Pattern Recognition Letters 34 (2013) 209–218

Contents lists available at SciVerse ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

A study on font-family and font-size recognition applied to Arabic word images at ultra-low resolution

Fouad Slimane [a,b,*], Slim Kanoun [c], Jean Hennebert [a], Adel M. Alimi [b], Rolf Ingold [a]

[a] DIVA Group, Department of Informatics, University of Fribourg, Bd. de Perolles 90, 1700 Fribourg, Switzerland
[b] REGIM Lab., National School of Engineers (ENIS), University of Sfax, BP 1173, Sfax 3038, Tunisia
[c] National School of Engineers (ENIS), University of Sfax, BP 1173, Sfax 3038, Tunisia

ARTICLE INFO

Article history:
Received 5 March 2012
Available online 25 September 2012

Communicated by G. Borgefors

Keywords:
Font/size recognition
GMM
HMM
Arabic OCR
Sliding window
Ultra-low resolution

ABSTRACT

In this paper, we propose a new font and size identification method for ultra-low resolution Arabic word images using a stochastic approach. The literature has proved the difficulty for Arabic text recognition systems to treat multi-font and multi-size word images. This is due to the variability induced by some font family, in addition to the inherent difficulties of Arabic writing including cursive representation, overlaps and ligatures. This research work proposes an efficient stochastic approach to tackle the problem of font and size recognition. Our method treats a word image with a fixed-length, overlapping sliding window. Each window is represented with a 102 features whose distribution is captured by Gaussian Mixture Models (GMMs). We present three systems: (1) a font recognition system, (2) a size recognition system and (3) a font and size recognition system. We demonstrate the importance of font identification before recognizing the word images with two multi-font Arabic OCRs (cascading and global). The cascading system is about 23% better than the global multi-font system in terms of word recognition rate on the Arabic Printed Text Image (APTI) database which is freely available to the scientific community.

1. Introduction

This work focuses on the recognition of ultra-low resolution Arabic text (<100 dpi), typically obtained with screen rendered images. Recognizing Arabic text with OCR is a challenging task which has to cope with several difficulties. A screen-rendered text is on ultra-low resolution and is generally anti-aliased to make it look better to the human eye. The same character of the same logical description (font, size, etc.) is often rendered differently within the same document depending on its position. Generally, the appearance of screen-rendered text depends on the used font, size, background, position, operating system, application and used anti-aliasing algorithm (Wachenfeld et al., 2006). Therefore, all these difficulties added to a multi-font, multi-size and multi-style text make the development of a recognition system complex.

Moreover, the Arabic script is represented by a cursive script for printed as well as handwritten text and is composed of inter-related characters written from right to the left. Some of these characters change their shapes according to the place where they occur in the word. Most of them have four shapes: isolated, initial, medial and final. More than half of the Arabic letters may include dots in their shapes. The number of dots is one, two or three and they can occur above or below the letter body, but never simultaneously bellow and above. The presence of these dots and their position allow us to differentiate between letters that belong to the same shape family. Some Arabic letters include a "loop" character, generally called an "occlusion shape" which differs from one character to another (Kanoun et al., 2011) (see Fig. 1(a)). Thus, the Arabic script recognition is a very complex task.

The domain of Arabic text recognition can be segmented into printed text (AL-Shatnawi et al., 2011) and handwritten text (Lorigo et al., 2006; AlKhateeb et al., 2011). Handwritten Arabic text images can be acquired off-line, typically from a scanner or camera, or on-line with a graphical tablet or touch-screen. The cursive nature of both handwritten and printed Arabic texts adds an inherent difficulty to segment into characters. For this reason, most of the recognition methods have converged into state-based statistical models able to segment and recognize characters at the same time, typically using Hidden Markov Models (Khorsheed, 2007; Al-Muhtaseb et al., 2008; Slimane et al., 2008). Other approaches have been proposed such as, template matching that has reported decent performances on easier to segment on-line Arabic handwriting (Sternby et al., 2009).

In spite of the numerous research works concerning Arabic text recognition, the obtained performances are still mostly

* Corresponding author at: DIVA Group, Department of Informatics, University of Fribourg, Bd. de Perolles 90, 1700 Fribourg, Switzerland. Fax: +41 26 300 97 26.

E-mail addresses: fouad.slimane@unifr.ch (F. Slimane), slim.kanoun@gmail.com (S. Kanoun), jean.hennebert@unifr.ch (J. Hennebert), Adel.Alimi@ieee.org (A.M. Alimi), rolf.ingold@unifr.ch (R. Ingold).

0167-8655/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.patrec.2012.09.012

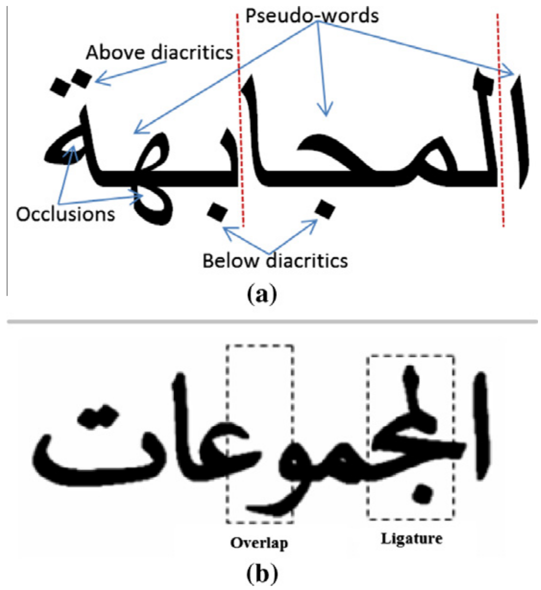

Fig. 1. An example of Arabic words: (a) Dots, occlusions and pseudo-words, (b) the word "Groups" written with a vertical ligature and a vertical overlap.

unsatisfactory. Moreover, commercial systems are not yet available for large vocabulary applications. The encountered difficulties are similar to those faced for handwritten Latin script, but often more complex due to the diversity of Arabic character shapes, the short bindings between successive characters, the lengthening of the horizontal bindings and the presence of vertical bindings. To our knowledge, commercialized Arabic OCR systems are limited to recognize a reduced number of fonts in the context of cleanly acquired inputs. For more details concerning Arabic script recognition, a good survey is presented in (Lorigo et al., 2006). On the basis of the difficulties in Arabic OCR and the great number of Arabic fonts (Ben Amara et al., 2004), several research works (Slimane et al., 2010; Jung et al., 1999; Ben Amara et al., 2003; Bapst et al., 1998) focused on the subdivision of printed Arabic word recognition in two complementary steps: the font identification and the mono-font word recognition.

While previous research studies were focused on high-resolution images typically acquired with scanners, this paper addresses a low-resolution scenario. So far, no work has been published about font and size recognition using low resolution Arabic documents. However, low resolution Arabic text images are becoming more frequent, either directly embedded in web sites as menus or decorations, or resulting from screen capture tools on rendered documents. Low resolution text images can also be acquired when taking a picture of a document with a mobile phone camera. However, such images present further variabilities such as distortion or illumination problems that are out of the scope of this paper.

In this paper, we propose a new robust method, in the framework of the a priori approach, for font and size recognition of Arabic word images at ultra-low resolution. The method is based on word image modelling with a sequence of typographical features extracted from a sliding window. This method presents many advantages. First, no a priori segmentation into characters/fragments of characters is needed, which is an important feature for Arabic text where characters are tied to each other and difficult to separate. Second, we use versatile and powerful GMMs that are able to finely model the distributions of our features, quite wide-scoped and largely multi-dimensional. Thirdly, we can use the same features for the different recognition systems.

Optical font recognition (OFR) is useful and necessary, especially in the following domains (Ben Amara et al., 2004):

- Text font knowledge may improve the recognition rate of OCR systems, because we believe that mono-font OCR may give more accurate results than omni-font OCR.
- Document indexing and information retrieval, where word indexes are generally printed in fonts different from those of the running text.
- Text style recognition for document reproduction, where knowledge of the font is necessary in order to reproduce the document.
- Recognition of logical document structures, where knowledge of the font used in a word, line, or text block may be useful for defining its logical label (chapter title, section title or paragraph).

The remainder of this paper is organized as follows. Section 2 introduces some complexities of the Arabic script. Section 3 presents some related work. In Section 4, we describe our different systems for font and size recognition. Finally, Section 5 is dedicated to the word image database used for the evaluation and the recognition results, and it is followed by a conclusion.

2. Characteristics of Arabic script

With a quite large user base of about 300 million people worldwide (Cheriet et al., 2006; Zhu et al., 2001), Arabic is very important in the culture of many people. Compared to printed Latin script, we can underline several important differences:

- Arabic is written from right to left.
- 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.

2.1. Arabic alphabet

The Arabic alphabet is richer than its Latin equivalent. It contains 28 letters (15 with dots and 13 without), most of which change shape according to their appearance at the beginning, middle or end of the word (see Table 1). Six letters have just two modes of appearance and cannot be connected to a following letter, i.e. their "begin" shapes correspond to their "isolated" shapes and their "middle" shapes correspond to their "end" shapes. Taking into account positioning, the number of different shapes rises to more than 100. All of them do not have a fixed character width (all character shapes do not have the same height and width) and also no fixed size.

In addition to this "positioning" variability, there are extra characters appearing as variations of some basic characters. The "ة (TaaaClosed)" is the same character "ت (Taaa)", but it is used just at the end of Arabic names; it cannot be used in verbs. Other characters are created by the combination of "ء (Hamza)" and "ا (Alif)" or "ء (Hamza)" and "و (Waaw)". They are almost pronounced the same way but their use depends on their position in the word. Taking into account these extra characters, there are overall 120 different shapes.

2.2. Calligraphic features

The Arabic word can be made up of one or more components (pseudo-word or PAW for Piece of Arabic Word) and the characters of the same connected component can be ligatured horizontally and vertically. The ligature is also dependent on the used font (in some fonts, up to four vertical ligatures are generated) (see

Table 1
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

Fig. 2). Fig. 1(b) shows a vertical ligature of three characters Alif, Miim and Jiim. Finally, vertical overlaps can occur at the intersection of pseudo-words and also within words for some sequence of characters.

To our knowledge, there are over 450 Arabic fonts (Ben Amara et al., 2004), all of which are used somewhere in the Muslim world. Fig. 2 shows ten of the mostly used fonts. They are also those we used in our experiments. Generated from the same input text, the lines of Fig. 2 illustrate well the varied complexity of shapes depending on the font. Some fonts are simpler with no or few overlaps and ligatures (Andalus, Tahoma). Some other fonts are more complex, richer in overlaps, ligatures and flourishes (Diwani Letter).

3. Review of font and size recognition approaches

A review of the literature shows that, during the last two decades, several works have focused on font recognition for many scripts: Arabic (Abuhaiba et al., 2005), Chinese (Yang et al., 2006), Cyrillic (Ma and Doermann, 2005), English (Ramanathan et al., 2009), French (Zramdini and Ingold, 1998), Greek, Hangul, Kannada, Korean (Einsele-Aazami, 2008), Spanish, Thai, etc. Printed script recognition is far from being solved and still remains an open research subject for many languages. Font recognition is principally treated with two different approaches: the a priori approach (also termed independent content approach or global approach) and the a posteriori approach (also termed dependent content approach or local approach) (Zramdini and Ingold, 1998).

The a priori font recognition approach consists of identifying the text font without any knowledge of the characters that appear in the text entity (text block, text line or word image). This approach is based on global features and does not require connected components analysis or segmentation techniques. Systems based on this approach differ by the text entity level analysis and the features used. There are principally three classes. The first class is based on the characterization of text block images by texture features, e.g., based on Gabor filters (Zhu et al., 2001; Ma and Doermann, 2005; Ramanathan et al., 2009), high order statistical moments (Avilés-Cruz et al., 2004; Avilés-Cruz et al., 2005; Avilé-Cruz et al., 2006) and Hilbert–Huang transforms (Yang et al., 2006; Mallikarjunaswamy and Karunakara, 2010). The second class is based on the characterization of text line or word images with typographical features, e.g., based on textons theory (Schreyer et al., 1999), vertical projection profile heights, density of black pixels, variance of horizontal projection profile derivative, text density, letter size, orientation and spacing, etc (Zramdini and Ingold, 1993; Zramdini and Ingold, 1998; Kim et al., 2004). The third class is based on clusters of word images (Khoubyari and Hull, 1996) or on stroke pattern analysis of decomposed word images using wavelets (Zhang et al., 2004).

The a posteriori font recognition approach consists of recognizing the font of a text using the knowledge of characters appearing in the text. This approach can therefore use features based on local properties of individual letters. Indeed, the letter shape depends on the font family (Times, Helvetica, etc.) and style (roman, italic, bold), such as the letters "g" and "g", "a" and "*a*", etc. Systems based on this approach differ in the features used for character description. There are mainly four classes. The first class is based on texture features such as Gabor filters (Ha et al., 2005), or wavelets (Fu et al., 2006; Ding et al., 2007). The second class is based on typographical features such as density of black pixels, projection profile codes, skeleton templates, stroke templates, linear interpolation analysis, etc. (Lin et al., 2001; Tsai et al., 2001; Kim et al., 2002; Sun et al., 2006; Jamjuntr and Dejdumrong, 2009). The third class is based on statistical features such as bitmaps, zoning features, Fourier descriptors, Zernike moments, Discrete Cosine Transform coefficients, Fourier coefficients, Karhunen–Loeve features or Eigencharacters, contour profiles, geometric invariant moments, projection histograms, etc. (Ozturk et al., 2001). The fourth class is based on an automatic extraction of spatial local features using non-negative matrix factorization (Lee et al., 2003). In (Jeong et al., 2003), the authors propose a system based on the hybridization of the a priori approach (at the word image level) and a posteriori approach.

For Arabic font and script recognition, systems based on the a priori approach can be subdivided into two categories: segmentation-free systems and segmentation-based systems. Considering the difficulties in Arabic script segmentation and recognition discussed before, most systems are segmentation-free and employ the a priori approach at text block level with texture features based on Gabor filters (Borji and Hamidi, 2007), wavelets (Imani et al., 2011), fractal geometry (Ben Moussa et al., 2006; Ben Moussa et al., 2010), wavelets and fractal geometry (Zaghden et al., 2006), Gray Level Co-Occurrence Matrix (Bataineh et al., 2011; Bataineh et al., 2012) and Scale Invariant Feature Transform (Zahe-

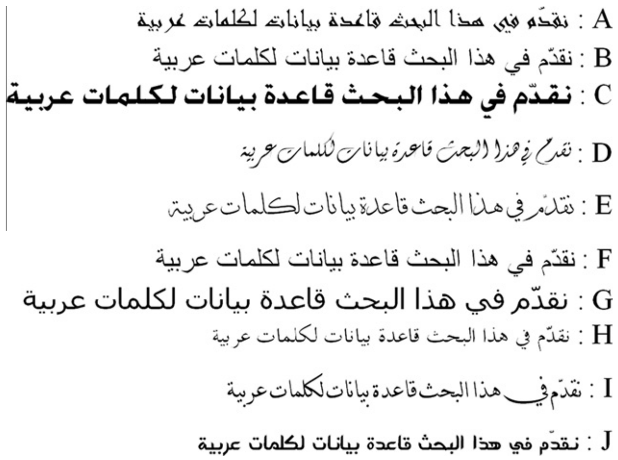

Fig. 2. Fonts used to generate the APTI database: (A) Andalus, (B) Arabic Transparent, (C) Advertising Bold, (D) Diwani Letter, (E) DecoType Thuluth, (F) Simplified Arabic, (G) Tahoma, (H) Traditional Arabic, (I) DecoType Naskh, (J) M Unicode Sara.

di and Eslami, 2011). In (Khosravi and Kabira, 2010), the authors describe a system at text line level using Sobel–Roberts features. In (Abuhaiba et al., 2005), the authors introduce a system at word level using statistical features based on horizontal and vertical projection profiles, Walsh coefficients, invariant moments, geometrical features, etc. In (Ben Amara et al., 2004), the authors propose a system at pseudo-word level using wavelet features. Only one system is segmentation-based (Abuhaiba, 2003). Finally, only one system is based on the a posteriori approach which performs font recognition of isolated characters (Chaker et al., 2010). This last system is based on the dissimilarity index computed on the polygonal approximation of the character.

The detailed study of research works concerning the a priori font recognition approach shows that the systems which use texture features at the level of text block images are robust to noisy and low resolution text image. These systems identify the font for more than one script (e.g., Latin, Greek and Cyrillic) and languages (e.g., Chinese and English) (Zhu et al., 2001). On the other hand, these systems are reliable only for uniform and homogeneous text block images (all words should have the same font). Moreover, the systems which use typographical features at the level of text lines and words are reliable only for noise-free and high resolution text images.

Concerning the works related to the a posteriori font recognition approach, they are based on feature analysis of individual characters. They do not work well when characters are inter-connected such as in Arabic or Cyrillic scripts. In (Kim et al., 2004), the authors consider that, for line-level font recognition, it is assumed that all text lines of a document are written in the same font. This is relevant for postal automation applications that handle international mail, where every line of an address is usually written in the same language, typeface, size and style. However, a text line in a document may contain words with different fonts and scripts (e.g., multi-script text). As an example, a word in Korean can be followed by a word in English in the same line. In addition, most OCR systems maintain words in the document to apply a lexicon-driven semantic analysis of the character recognition results. Document indexing and retrieval are also performed with respect to words since the keyword in a user query or a thesaurus is matched against every word in the document image. Therefore, font recognition at word level is necessary in these cases. Yet most OFR systems have different performances on different text entities (words, lines, paragraphs). They are also dependent on document content and language (Zramdini and Ingold, 1993).

Throughout the state of art, it appears that almost all studies deal with resolutions greater than 100 dpi. In this paper, our objective is to recognize fonts at an ultra-low resolution (down-sampled to 72 dpi) and synthetic Arabic word image. As stated above, the a posteriori approach is not well-suited for Arabic font recognition because the word segmentation into characters is very difficult for complex fonts. To our knowledge, no segmentation technique could perfectly segment Arabic omni-font words. Thus, the a priori approach without word segmentation is considered to be more appropriate for font recognition in Arabic word images. Considering that the larger the text entity is, the better texture features perform, we confirm that the presented features in the previous section are inappropriate for font recognition of Arabic word image.

Some authors have proposed to improve the overall OCR performance by performing a stroke width estimation or font-size recognition. Such information can be used to ease the segmentation into character, to estimate the baseline or to use specific models for the recognized font-size. Besides, font-size recognition can also be used when regenerating the recognized text into a word processor for edition purposes. The most frequently used method to determine the stroke width are based on the projection profiles and the distribution of run lengths (Mehran et al., 2005; Omidyeganeh et al., 2005). However, it has been noted that the performances of stroke width estimation using projection profiles are degrading when the input images are skewed (Shirali-Shahreza et al., 2006). Some authors are proposing to estimated the font-size using only the detected dots in the text image, making the system less sensitive to skew (Shirali-Shahreza et al., 2006). The disadvantage of this last approach is the need to binarize the images. Also, the font-size recognition is arguably less precise using only the dots instead of using full character information.

Another method is presented in (Bushofa and Spann, 1997) where the stroke width is estimated using line and character contour information.

Table 2 summarizes some recognition techniques and recognition rates of some OFR systems. Note that since we do not use the same database for evaluation, most of the performance results reported in the literature cannot be directly compared with our OFR system.

Most of the performance results reported either in the literature or by manufacturers of OFR systems are derived from test datasets that are font-specific, small or based on high quality images.

Regarding the classification, many models have been used in the literature such as weighted Euclidean distance (Zhu et al., 2001; Yang et al., 2006; Khoubyari and Hull, 1996; Ha et al., 2005), decision trees (Abuhaiba et al., 2005) and support vector machines (SVM) (Ramanathan et al., 2009; Imani et al., 2011). These different approaches present all advantages and disadvantages that have conducted their choice considering the specificities of the datasets. For example, Euclidean distance based decision systems are known to perform well in the case of small datasets. SVMs are known for their ability to build discriminant models that are robust if the training and testing conditions remains similar. In our case, we benefited of the availability of APTI, a very large database. It encouraged us to use a generative approach based on GMM models that are able to model in a precise way class-conditional probability density functions provided that the datasets are large. The results reported in the papers should be interpreted as reference results for the freely available APTI database. We invite researchers to test their systems and use this database as a benchmarking database for the recognition of Arabic font-families, font-sizes or word images at ultra-low resolution.

4. Proposed approach

In this article, we propose three systems: a font recognition system, a size recognition system and a hybrid system for font and size recognition. As presented in Fig. 3, they work in two steps: (1) training, (2) [font/size/font and size] recognition. All of them share the same architecture and the same feature extraction.

The proposed technique is based on the sliding window for feature extraction. The obtained feature vectors are used to train the [font/size/font and size] GMM models using the expectation maximization (EM) algorithm. The recognition is performed through a simple score comparison of the trained GMM models. The CPU cost of our approach increases linearly with the number of fonts and font-sizes, and increases also linearly with the width of the word.

4.1. Feature extraction

All word images in our dataset are in gray level. Some features are extracted from the gray level images, others require a binarization. The used binarization technique is the fixed threshold method. Each word image is normalized to a size of 30 pixels height and then transformed into a sequence of feature vectors computed from a narrow analysis window of four pixels sliding from right to left over the word image. In our settings, the uniform analysis window is shifted by one pixel. We performed several tests to determine the optimal size of the window and the normalized height. A feature vector is extracted from each analysis window. As a result, no segmentation into letters is made and the word image is transformed into a sequence of feature vectors where the number of rows corresponds to the number of analysis windows, and the number of columns is equal to the number of components of each feature vector. The feature extraction is divided into two parts. The first part extracts, for each window:

1. Number N_1 of connected black components.
2. Number N_2 of connected white components.
3. Ratio N_1/N_2.
4. Position of the smallest black connected component divided by the height of the window.
5. Sum of perimeter P of all components in the window divided by the perimeter of window P_w.
6. Compactness $(p^2/(4\pi A))$where p is the shape perimeter in the window and A the area.
7. Gravity center of the window, of the right and left half and of the first third, the second and the last part of the window:

$$G_x = \frac{1}{nW}\sum_{i=1}^{n} x_i; \quad G_y = \frac{1}{nH}\sum_{i=1}^{n} y_i$$

where W the width and H the height of the window.

8. Position of the horizontal projection histogram maxima/ height of window.
9. Number of local maxima of the horizontal projection histogram.
10. Number of local maxima of the vertical projection histogram.
11. Density of pixels in the window.
12. Density of pixels in the first quarter, the second, the third and the last part of the window.

The features described above from line 7–12 are computed using the gray level values of the images. They are, therefore, extracting as much as possible of the information available in the low resolution images. The second part of the feature extraction consists of resizing the window into a normalized size of 20 pixels height and computing the horizontal and vertical projection values. These features are also computed using the gray level values of the pixels. For each analysis window, the feature extraction, overall, results in a vector of 51 coefficients.

Once we have a vector sequence, we calculate the so-called *delta* between both vectors using the following formula:

$$\Delta x_n = x_{n+1} - x_{n-1}, \quad \forall 1 < n < T, \text{ with } T \text{ the number of analysis windows}$$

The delta is computed in a similar way as in speech recognition, to include larger contextual information in an analysis window. Including the delta, we obtain $D \times 1$ feature vectors with, in our case $D = 102$.

4.2. Modelling of the likelihoods with GMMs

GMMs are used to model the likelihoods of the features extracted from the image. They are well-known as versatile and flexible modelling tools able to approximate any probability density function. With GMMs, the probability density function $p(x_n|M_f)$ or *likelihood* of a D-dimensional feature vector x_n given the model of a font category M_f, is estimated as a weighted sum of multivariate Gaussian densities:

$$p(x_n|M_f) \cong \sum_{i=1}^{I} w_i \mathcal{N}(x_n, \ \mu_i, \ \Sigma_i)$$

in which I is the number of mixtures and w_i is the weight for mixture i. The Gaussian densities $\mathcal{N}$ are parameterized by a mean $D \times 1$ vector μ_i, and a $D \times D$ covariance matrix, Σ_i. To simplify the computation, we make the hypothesis that the coefficients of the feature vectors are not correlated. The covariance matrix is then simplified to a diagonal matrix. This approximation is classically done when using GMMs and, actually, is somehow compensated by the fact that the probability density functions can still be approximated by a weighted sum of simpler Gaussian densities. Considering the hypothesis of feature vector independence, the log-likelihood of a model M_f for the sequence of feature vectors, $X = \{x_1, \ldots, x_N\}$ is computed as follows:

$$S_f = log\, p(X|M_f) = \sum_{i=1}^{N} log\, p(x_i|M_f)$$

Assuming equal a priori probabilities $P(M_f)$ of each font f, the [font/size/font and size] recognition is performed by selecting the [font/size/font and size] f^* leading to the highest value of S_f. As the local likelihood values $p(x_n|M_f)$ are usually very small, the global likelihood S_f is usually expressed in the log domain to avoid running below machine representation limits.

During training, the expectation-maximization (EM) algorithm (Dempster et al., 1977) is used to iteratively refine the component weights, means and variances to monotonically increase the likelihood of the training feature vectors. In our experiments, we used the EM algorithm to build the models by applying a simple binary splitting procedure to increase the number of Gaussian mixtures through the training procedure. Fig. 4 shows the evolution of the font recognition rate as a function of the number of Gaussians. The Figure shows a saturation of the performance for most fonts between 2048 and 8192 Gaussians. As our objective is here to maximize the recognition performance, we have chosen to use 8192 Gaussians as reference for the font and size recognition parts of our system.

From a practical point of view, GMMs can be seen as one-state Hidden Markov Models. We therefore used the HTK toolkit[1] to implement our modelling scheme. At recognition time, the GMMs

[1] http://htk.eng.cam.ac.uk/.

Table 2
The characteristics and performances of some OFR systems.

References	Languages	Database	No. of styles	No. of sizes	No. of fonts	Resolution	Performances
Abuhaiba et al. (2005)	Arabic	108,000 word images	3	3	3	300	90.8
Ben Moussa et al. (2006)	Arabic	450 block images			10	300	98.0
Ben Moussa et al. (2010)	Arabic	1000 block images		4	10	200	96.6
	Latin	800 block images			8	200	99.3
Zaghden et al. (2006)	Arabic	2500 block images			10	300	96.5
						75	96.0
Bataineh et al. (2012)	Arabic	700 block images			7	96	98.0
Abuhaiba (2003)	Arabic	185,839 word images	4	3	3	300	77.4
Chaker et al. (2010)	Arabic	computer-generated database (size not indicated)			10		100
Yang et al. (2006)	Chinese	100 computer-generated text block images	4		6	72	97.2
Ha et al. (2005)	Chinese	165,400 character images			4		99.1
Fu et al. (2006)	Chinese	150,200 character images			4		99.2
Ding et al. (2007)	Chinese	2741,150 character samples			7		91.3
Lin et al. (2001)	Chinese	31,995 character images			5	300	97.8
Tsai et al. (2001)	Chinese	5401 character images		12	5		97.3
Zhu et al. (2001)	Chinese	14,000 block images	4		6	100	99.1
	English		4		8		
Sun et al. (2006)	Chinese English	675 character images			20 Chinese and 20 English	300	97.0
Ramanathan et al. (2009)	English	216 block images	4		6		93.5
Zramdini and Ingold (1998)	English	100 line images	7	4	10	300	96.9
Kim et al. (2004)	English	168,000 word images	4	3	2	300	97.3
		96,000 word images	4	3	2	300	99.1
Khoubyari and Hull (1996)	English	1000 word images			33	300	85.0
Zhang et al. (2004)	English	22,384 computer-generated word images	4		2		96.1
Khosravi and Kabira (2010)	Farsi	20,000 line images			10	100	94.2
Zahedi and Eslami (2011)	Farsi/Arabic	1400 block images			20	300	≈ 100
Mallikarjunaswamy and Karunakara (2010)	Kannada	120 block images			4		90.6
Jeong et al. (2003)	Korean	7200 Korean word images			2		99.1
Borji and Hamidi (2007)	Persian	computer-generated blocks (size not indicated)	4		7		82.0
Imani et al. (2011)	Persian	5000 block images			10		95.0
Avilés-Cruz et al. (2004)	Spanish	800 window block images	4		8	300	100
						75	97.2
Jamjuntr and Dejdumrong (2009)	Thai	character images (size not indicated)			10		84.0

are fed in parallel with the features extracted from the image. The GMM issuing the highest likelihood score is selected and determines the [font/size/font and size] hypothesis. Performances are evaluated in terms of [font/size/font and size] recognition rates using an unseen set of word images.

5. Evaluation

To evaluate the performance of our recognition systems, experiments have been conducted on some parts of the large APTI (Arabic Printed Text Images) database (Slimane et al., 2009). In all tests, recognition scores have been evaluated at word level.

5.1. APTI database

Available since July 2009, APTI has been freely distributed to the scientific community for benchmarking purposes.[2] At the time of writing this paper, more than 25 research groups all over the world have started using the APTI database. To compare systems developed by different groups, a first competition was held at ICDAR'2011. The goal of that competition was to evaluate the capacity of recognition systems to handle different sizes and fonts using digitally low resolution images for robust, screen-based OCR (Slimane et al., 2011, 2009).

[2] http://diuf.unifr.ch/diva/APTI/.

The APTI database is developed using Arabic words already segmented from text lines. APTI contains a mix of decomposable and non-decomposable word images. The parsing procedure totalled 113,284 distinct Arabic words, leading to a pretty good coverage of the Arabic words used in various texts. For more details about the total number of pseudo-words, word images and characters in APTI, we refer to Slimane et al. (2009).

APTI sizes, fonts and styles are widely used on computer screens, Arabic newspapers and many other documents. The combination of fonts, styles and sizes guarantees a wide variety of images in the database.

In our tests for the different font/size/font and size recognition systems, we used 1000 word images for each font and size. With 10 fonts and 10 font-sizes (6, 7, 8, 9, 10, 12, 14, 16, 18 and 24), 100,000 word images in *plain* style were used in the training phase and an additional 100,000 different word images were used for the test phase. In our tests for the word recognition system, we used 18,897 word images for each font and size in the training phase and an additional 18,868 different word images for each font and size were used for the test phase.

5.2. Experimental results and discussion

In this section, we summarize all results of four recognition experiments:

1. font recognition;
2. font-size recognition;

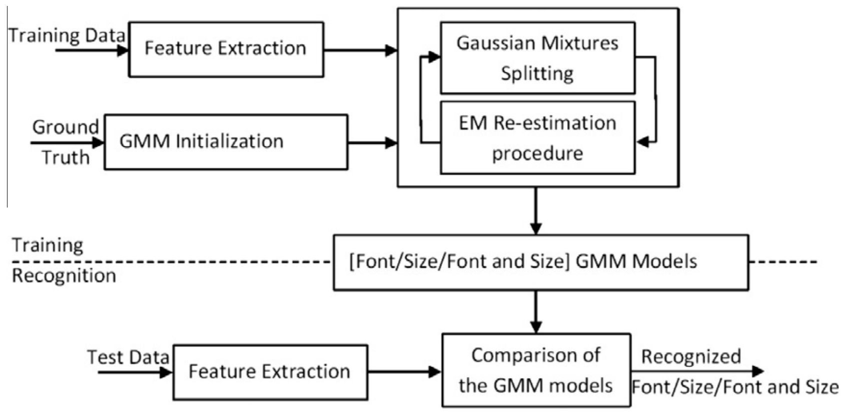

Fig. 3. [Font/size/font and size] recognition steps using Gaussian Mixture Models.

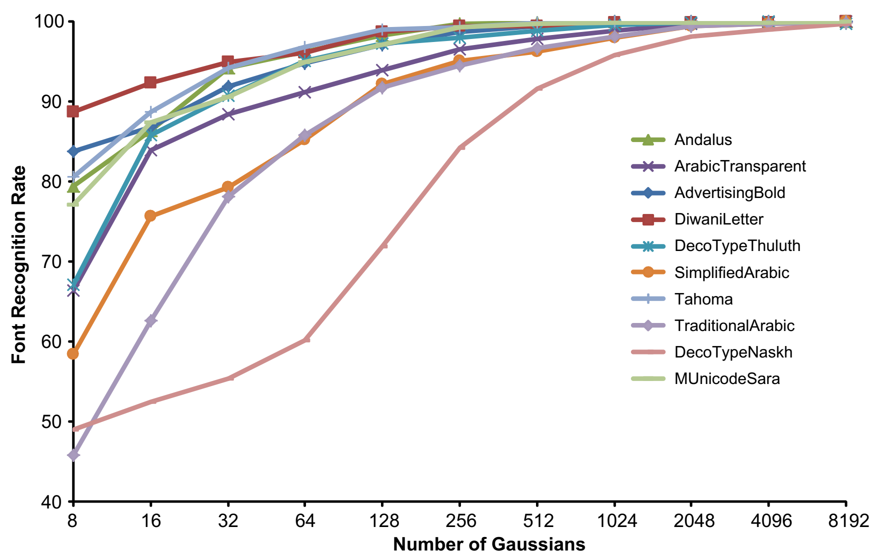

Fig. 4. Font recognition rate as a function of the number of Gaussians in the font model with GMMs.

3. font and size recognition;
4. word recognition.

5.2.1. Font recognition results

The font recognition results are shown in Table 3 (a). The results are overall good excepted for two fonts "Arabic Transparent" and "Simplified Arabic" (see Fig. 2, font B and F). The morphological similarity between these two fonts explains such results. We performed a confusion matrix analysis, clearly showing that most of the errors are between these similar fonts. For this reason, we have extended our tests by considering these two fonts as a single font in the training and recognition step. A significant improvement in the recognition rate was recorded from 80.8% to 99.5% for the "Arabic Transparent" font and from 67.0% to 99.3% for the "Simplified Arabic" font. The global average of font recognition rate increased from 94.5% to 99.6%.

As shown in Table 3(a), we observe that the best results are obtained for font-sizes between 10 and 16. Slight degradations are observed for font-sizes below 10 and above 16. The reason for this has to be found in the normalization procedure of all images towards 30 pixels height that introduces up-scaling and down-scaling variabilities for, respectively, the small font-sizes and for the large font sizes.

The reached performance of 99.6% is the result of an overall tuning of the proposed font recognition system, including a selection of features (some specific to low-resolution images) and a systematic optimization of the GMMs parameters. An interesting question is in the applicability of the obtained system on other databases. For this reason, we have also attempted to compare the performance of our font recognition system using high resolution gray scale images obtained with a 300 dpi scanner. The fonts of this database are the same as in APTI. We have used 1000 words for each font in the training step and 1000 different words for each font in the test step. The obtained mean recognition rate for all fonts is 92%. This performance is below the one measured on the APTI database. However, the system remains relatively robust considering the extra variabilities due to the scanning artefacts.

5.2.2. Size recognition results

Supposing that we know the font, we have developed a size recognition system (to identify the exact size value among the ten used sizes) for each font. For all fonts, we have good results (*Advertising Bold*: 96.9%; *Andalus*: 99.3%; *Arabic Transparent*: 98.2%; *M Unicode Sara*: 98.9%; *Tahoma*: 98.8%; *Simplified Arabic*: 97.8%; *Traditional Arabic*: 96.2%; *DecoType Naskh*: 92.2%; *DecoType*

Table 3
Recognition system results when where "Arabic Transparent" and "Simplified Arabic" are considered separately: (a) Font recognition rate, (b) Font and size recognition rates.

Font/size	6	7	8	9	10	12	14	16	18	24	Mean RR
(a)											
Andalus	99.5	99.9	99.9	99.9	100.0	100.0	100.0	100.0	100.0	100.0	99.9
Arabic transparent	73.8	82.8	84.2	81.7	80.5	82.9	80.3	83.2	81.4	77.5	80.8
Advertising bold	99.3	99.8	99.7	99.7	99.8	99.9	99.8	99.8	99.8	99.7	99.7
Diwani letter	99.0	99.4	99.5	99.6	99.8	99.9	99.8	99.8	99.6	99.4	99.6
DecoType Thuluth	98.3	99.1	99.6	99.6	99.8	99.8	99.7	99.6	99.6	99.1	99.4
Simplified Arabic	65.6	69.2	69.8	67.6	67.4	66.1	65.2	64.9	66.5	67.6	67.0
Tahoma	98.9	99.7	99.7	100.0	100.0	100.0	100.0	100.0	100.0	100.0	99.8
Traditional Arabic	100.0	99.9	99.7	99.8	99.6	99.6	99.6	99.8	99.7	99.6	99.7
DecoType Naskh	95.3	98.2	98.7	99.3	99.6	99.7	99.7	99.8	99.9	99.9	99.0
M Unicode Sara	100.0	100.0	100.0	100.0	99.9	99.9	99.9	99.9	99.8	99.9	99.9
Mean RR	93.0	94.8	95.1	94.7	94.6	94.8	94.4	94.7	94.6	94.3	94.5
(b)											
Andalus	99.3	99.4	99.1	99	98.8	99.3	99.1	98.8	98.9	99.5	99.1
Arabic transparent	80.5	82.2	85.6	82.3	84.6	85.1	82.5	81.1	82.3	80.2	82.6
Advertising bold	98.6	98.6	97.1	97.6	97.8	95.8	96.3	94.8	94.2	95.6	96.6
Diwani letter	95.1	91.2	92.9	91	91.9	90.7	88	90.2	90.6	96.2	91.8
DecoType Thuluth	96.4	92.7	92.8	89.7	92.1	89.4	88.2	89.2	91.7	95.4	91.8
Simplified Arabic	77.5	72.2	72.9	71.8	73.6	69	69.1	70	71.2	74.4	72.2
Tahoma	98.7	98.7	99	98.8	98.6	99	99.2	99.3	97.8	98.2	98.7
Traditional Arabic	98.7	95.5	94.5	95.6	96.1	96.5	93.9	92.5	95.1	98.7	95.7
DecoType Naskh	96.8	94	90.6	92.6	92.2	90.5	89.6	89.7	91.3	94.5	92.2
M Unicode Sara	99.2	99.7	98.8	98.9	99.2	98.7	98.3	98.5	98.1	99	98.8
Mean RR	94.1	92.4	92.3	91.7	92.5	91.4	90.4	90.4	91.1	93.2	91.9

Table 4
Word recognition results: (a) Global multi-font System, (b) Cascading multi-font system.

Font	Character RR	Word RR	Font	Character RR	Word RR
(a)					
Andalus	98.0	85.3	Simplified Arabic	98.6	88.2
Arabic transparent	99.3	91.2	Tahoma	98.6	83.5
Advertising bold	97.3	78.3	Traditional Arabic	92.7	62.3
Diwani letter	77.6	28.8	DecoType Naskh	87.2	49.8
DecoType Thuluth	86.4	47.0	M Unicode Sara	97.7	84.7
Mean RR				93.3	69.9
(b)					
Andalus	99.6	99.1	Simplified Arabic	99.0	96.7
Arabic transparent	99.4	97.7	Tahoma	99.5	96.7
Advertising bold	98.9	96.5	Traditional Arabic	98.8	95.3
Diwani letter	96.7	91.4	DecoType Naskh	96.1	81.6
DecoType Thuluth	98.0	92.3	M Unicode Sara	99.1	95.7
Mean RR				98.4	93.7

Thuluth: 92.0%; *Diwani Letter*: 91.7%) with an average of 96.2% for size recognition rate.

The size recognition system shows better performances for simple fonts with no or few ligatures (e.g., "Arabic Transparent", "Tahoma" and "M Unicode Sara") than for complex fonts rich in overlaps (e.g., "Diwani Letter" and "Decotype Naskh")

5.2.3. Font and size recognition results

Instead of training two distinct systems for font and size recognition, we have also investigated conjoint training and recognition of 100 (font, size) classes. All results are described in Table 3(b). We noticed that we have the same problem for font recognition related to the strong similarity between "Arabic Transparent" and "Simplified Arabic". Considering these fonts as different ones, the mean recognition rate of our font and size recognition system is about 91.9% (see Table 3(b)). And considering "Arabic Transparent" and "Simplified Arabic" as the same font (one model for both fonts), the mean recognition rates increase from 82.6% to 98.0% for the "Arabic Transparent" font and from 72.2% to 97.9% for the "Simplified Arabic" font. The global average of font and size recognition rate increased to 96.1%. In this system, we also recognize simple fonts better than complex ones.

5.2.4. Word recognition results

To show the impact of a font recognition system in a multi-font OCR, we compare the results of two Arabic recognition systems for ultra-low resolution word images. The first is a global multi-font system working in two steps: feature extraction and word recognition using font-independent models. The second is a cascading system working in four steps: font feature extraction, font recognition, word feature extraction and word recognition using font-dependent models. Both systems share the same word feature extraction and are based on Hidden Markov models (HMMS). The evaluation is carried on using HTK, a freely available toolkit for HMMs (Young et al., 2001). Mixture of Gaussians is used to model each character. A Baum–Welch estimation procedure is used to iteratively refine the model parameters including weights, means and variances. In the training procedure, we applied a classical binary splitting procedure to increase the number of Gaussian mixtures up to 512 mixtures per HMM states. At recognition time, an ergodic HMM is composed using all sub-models. All transitions from one sub-model to the other are permitted. This approach allows recognizing potentially any word in an open vocabulary fashion and no constraints from a dictionary. The recognition is done by computing the best state sequence in the HMM using a Viterbi procedure. Similar systems are presented in (Slimane et al., 2010).

We report in Table 4 the word recognition results of the global and cascading multi-font word recognition systems. The mean performances of the global multi-font system (see Table 4(a)) are 69.9% and 93.3% for word and character recognition respectively. In contrast, the average recognition rates are 93.7% and 98.4% for word and character recognition using the cascading (see Table 4(b)) system (Font Recognition system followed by mono-font word recognition system). This result shows clearly the potential benefit of using a cascading system, i.e. font identification followed by word/text recognition: we earn more than 23% on word recognition and 5% on character recognition corresponding with an error reduction of 79% and 76%, respectively.

6. Conclusion

In this paper, a new, simple and robust method for font/size/font and size recognition in ultra-low resolution Arabic text images has been proposed. It is based on Gaussian Mixture Models (GMMs) for the estimation of font/size/font and size model likelihoods with respect to local features. The feature extraction uses a fixed-length sliding window from right to left on the text image. A main advantage of this approach is that no a priori segmentation into characters is needed.

In an experimental evaluation on the APTI database, we have demonstrated the high potential of the proposed font recognition method in the context of single word recognition. Both the character and word recognition error could be reduced by over 70% when using font recognition first, followed by font-specific OCR.

References

Abuhaiba, Ibrahim, 2003. Arabic font recognition based on templates. The Internat. Arab J. Inform. Technol. 1, 333.

Abuhaiba, Ibrahim, 2005. Arabic font recognition using decision trees built from common words. J. Comput. Inform. Technol. 13, 21122.

AlKhateeb, Jawad H., Ren, Jinchang, Jiang, Jianmin, Al-Muhtaseb, Husni, 2011. Offline handwritten arabic cursive text recognition using hidden markov models and re-ranking. Pattern Recognition Lett. 32 (8), 1081–1088.

Al-Muhtaseb, A. Husni, Mahmoud, A. Sabri, Qahwaji, Rami S., 2008. Recognition of off-line printed arabic text using hidden markov models. Signal Process. 88 (12), 2902–2912.

AL-Shatnawi, Atallah Mahmoud, AL-Salaimeh, Safwan, AL-Zawaideh, Farah Hanna, Omar, Khairuddin, 2011. Offline arabic text recognition an overview. World Comput. Sci. Inform. Technol. J. 1 (5), 184–192.

Avilé-Cruz, Carlos, Villegas-Cortes, Juan, Ocampo-Hidalgo, J., 2006. A robust font recognition using invariant moments. In: Proc. of the 5th WSEAS Internat. Conf. on Applied Computer Science, ACOS'06, pp. 114–117.

Avilés-Cruz, Carlos, Villegas, Juan, Arechiga-Martnez, Ren, Escarela-Perez, Rafael 2004. Unsupervised font clustering using stochastic versio of the em algorithm and global texture analysis. In: Alberto Sanfeliu, JosMartnez Trinidad, Jess Carrasco Ochoa (Eds.), Progress in Pattern Recognition, Image Analysis and Applications, vol. 3287 of Lecture Notes in Computer Science, pp. 3–25.

Avilés-Cruz, Carlos, Rangel-Kuoppa, Risto, Reyes-Ayala, Mario, Andrade-Gonzalez, A., Escarela-Perez, Rafael, 2005. High-order statistical texture analysis-font recognition applied. Pattern Recognition Lett. 26, 135–145, Elsevier Science Inc., New York, USA.

Bapst, Frédéric, Ingold, Rolf, 1998. Using typography in document image analysis. In: Proc. of the 7th Internat. Conf. on Electronic Publishing, Held Jointly with the 4th Internat. Conf. on Raster Imaging and Digital Typography: Electronic Publishing, Artistic Imaging, and Digital Typography, EP '98/RIDT '98, pp. 240–251.

Bataineh, Bilal, Abdullah, Siti Norul Huda Sheikh, Omar, Khairudin, 2011. A statistical global feature extraction method for optical font recognition. In: Proc. of the Third Internat. Conf. on Intelligent Information and Database Systems, volume Part I of ACIIDS'11, pp. 257–267.

Bataineh, Bilal, Abdullah, Siti Norul Huda Sheikh, Omar, Khairuddin, 2012. A novel statistical feature extraction method for textual images: optical font recognition. Expert Syst. Appl. 39, 5470–5477.

Ben Amara, Najoua Essoukri, Faouzi, Bouslama, 2003. Classification of arabic script using multiple sources of information: state of the art and perspectives. Internat. J. Document Anal. Recognition 5, 195–212.

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.

Ben Moussa, S., Zahour, A., Kherallah, M., Benabdelhafid, A., Alimi, A., 2006. Utilisation de nouveaux paramètres à base de fractale pour la discrimination des fontes arabes. In: Laurence Likforman-Sulem (Ed.), Actes du 9ème Colloque International Francophone sur l'Ecrit et le Document, pp. 283–288. SDN06.

Ben Moussa, S., Zahour, A., Benabdelhafid, A., Alimi, A.M., 2010. New features using fractal multi-dimensions for generalized arabic font recognition. Pattern Recognition Lett. 31, 361–371, Elsevier Science Inc., New York, USA.

Borji, A., Hamidi, M., 2007. Support vector machine for persian font recognition. Eng. Technol. 2, 10–13.

Bushofa, B.M.F., Spann, M., 1997. Segmentation of arabic characters using their contour information. In: 13th Internat. Conf. on Digital Signal Processing Proc. DSP'97, vol. 2 July, pp. 683–686.

Chaker, Ilham, Harti, Mostafa, Qjidah, Hassan, Ben Slimane, Rachid, 2010. Recognition of arabic characters and fonts. Internat. J. Eng. Sci. Technol. 2, 5959–5969.

Cheriet, Mohamed 2008. Visual recognition of arabic handwriting: challenges and new directions. In: Proc. of the 2006 Conference on Arabic and Chinese handwriting recognition, SACH'06, pp. 1–21.

Dempster, A.P., Laird, N.M., Rubin, D.B., 1977. Maximum likelihood from incomplete data via the em algorithm. Roy. Statis. Soc. Ser. B Methodol. 39, 1–38.

Ding, Xiaoqing, Chen, Li, Wu, Tao, 2007. Character independent font recognition on a single chinese character. IEEE Trans. Pattern Anal. Mach. Intell. 29, 195–204.

Farshideh Einsele-Aazami, 2008. Recognition of Ultra Low resolution, Anti-Aliased text with Small Font Sizes. PhD thesis. Faculty of Science, University of Fribourg, Switzerland.

Fu, Jiacai, Guo, Mingliang, Tang, Xianglong, 2006. Font recognition of chinese character based on multi-scale wavelet. In: Proc. of the 5th WSEAS Internat. Conf. on Circuits, Systems, Electronics, Control & Signal Processing, pp. 90–93.

Ha, Ming-Hu, Tian, Xue-Dong, Zhang, Zi-Ru 2005. Optical font recognition based on gabor filter. In: Proc. of 2005 Internat. Conf. on Machine Learning and Cybernetics, vol. 8, pp. 4864–4869.

Imani, Maryam Bahojb, Keyvanpour, Mohamad Reza, Azmi, Reza, 2011. Semi-supervised persian font recognition. Procedia Comput. Sci. 3, 336–342.

Jamjuntr, Pitchaya, Dejdumrong, Natasha, 2009. Thai font type recognition using linear interpolation analysis. In: Proc. of the 2009 Sixth Internat. Conf. on Computer Graphics, Imaging and Visualization, CGIV '09, pp. 406–409.

Jeong, C., Kwag, H., Kim, S., Kim, J., Park, S., 2003. Identification of font styles and typefaces in printed korean documents. In: Tengku Sembok, Halimah Zaman, Hsinchun Chen, Shalini Urs, and Sung-Hyon Myaeng (Eds.) Digital Libraries: Technology and Management of Indigenous Knowledge for Global Access, volume 2911 of Lecture Notes in Computer Science, pp. 666–669.

Jung, M., Shin, Y., Srihari, S., 1999. Multifont classification using typographical attributes. In: Proc. of the Fifth Internat. Conf. on Document Analysis and Recognition, ICDAR '99, pp. 353–356.

Kanoun, Slim, Alimi, Adel M., Lecourtier, Yves, 2011. Natural language morphology integration in off-line arabic optical text recognition. IEEE Trans. Syst. Man Cybernet. Part B: Cybernet. 41, 579–590.

Khorsheed, M.S., 2007. Offline recognition of omnifont arabic text using the hmm toolkit (htk). Pattern Recognition Lett. 28 (12), 1563–1571.

Khosravi, Hossein, Kabira, Ehsanollah, 2010. Farsi font recognition based on Sobel–Roberts features. Pattern Recognition Lett. 31, 75–82.

Khoubyari, Siamak, Hull, Jonathan, 1996. Font and function word identification in document recognition. Comput. Vis. Image Underst. 63, 66–74, Elsevier Science Inc., New York, NY, USA.

Kim, S.H., Kim, S.S., Kwag, H.K., Lee, G.S., 2002. Optical font recognition for printed korean characters using serif pattern of strokes. In: Proc. of the Internat. Technical Conf. on Circuits/Systems, Computers and Communications, p. 916–919.

Kim, Soo hyung, Kwag, Hee K., Suen, ChingY., 2004. Word-level optical font recognition using typographical features. Internat. J. Pattern Recognition Artificial Intell. 18, 541–561.

Lee, Chang Woo, Kang, Hyun, Jung, Keechul, Kim, Hang Joon , 2003. Font classification using nmf. In: Nicolai Petkov, Michel Westenberg (Eds.) Computer Analysis of Images and Patterns, vol. 2756 of Lecture Notes in Computer Science, pp. 470–477.

Lin, Chi-Fang, Fang, Yu-Fan, Juang, Yau-Tarng, 2001. Chinese text distinction and font identification by recognizing most frequently used characters. Image Vision Comput., 329–338.

Lorigo, Liana M., Govindaraju, Venu, 2006. Offline arabic handwriting recognition: a survey. IEEE Trans. Pattern Anal. Mach. Intell. 28, 712–724.

Ma, Huanfeng, Doermann, David S., 2005. Font identification using the grating cell texture operator. In: SPIE Conf. on Document Recogntion and Retreival XXII, pp. 148–156.

Mallikarjunaswamy, B.P., Karunakara, K., 2010. Classification of kannada font using hilberthuang transform method. Internat. J. Inform. Technol. Knowl. Manage. 2, 333–335.

Mehran, Ramin, Pirsiavash, Hamed, Razzazi, Farbod, 2005. A front-end ocr for omni-font persian/arabic cursive printed documents. In: Digital Image Computing: Techniques and Applications, vol. 0, IEEE Computer Society, Los Alamitos, CA, USA, p. 56.

Omidyeganeh, M., Nayebi, K., Azmi, R., Javadtalab, A., 2005. A new segmentation technique for multi font farsi/arabic texts. In: Proc. of IEEE Internat. Conf. on Acoustics Speech and Signal Processing (ICASSP '05) 2 (18–23) pp. 757–760.

Ozturk, S., Sankur, B., Abak, A.T., 2001. Font clustering and cluster identification in document images. J. Electronic Imaging 10, 418–430.

Ramanathan, R., Soman, K.P., Thaneshwaran, L., Viknesh, V., Arunkumar, T., Yuvaraj, P., 2009. A novel technique for english font recognition using support vector machines. In: Internat. Conf. on Advances in Recent Technologies in Communication and Computing, pp. 766–769.

Schreyer, Angela, Suda, Peter, Maderlechner, Gerd, 1999. A formal approach to textons and its application to font style detection. In: Selected Papers from the Third IAPR Workshop on Document Analysis Systems: Theory and Practice, DAS '98, pp. 72–83.

Shirali-Shahreza, Mohammed Hassan, Shirali-Shahreza, Sajad., 2006. Persian/arabic text font estimation using dots. In: Internat. Symposium on Signal Processing and Information Technology, vol. 0, pp. 420–425.

Slimane, Fouad, Ingold, Rolf, Alimi, Adel M., Hennebert, Jean, 2008. Duration models for arabic text recognition using hidden markov models. CIMCA, 838–843.

Slimane, Fouad, Ingold, Rolf, Kanoun, Slim, Alimi, Adel M., Hennebert, Jean, 2009. A new arabic printed text image database and evaluation protocols. In: Proc. of the Eleventh Internat. Conf. on Document Analysis and Recognition, pp. 946–950.

Slimane, Fouad, Ingold, Rolf, Kanoun, Slim, Alimi, Adel M., Hennebert, Jean, 2009. Database and evaluation protocols for arabic printed text recognition. Technical report. DIUF-University of Fribourg – Switzerland.

Slimane, Fouad, Kanoun, Slim, Alimi, Adel M., Hennebert, Jean, Ingold, Rolf, 2010. Comparison of global and cascading recognition systems applied to multi-font arabic text. In: Proc. of the 10th ACM Symposium on Document Engineering, DocEng '10, pp. 161–164.

Slimane, Fouad, Kanoun, Slim, El Abed, Haikal, Alimi, Adel M., Ingold, Rolf, Hennebert, Jean, 2011. In: ICDAR 2011 – Arabic recognition competition: Multi-font multi-size digitally represented text, pp. 1449–1453.

Sternby, Jakob, Morwing, Jonas, Andersson, Jonas, Friberg, Christer, 2009. On-line arabic handwriting recognition with templates. Pattern Recognition 42 (12), 3278–3286, New Frontiers in Handwriting Recognition.

Sun, Hung-Ming, 2006. Multi-linguistic optical font recognition using stroke templates. In: Proc. of the 18th Internat. Conf. on Pattern Recognition, volume 2 of ICPR '06, pp. 889–892.

Tsai, M.Y., Liu, C.L., Chuang, K.T., Huang, S.M., 2001. Toward font identification for printed chinese characters. In: Proc. of the National Computer Symposium, vol. B, p. 170177.

Wachenfeld Steffen, Klein, Hans-Ulrich, Jiang, Xiaoyi, 2006. Recognition of screen-rendered text. In: Proc. of the 18th Internat. Conf. on Pattern Recognition, volume 2 of ICPR '06, pp. 1086–1089.

Yang, Zhihua, Yang, Lihua, Qi, Dongxu, Suen, Ching Y., 2006. An emd-based recognition method for chinese fonts and styles. Pattern Recognition Lett. 27, 1692–1701.

Young, S., Evermann, G., Kershaw, D., Moore, D., Odell, J., Ollason, D., Valtchev, V., Woodland, P., 2001. The HTK Book. Cambridge University Engineering Dept..

Zaghden, Nizar, Ben Moussa, Sami, Alimi, Adel M., 2006. Reconnaissance des fontes arabes par l'utilisation des dimensions fractales et des ondelettes. In: Actes du 9ème Colloque International Francophone sur l'Ecrit et le Document, pp. 277–282.

Zahedi, Morteza, Eslami, Saeideh, 2011. Farsi/arabic optical font recognition using sift features. Procedia Comput. Sci. 3, 1055–1059.

Zhang, Li, Lu, Yue, Tan, Chew Lim, 2004. Italic font recognition using stroke pattern analysis on wavelet decomposed word images. In: Proc. of the 17th Internat. Conf. on Pattern Recognition, vol. 4, pp. 835–838.

Zhu, Yong, Tan, Tieniu, Wang, Yunhong, 2001. Font recognition based on global texture analysis. IEEE Trans. Pattern Anal. Mach. Intell. 23, 1192–1200.

Zramdini, Abdelwahab, Ingold, Rolf, 1993. Optical font recognition from projection profiles. Electronic Publishing, pp. 249–260.

Zramdini, Abdelwahab, Ingold, Rolf, 1998. Optical font recognition using typographical features. IEEE Trans. Pattern Anal. Mach. Intell. 20, 877–882.