mahmoud, “Offline arabic handwritten text recognition: A Survey,” ACM Comput. Surv., vol. 45, no. 2, p. 23:1–23:35, 2013.

[51] Z. Guo and R. W. Hall, “Fast fully parallel thinning algorithms,” CVGIP Image Underst., vol. 55, no. 3, pp. 317–328, 1992.

[52] C. Hilditch, “Comparison of thinning algorithms on a parallel processor,” Image Vis. Comput., vol. 1, no. 3, pp. 115–132, 1983.

[53] P. S. P. Wang and Y. Y. Zhang, “A Fast and Flexible Thinning Algorithm,” IEEE Trans. Comput., vol. 38, no. 5, pp. 741–745, 1989.

[54] S. A. Mahmoud, I. AbuHaiba, and R. J. Green, “Skeletonization of Arabic characters using clustering based skeletonization algorithm (CBSA),” Pattern Recognit., vol. 24, no. 5, pp. 453–464, 1991.

[55] M. Melhi, S. S. Ipson, and W. Booth, “A novel triangulation procedure for thinning hand-written text,” Pattern Recognit. Lett., vol. 22, no. 10, pp. 1059–1071, 2001.

[56] J. Cowell and F. Hussain, “Thinning Arabic characters for feature extraction,” in Proceedings of the International Conference on Information Visualisation, 2001, vol. 2001–Janua, pp. 181–185.

[57] M. Altuwaijri and M. Bayoumi, “A new thinning algorithm for Arabic characters using self-organizing neural network,” in Circuits and Systems, 1995. ISCAS’95., 1995 IEEE International Symposium on, 1995, vol. 3, pp. 1824–1827.

[58] M. A. Alghamdi and W. J. Teahan, “A New Thinning Algorithm for Arabic Script,” Int. J. Comput. Sci. Inf. Secur. (IJCSIS), vol. 15, no. 1, pp. 204–211, 2017.

[59] L. M. Lorigo and V. Govindaraju, “Offline arabic handwriting recognition: A survey,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 28, no. 5. pp. 712–724, 2006.

[60] S. Naz, K. Hayat, M. Imran Razzak, M. Waqas Anwar, S. A. Madani, and S. U. Khan, “The optical character recognition of Urdu-like cursive scripts,” in Pattern Recognition, 2014, vol. 47, no. 3, pp. 1229–1248.

[61] R. Plamondon and S. N. Srihari, “On-line and off-line handwriting recognition: A comprehensive survey,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 22, pp. 63–84, 2000.

[62] M. Amara, K. Zidi, K. Ghedira, and S. Zidi, “New Rules to Enhance the Performances of Histogram Projection for Segmenting Small-Sized Arabic Words,” in Hybrid Intelligent Systems, Springer, 2016, pp. 167–176.

[63] S. N. Nawaz, M. Sarfraz, A. Zidouri, and W. G. Al-Khatib, “An approach to offline Arabic character recognition using neural networks,” in Electronics, Circuits and Systems, 2003. ICECS 2003. Proceedings of the 2003 10th IEEE International Conference on, 2003, vol. 3, p. 1328–1331 Vol.3.

[64] N. Arica and F. T. Yarman-Vural, “An overview of character recognition focused on off-line handwriting,” IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews, vol. 31, no. 2. pp. 216–233, 2001.

[65] J. Alkhateeb, “Word Based Off-line Handwritten Arabic Classification and Recognition,” Ph. D. thesis, School of Computing, Informatics and Media, University of Bradford, 2010.

[66] L. O’Gorman, “The Document Spectrum for Page Layout Analysis,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 15, no. 11, pp. 1162–1173, 1993.

[67] S. N. Srihari and G. Ball, “An assessment of Arabic handwriting recognition technology,” in Guide to OCR for Arabic Scripts, Springer, 2012, pp. 3–34.

[68] A. Amin and G. Masini, “Machine Recognition of Multi Font Printed {Arabic} Texts,” in Proceedings, Eighth International Conference on Pattern Recognition (Paris, France, October 27--31, 1986), 1986, pp. 392–395.

[69] J. F. Mari, “Machine Recognition and Correction of Printed Arabic Text,” IEEE Trans. Syst. Man Cybern., vol. 19, no. 5, pp. 1300–1306, 1989.

[70] A. Ymin and Y. Aoki, “On the segmentation of multi-font printed Uygur scripts,” in Pattern Recognition, 1996., Proceedings of the 13th International Conference on, 1996, vol. 3, pp. 215–219.

[71] A. Cheung, M. Bennamoun, and N. W. Bergmann, “Arabic optical character recognition system using recognition-based segmentation,” Pattern Recognit., vol. 34, no. 2, pp. 215–233, 2001.

[72] M. S. Khorsheed, “Offline recognition of omnifont Arabic text using the HMM ToolKit (HTK),” Pattern Recognit. Lett., vol. 28, no. 12, pp. 1563–1571, 2007.

[73] A. Amin, “Off-line Arabic character recognition: the state of the art,” Pattern Recognit., vol. 31, no. 5, pp. 517–530, 1998.

[74] H. Al-Yousefi and S. S. Udpa, “Recognition of arabic characters,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 14, no. 8, pp. 853–857, 1992.

[75] I. S. I. Abuhaiba, S. A. Mahmoud, and R. J. Green, “Recognition of handwritten cursive Arabic characters,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 16, no. 6, pp. 664–672, 1994.

[76] I. Supriana and A. Nasution, “Arabic Character Recognition System Development,” Procedia Technol., vol. 11, no. Iceei, pp. 334–341, 2013.

[77] J. H. AlKhateeb, J. Jiang, J. Ren, and S. Ipson, “Component-based Segmentation of words from handwritten Arabic text,” Int. J. Comput. Syst. Sci. Eng., vol. 5, no. 1, pp. 54–58, 2009.

[78] P. Xiu, L. Peng, X. Ding, and H. Wang, “Offline handwritten Arabic character segmentation with probabilistic model,” in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2006, vol. 3872 LNCS, pp. 402–412.

[79] E. J. Erlandson, J. M. Trenkle, and R. C. Vogt, “Word-level recognition of multifont Arabic text using a feature vector matching approach,” in Document Recognition III, 1996, vol. 2660, pp. 63–71.

[80] A. Choudhary, “A review of various character segmentation techniques for cursive handwritten words recognition,” Int. J. Inf. Comput. Technol, vol. 4, no. 6, pp. 559–564, 2014.

[81] S. Naz, A. I. Umar, S. H. Shirazi, S. B. Ahmed, M. I. Razzak, and I. Siddiqi, “Segmentation techniques for recognition of Arabic-like scripts: A comprehensive survey,” Educ. Inf. Technol., pp. 1–17, 2015.

[82] B. A. Najoua and E. Noureddine, “A robust approach for Arabic printed character segmentation,” in Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on, 1995, vol. 2, pp. 865–868.

[83] B. Parhami and M. Taraghi, “Automatic recognition of printed Farsi texts,” Pattern Recognit., vol. 14, no. 1, pp. 395–403, 1981.

[84] L. Zheng, A. H. Hassin, and X. Tang, “A new algorithm for machine printed Arabic character segmentation,” Pattern Recognit. Lett., vol. 25, no. 15, pp. 1723–1729, 2004.

[85] L. Lorigo and V. Govindaraju, “Segmentation and pre-recognition of arabic handwriting,” in Proceedings of the International Conference on Document Analysis and Recognition, ICDAR, 2005, vol. 2005, pp. 605–609.

[86] D. Koteswara Rao and A. Negi, “Advances in Signal Processing and Intelligent Recognition Systems: Proceedings of Second International Symposium on Signal Processing and Intelligent Recognition Systems (SIRS-2015) December 16-19, 2015, Trivandrum, India,” M. S. Thampi, S. Bandyopadhyay, S. Krishnan, K.-C. Li, S. Mosin, and M. Ma, Eds. Cham: Springer International Publishing, 2016, pp. 633–644.

[87] A. M. Zeki, M. S. Zakaria, and C.-Y. Liong, “Segmentation of Arabic Characters: A Comprehensive Survey,” Int. J. Technol. Diffus., vol. 2, no. 4, pp. 48–82, 2011.

[88] A. Cheung, M. Bennamoun, and N. W. Bergmann, “An Arabic optical character recognition system using recognition-based segmentation,” Pattern Recognit., vol. 34, no. 2, pp. 215–233, 2001.

[89] C. H. Chen and J. L. DeCurtins, “Word recognition in a segmentation-free approach to OCR,” Proc. 2nd Int. Conf. Doc. Anal. Recognit. ICDAR 93, pp. 573–576, 1993.

[90] B. A. Srinivas, A. Agarwal, and C. R. Rao, “An Overview of OCR Research in Indian Scripts,” vol. 2, no. 2, 2008.

[91] Ø. Due Trier, A. K. Jain, and T. Taxt, “Feature extraction methods for character recognition-A survey,” Pattern Recognit., vol. 29, no. 4, pp. 641–662, 1996.

[92] E. Moubtahij, A. H. Hicham, and K. SATORI, “Review of Feature Extraction Techniques for Offline Handwriting Arabic Text Recognition,” Int. J. Adv. Eng. Technol., vol. 7, no. 1, pp. 50–58, 2014.

[93] D. V. Sharma, G. Saini, and M. Joshi, “Statistical Feature Extraction Methods for Isolated Handwritten Gurumukhi Script,” vol. 2, no. 4, pp. 380–384, 2012.

[94] H. Al-Muhtaseb and R. Qahwaji, “Arabic Optical Character Recognition: Recent Trends and Future Directions,” in Applied Signal and Image Processing: Multidisciplinary Advancements, IGI Global, 2011, pp. 324–346.

[95] M. Abd, S. Al Rubeaai, and G. Paschos, “Hybrid Features for an Arabic Word Recognition System,” Davidpublishing.Com, vol. 3, no. 1, pp. 685–691, 2012.

[96] R. Saabni, "Efficient recognition of machine printed Arabic text using partial segmentation and Hausdorff distance," 6th Int. Conf. Soft Comput. Pattern Recognition, SoCPaR 2014, pp. 284–289, 2015.

[97] a. M. Elgammal and M. a. Ismail, "A graph-based segmentation and feature extraction framework for\nArabic text recognition," Proc. Sixth Int. Conf. Doc. Anal. Recognit., 2001.

[98] M. S. Khorsheed and W. F. Clocksin, "Structural Features of Cursive Arabic Script.," in BMVC, 1999, pp. 1–10.

[99] I. Ahmad, L. Rothacker, G. A. Fink, and S. A. Mahmoud, "Novel sub-character HMM models for arabic text recognition," Proc. Int. Conf. Doc. Anal. Recognition, ICDAR, pp. 658–662, 2013.

[100] D. Ghosh, T. Dube, and A. Shivaprasad, "Script Recognition-a review," IEEE Trans. Pattern Anal. Mach. Intell., vol. 32, no. 12, pp. 2142–2161, 2010.

[101] H. Almohri, J. S. Gray, and H. Alnajjar, A real-time DSP-based optical character recognition system for isolated arabic characters using the TI TMS320C6416T. University of Hartford, 2007.

[102] M. Z. Hossain, "Rapid Feature Extraction for Optical Character Recognition," pp. 1–5, 2012.

[103] G. Kumar and P. K. Bhatia, "A detailed review of feature extraction in image processing systems," in International Conference on Advanced Computing and Communication Technologies, ACCT, 2014, pp. 5–12.

[104] H. Aboaisha, Z. Xu, and I. El-Feghi, "An investigation on efficient feature extraction approaches for Arabic letter recognition," 2012.

[105] M. Kef, L. Chergui, and S. Chikhi, "Comparative study of the use of geometrical moments for Arabic handwriting recognition," in Sciences of Electronics, Technologies of Information and Telecommunications (SETIT), 2012 6th International Conference on, 2012, pp. 303–308.

[106] M. A. Abd and G. Paschos, "Effective arabic character recognition using support vector machines," Innov. Adv. Tech. Comput. Inf. Sci. Eng., pp. 7–11, 2007.

[107] M. Oujaoura, R. El Ayachi, M. Fakir, B. Bouikhalene, and B. Minaoui, "Zernike moments and neural networks for recognition of isolated Arabic characters," Int. J. Comput. Eng. Sci., vol. 2, pp. 17–25, 2012.

[108] Z. Shaaban, "A new recognition scheme for machine-printed Arabic texts based on neural networks," in Proceedings of World Academy of Science, Engineering and Technology, 2008, vol. 31, pp. 25–27.

[109] I. A. Elrube, M. T. El Sonni, and S. S. Saleh, "Printed Arabic sub-word recognition using moments," World Acad. Sci. Eng. Technol., vol. 66, pp. 737–741, 2010.

[110] P. B. Pati and A. G. Ramakrishnan, "OCR in Indian scripts: A survey," IETE Tech. Rev., vol. 22, no. 3, pp. 217–227, 2005.

[111] D. Zhang and G. Lu, "A Comparative Study on Shape Retrieval Using Fourier Descriptors with Different Shape Signatures," Int. Conf. Intell. Multimed. Distance Educ., vol. 1, pp. 1–9, 2001.

[112] N. Ben Amor and N. E. Ben Amara, "Multifont Arabic Characters Recognition Using HoughTransform and HMM/ANN Classification," J. Multimed., vol. 1, no. 2, pp. 50–54, 2006.

[113] S. Touj, N. E. Ben Amara, and H. Amiri, "Generalized Hough Transform for Arabic Printed Optical Character Recognition.," Int. Arab J. Inf. Technol., vol. 2, no. 4, pp. 326–333, 2005.

[114] N. Ben Amor and N. E. Ben Amara, A Novel Method for Multifont Arabic Characters Features Extraction. INTECH Open Access Publisher, 2012.

[115] R. Mehran, H. Pirsiavash, and F. Razzazi, "A front-end OCR for omni-font Persian/Arabic cursive printed documents," in Digital Image Computing: Techniques and Applications, 2005. DICTA'05. Proceedings 2005, 2005, p. 56.

[116] M. Amara, K. Zidi, S. Zidi, and K. Ghedira, "Arabic Character Recognition Based M-SVM," in International Conference on Advanced Machine Learning Technologies and Applications, 2014, pp. 18–25.

[117] M. Shafii, "Optical Character Recognition of Printed Persian/Arabic Documents," 2014.

[118] S. Arora, D. Bhattacharjee, M. Nasipuri, L. Malik, M. Kundu, and D. K. Basu, "Performance Comparison of SVM and ANN for Handwritten Devnagari Character Recognition," Int. J. Comput. Sci. Issues, vol. 7, no. 3, pp. 1–10, 2010.

[119] M. S. Khorsheed, "Recognizing Cursive Typewritten Text Using Segmentation-Free System," vol. 2015, 2015.

[120] S. M. Awaida and M. S. Khorsheed, "Developing discrete density Hidden Markov Models for Arabic printed text recognition," Proceeding - 2012 IEEE Int. Conf. Comput. Intell. Cybern. Cybern. 2012, pp. 35–39, 2012.

[121] F. Slimane, S. Kanoun, J. Hennebert, A. M. Alimi, and R. Ingold, "A study on font-family and font-size recognition applied to Arabic word images at ultra-low resolution," Pattern Recognit. Lett., vol. 34, no. 2, pp. 209–218, 2013.

[122] H. A. Al-Muhtaseb, S. A. Mahmoud, and R. S. Qahwaji, "Recognition of off-line printed Arabic text using Hidden Markov Models," Signal Processing, vol. 88, no. 12, pp. 2902–2912, 2008.

[123] I. A. Doush and A. M. Al Trad, "Improving post-processing optical character recognition documents with Arabic language using spelling error detection and correction," Int. J. Reason. Intell. Syst., vol. 8, no. 3/4, p. 91, 2016.

[124] W. Magdy and K. Darwish, "Effect of OCR error correction on Arabic retrieval," Inf. Retr. Boston., vol. 11, no. 5, pp. 405–425, 2008.

[125] K. Taghva and E. Stofsky, "OCRSpell: An interactive spelling correction system for OCR errors in text," Int. J. Doc. Anal. Recognit., vol. 3, no. 3, pp. 125–137, 2001.

[126] A. H. Hassin, X.-L. Tang, J.-F. Liu, and W. Zhao, "Printed Arabic character recognition using HMM," J. Comput. Sci. Technol., vol. 19, no. 4, pp. 538–543, 2004.

[127] I. Aljarrah, O. Al-Khaleel, K. Mhaidat, M. Alrefai, A. Alzu'Bi, and M. Rabab'Ah, "Automated system for arabic optical character recognition with lookup dictionary," J. Emerg. Technol. Web Intell., vol. 4, no. 4, pp. 362–370, 2012.

[128] P. Damien, N. Wakim, and M. Eg??a, "Phoneme-viseme mapping for modern, classical arabic language," in 2009 International Conference on Advances in Computational Tools for Engineering Applications, ACTEA 2009, 2009, pp. 547–552.

[129] R. Prasad, S. Saleem, M. Kamali, R. Meermeier, and P. Natarajan, "Improvements in hidden Markov model based Arabic OCR," Proc. 19th Int. Conf. Pattern Recognit., pp. 1–4, 2008.

[130] P. Natarajan, S. Saleem, R. Prasad, E. MacRostie, and K. Subramanian, "Multi-lingual Offline Handwriting Recognition Using Hidden Markov Models: A Script-Independent Approach," in Arabic and Chinese Handwriting Recognition, 2008, pp. 231–250.

[131] Y. Bassil and M. Alwani, "OCR Post-Processing Error Correction Algorithm Using Google ' s Online Spelling Suggestion," J. Emerg. Trends Comput. Inf. Sci., vol. 3, no. 1, pp. 90–99, 2012.

[132] A. G. Krayem, "A high level approach to Arabic sentence recognition," Nottingham Trent University, 2013.

[133] S. V Rice, "Measuring the accuracy of page-reading systems," University of Nevada, Las Vegas, 1996.

[134] S. Mihov, K. U. Schulz, C. Ringlstetter, V. Dojchinova, V. Nakova, K. Kalpakchieva, O. Gerasimov, A. Gotscharek, and C. Gercke, "A corpus for comparative evaluation of OCR software and postcorrection techniques," Proc. Int. Conf. Doc. Anal. Recognition, ICDAR, vol. 2005, pp. 162–166, 2005.

[135] T. Kanungo, G. A. Marton, and O. Bulbul, "OmniPage vs. Sakhr: Paired model evaluation of two Arabic OCR products," in Electronic Imaging'99, 1999, pp. 109–120.

[136] T. Kanungo, G. a Marton, and O. Bulbul, "Performance evaluation of two Arabic OCR products," Proc. SPIE, pp. 76–83, 1999.

[137] K. Jung, K. I. Kim, and A. K. Jain, "Text information extraction in images and video: A survey," Pattern Recognit., vol. 37, no. 5, pp. 977–997, 2004.

[138] I. Ahmad, S. A. Mahmoud, and G. A. Fink, "Open-vocabulary recognition of machine-printed Arabic text using hidden Markov models," Pattern Recognit., vol. 51, pp. 97–111, 2016.

[139] M. A. Alghamdi, I. S. Alkhazi, and W. J. Teahan, "Arabic OCR evaluation tool," in 2016 7th International Conference on Computer Science and Information Technology (CSIT), 2016, pp. 1–6.