Font Discrimination using Fractal Dimensions

A. A. Hajian Nezhad* and S. Mozaffari*(C.A)

Abstract: One of the important problems in OCR systems is discrimination of fonts in machine printed document images. This task improves performance of OCR systems by providing some information about document logical structure or increasing recognition rates. Proposed methods for font discrimination in this paper are based on various fractal dimensions. First, some predefined fractal dimensions were combined with directional methods to enhance font differentiation. Then, a novel fractal dimension (FTCPH) was introduced in this paper which considers font recognition as texture identification. This new descriptor is independent of document content and can be used for font discrimination in different languages. Experimental results on different pages, written by several types of fonts, show that fractal geometry can overcome the complexities of font recognition problem.

Keywords: Fractal Dimension (FD), Fractal Geometry, Optical Character Recognition (OCR), Optical Font Recognition (OFR).

1 Introduction

Image processing and computer vision play a vital role in our daily life. They have found a wide range of application from information security [1] and processing [2] to video surveillance [3]. Optical Character Recognition (OCR) was one of the very first applications of computer vision industry with tangible benefits.

Converting historical books, newspapers, and other types of documents into electronic scripts through scanners is an essential task in digital libraries. Nowadays, OCR systems have been utilized by many individuals to change these scanned text images into machine-encoded forms [4].

Typical OCR systems have been made of several modules such as preprocess, layout analysis, character recognition and etc. Since character shapes in different fonts have different appearance, Optical Font Recognition (OFR) is a new module recently added to OCR systems.

Previous font recognition methods can be classified into two groups [5]: typographical and textural features. Typographical features include features such as character skews, height and width, projections in upper, centre and lower zones of the line and etc. Nowadays typographical based algorithms are not so popular because of some disadvantages such as being sensitive to noise and requiring high resolution scanned images. Some typographical features are proposed in [6-8].

Wavelet transform, Gabor filter, Sobel-Robert gradients, and Fractal dimensions are the most famous textural features used for OFR [5, 9-11]. Previous efforts demonstrated that textural features are more applicable than typographical.

Although font recognition has been performed in many languages, unfortunately it is still in the beginning steps in Farsi/Arabic languages. Lack of OFR module in Farsi/Arabic OCR systems is partly due to the morphological complexities of these languages scripts. Some of the most important challenges are connection among characters in words, overlap among characters of words, variety of different shapes of characters in beginning, middle and end of words. According to Fig. 1, there are more than 100 character shapes in Farsi/Arabic scripts.

As mentioned before, font recognition is a complicated task especially in Farsi/Arabic languages. Since fractal geometry overcomes complexities in other fields, we decided to utilize it for font discrimination purpose.

In 1983, Mandelbrot established fractal geometry to describe complex phenomenons that Euclidean geometry had been failed. Because Euclidean geometry only deals with integer dimension objects but fractal geometry deals with fractional objects.

Over the last years, fractal geometry has been applied frequently in many applications such as pattern recognition, texture analysis, segmentation and etc. Various methods were proposed to estimate Fractal

Iranian Journal of Electrical & Electronic Engineering, 2014.
Paper first received 12 Nov. 2012 and in revised form 6 Oct. 2013.
* The Authors are with the Electrical and Computer Engineering Department, Semnan University, Semnan, Iran.
E-mails: a_hajiannezhad@sun.semnan.ac.ir and Mozaffari@semnan.ac.ir.

N0	Character	Isolated	First	Middle	Last
1	Alef	ا(آ)	ا(آ)	ـا	ـا
2	Be	ب	بـ	ـبـ	ـب
3	* Pe	پ	پـ	ـپـ	ـپ
4	Te	ت	تـ	ـتـ	ـت
5	Se	ث	ثـ	ـثـ	ـث
6	Jim	ج	جـ	ـجـ	ـج
7	* Che	چ	چـ	ـچـ	ـچ
8	He	ح	حـ	ـحـ	ـح
9	Khe	خ	خـ	ـخـ	ـخ
10	Dal	د	د	ـد	ـد
11	Zal	ذ	ذ	ـذ	ـذ
12	Re	ر	ر	ـر	ـر
13	Ze	ز	ز	ـز	ـز
14	* Zhe	ژ	ژ	ـژ	ـژ
15	Sin	س	سـ	ـسـ	ـس
16	Shin	ش	شـ	ـشـ	ـش
17	Sad	ص	صـ	ـصـ	ـص
18	Zad	ض	ضـ	ـضـ	ـض
19	Ta	ط	طـ	ـطـ	ـط
20	Za	ظ	ظـ	ـظـ	ـظ
21	Ayn	ع	عـ	ـعـ	ـع
22	Ghayn	غ	غـ	ـغـ	ـغ
23	Fe	ف	فـ	ـفـ	ـف
24	Ghaf	ق	قـ	ـقـ	ـق
25	Kaf	ک	کـ	ـکـ	ـک
26	* Gaf	گ	گـ	ـگـ	ـگ
27	Lam	ل	لـ	ـلـ	ـل
28	Mim	م	مـ	ـمـ	ـم
29	Noon	ن	نـ	ـنـ	ـن
30	Waw	و	و	ـو	ـو
31	He	ه	هـ	ـهـ	ـه
32	Ye	ی	یـ	ـیـ	ـی

Fig. 1 Farsi characters in different situations.

Dimension (FD) of an object. In this article some of them will be utilized for font recognition.

All the fractal objects have these three properties [12]: being self-similar, being complicated in tiny scales, having fractured dimension. Self-similarity (the first property) in fractal objects can be categorized into three categories: perfect self-similar objects such as Broccoli cabbage, imperfect self-similar objects such as mountains, statistical self-similar objects.

Researches show that a huge number of environs objects are located in statistical self-similar objects category. The obtained experiments in [11] demonstrated that there are fractal properties in text images.

2 Related Works

In this section a review of some textural algorithms for font recognition will be presented.

Multi-channel Gabor filtering technique has been shown to be particularly useful for analyzing text images and it was proposed by Zhu et al to identify different English fonts [10]. The spatial frequency and orientation contents represent the features of each texture. They used pairs of isotropic Gabor filters with quadrature phase relationship. From an isotropic Gaussian function, even and odd symmetric Gabor filters $h_e(x,y), h_o(x,y)$ are obtained. For a given input image, the outputs of $H_e(u,v)$ and $H_o(u,v)$ are combined to provide a single channel output.

Yang et al in [13] proposed a font recognition method based on Empirical Mode Decomposition (EMD) method in which any complicated dataset can be decomposed into finite (often smaller) number of intrinsic mode functions. They applied Hilbert-Huang transform on document image and performed font recognition. Hilbert-Huang Transform (HHT) is an analysis method used for nonlinear and non-stationary data. By analyzing and comparing a great number of Chinese characters, five basic strokes had been selected to characterize stroke features of Chinese fonts. Based on them, stroke feature sequence of a given text block are calculated.

Ding et al in [9], employed a 3-level wavelet transform for Chinese font identification. They applied a wavelet transform on character images and extracted wavelet features from the transformed images. After a Box-Cox transformation and LDA (Linear Discriminant Analysis) process, the discriminating features for font recognition were obtained.

Khosravi and et al [5] proposed a new feature extraction method for Farsi font recognition in line level. The proposed feature was based on image gradients in 16 directions using Sobel and Roberts operators. Since Sobel is a horizontal/vertical filter and Roberts is diagonal, combination of these two operators improves font recognition performance. They claimed that their proposed algorithm requires much less computation time.

Sami Ben Moussa et al [11] used two FD methods namely Box Counting Dimension (BCD) and Dilation Counting Dimension (DCD) for ten Arabic font recognition. Box Counting is widely used due to relatively simple mathematical calculation and estimation. The binary image is divided into a grid of boxes of size 'r' and number of boxes which are not empty is counted as $N(r)$. These steps are repeated for different amounts of 'r'. The slope of the linear regression of the graph $\log N(r)$ versus $\log(1/r)$ is BCD FD as Eq. (1).

$$\mathrm{BCD}(r) = \lim_{r \to 0} \frac{\log N(r)}{\log(1/r)} \tag{1}$$

In DCD, each occupied point is surrounded by a square of size d. The size of these squares is then gradually enlarged and then the total surface V(d) covered at each stage is measured. By dividing this total surface by the surface of a test square (d), we get an approximation of the number of elements V(d) necessary to cover the whole. DCD FD can be calculated through Eq. (2).

$$DCD(d) = lim_{d\to 0}\left(n - \frac{log\, V(d)}{log(d)}\right) \quad (2)$$

where d is the maximum dilation radius, n is the dimension of a space, and V is the dilation body.

3 Overview of Fractal Dimension Algorithms

Various methods have been proposed to measure FD in objects. In this paper BCD, DLA, Higuchi and Variogram FD algorithms are utilized.

3.1 Box Counting Dimension

As mentioned above, there are several techniques to calculate an object FD. The simplest and most widely method is BCD. The BCD is obtained as follows:

1. A binarization algorithm is applied on the input image.
2. The binary image is divided into a grid of boxes of size 'r' as shown in Fig. 2.
3. Number of boxes which are not empty is counted as $\mathrm{N(r)}$.
4. The two previous steps must be repeated for different amounts of 'r'.
5. A graph of $\log\ \mathrm{N}(r)$ versus $\log(1/r)$ for each image is produced.
6. BCD is estimated by linear regression of the graph $\log\ N(r)$ versus $\log(1/r)$.
7. The obtained slope of the linear regression is BCD FD.

Assume a square as shown in Fig. 2.

$$\begin{cases} r_0 = 1 \ \rightarrow \log\left(\frac{1}{1}\right) = 0 \\ N(r_0) = 1 \rightarrow \log(1) = 0 \end{cases}$$

$$(0,0)$$

$$\begin{cases} r_1 = {}^1\!/_2 \rightarrow \log\left(\frac{1}{1/2}\right) = 0.301 \\ N(r_1) = 4 \rightarrow \log(4) = 0.602 \end{cases}$$

$$(0.301,0.602)$$

$$\begin{cases} r_2 = {}^1\!/_4 \rightarrow \log\left(\frac{1}{1/4}\right) = 0.602 \\ N(r_2) = 16 \rightarrow \log(16) = 1.204 \end{cases} \quad (3)$$

$$(0.602,1.204)$$

$$\begin{cases} r_3 = {}^1\!/_8 \rightarrow \log\left(\frac{1}{1/8}\right) = 0.90308 \\ N(r_3) = 64 \rightarrow \log(64) = 1.8061 \end{cases}$$

$$(0.90308, 1.8061)$$

The coordinates in Eq. (3) are plotted in Fig. 3 and the slope of the obtained line in this figure corresponds the BCD FD of square. As it can be seen, FD of the square converges to 2 which is the same as Euclidean dimension.

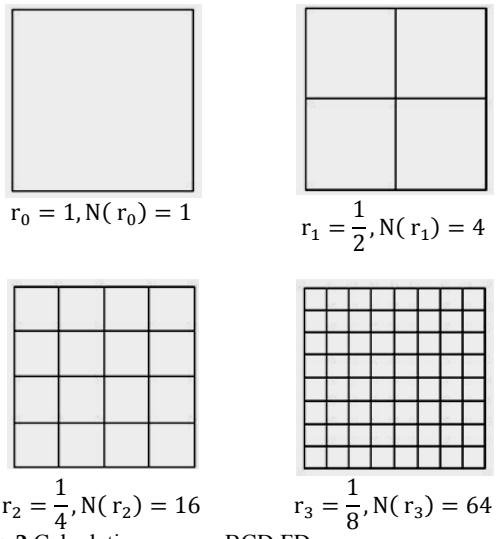

Fig. 2 Calculating a squre BCD FD.

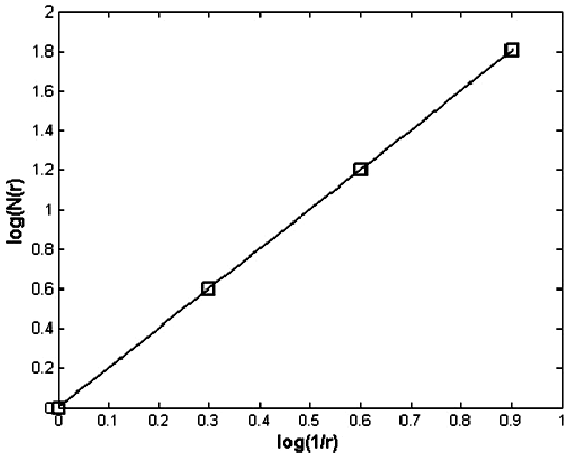

Fig. 3 Linear regression for a squre BCD FD calculation.

3.2 Diffusion Limited Aggregate

Similar to BCD, diffusion limited aggregate (DLA) FD method also must be applied to binary images. The main steps of this algorithm are as follows [14]:

1. Apply a binarization algorithm on the input image.
2. Individuating the containing information pixels (black pixels).
3. Selecting some pixels from individuated pixels randomly.
4. Put some boxes of the radius R_i centered on the selected randomly pixels as shown in Fig. 4.
5. Calculate number of pixels containing information in every box of radius R_i.
6. While $R_i < R_{max}$ calculate the average number of pixels containing information in all the boxes of the same radius.
7. Calculate average number of pixels containing information in all boxes for all radiuses.

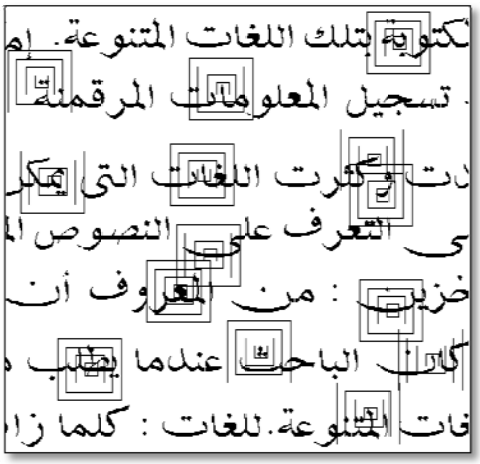

Fig. 4 Selecting randomly some pixels containing information.

8. Repeat previous step several times (the repetition number is arbitrary).
9. Finally, DLA can be estimated by linear regression between $\log[(\frac{\mathrm{M(R)}}{\mathrm{M_0}})^{\mathrm{q-1}}]$ and $\frac{1}{\mathrm{q-1}} \times \log\left[\frac{\mathrm{R}}{\mathrm{L}}\right]$, through Eq. (4).

$$\begin{cases} If \quad q = 1 \quad then \quad D_1 \approx \dfrac{(D_{1-c} + D_{1+c})}{2} \\ \\ Else \quad D_q = \dfrac{1}{q-1} \dfrac{log[(\frac{M(R)}{M_0})^{q-1}]}{log[\frac{R}{L}]} \end{cases} \tag{4}$$

where $\mathrm{M_0}$ is the number of containing information pixels. M(R) represents the average number of pixels containing information in the boxes of the size $\mathrm{R_i}$. L is the average number of pixels containing information for all box sizes and c parameter is epsilon and q is an integer number in the range of $-\infty < q < +\infty$.

3.3 Higuchi

Unlike BCD and DLA fractal dimensions, Higuchi is a FD algorithm which is just applicable on 1D signals like $\{X(1), X(2), \cdots, X(N)\}$. Higuchi's principle is based on a measure of length $L(k)$, based on a segment of k samples as a unit (see Eq. (5)).

$$X_m^k : \{X(m), X(m+k), X(m+2k), \cdots, X(m+int((N-m)/k) * k)\} \tag{5}$$

where N is total number of samples. Initial value is represented by $m\,(m = 1,2,\cdots,k)$ where k is interval value. Operator $int(r)$ shows integer part of a real number r. For every X_m^k sequence as shown in Eq. (5), $L_m(k)$ is calculated by Eq. (6).

$$L_m(k) = \frac{\sum_{i=1}^{\left[\frac{L-m}{k}\right]} |X(m+ik) - X(m+(i-1)k)|(N-1)}{\left[\frac{N-m}{k}\right]k} \tag{6}$$

Finally, Higuchi FD is calculated through Eqs. (7) and (8).

$$L(k) = \sum_{i=1}^{k} L_m(k) \tag{7}$$

$$L(k) \approx k^{-D} \tag{8}$$

where D is the estimated Higuchi FD.

3.4 Variogram

Variogram is a directional FD algorithm which can be used for 1D, 2D and 3D signals. This algorithm is based on the differences between pairs of samples in specific relative orientation. Variogram FD can be estimated as follows [15-17] (through Eqs. (9) and (10)):

$$\gamma(h) = \frac{1}{2n(h)} \sum_{i=1}^{n(h)} [Z_i - Z_{i+h}]^2 \tag{9}$$

$$FD = 2 - \frac{1}{2}\left(\frac{log\ (\gamma(h))}{log\ (h)}\right) \tag{10}$$

where h is the distance between samples. Z_i, and Z_{i+h} are the signal values with difference h. Function $n(h)$ is the number of pairs or differences for each lag.

4 Proposed Feature Extraction Methods

Feature extraction is a crucial step in every pattern recognition system. These features should be compact and be able to differentiate between samples of different classes. As recent scientific researches show, FD algorithms produce very low dimensional feature vectors, often 1D or 2D, and are suitable to describe complicated patterns.

In order to enhance performance of OFR systems, FD methods, described in the previous section, are combined with directional algorithms which shown good results in OFR systems [5, 10].

4.1 Gaussian Directional Filter Bank-BCD

To achieve better results, in the proposed method, Gaussian directional filter bank is combined with BCD method. Gaussian filter bank produces directional sub-images and applying BCD method on each of them yields a new feature vector.

Directional range of 0 to π is quantized into 12 seeds. So, 12 directional sub-images were obtained each corresponds to a seed (some of them are shown in Fig. 5). Feature extraction stages are as follow:

1. Apply Gaussian directional filter bank on a text image.
2. Calculate the FD of each sub-image with BCD (as described in section 3.1).
3. Concatenate the extracted features to obtain 12D feature vector.

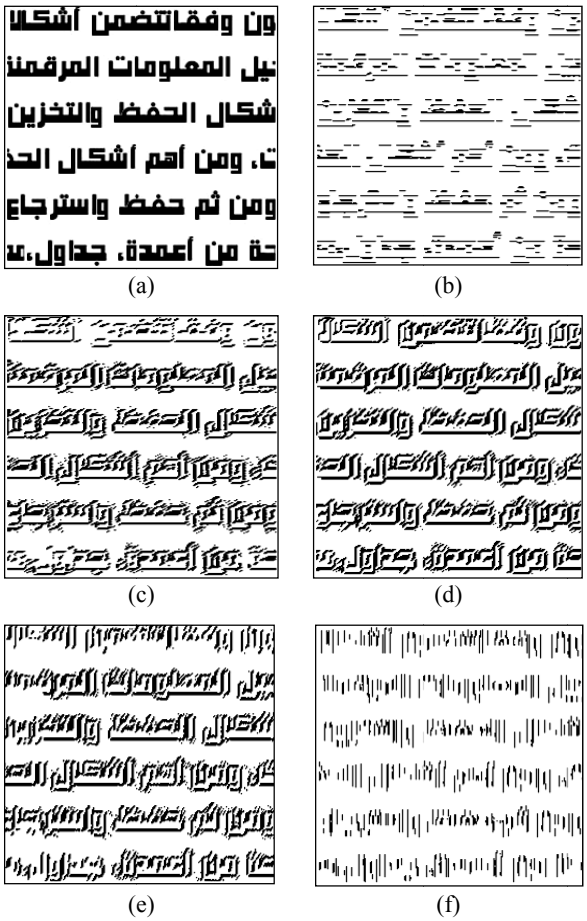

Fig. 5 (a) Original text image. Some Subimages: (b) 0°, (c) 30°, (d) 45°, (e) 60°, (f) 90°.

4.2 Extended DLA

Unlike many FD methods, dimension of feature vector obtained by diffusion limited aggregate can be more than two which is dependent to the range of q (see Eq.(4)).

Since we select $0 < q < 20$ for font recognition, the feature vector is 21D. The number of randomly selected pixels and repetition number respectively were chosen 200 and 100 in our computer simulations.

Among described FD methods in section 3, BCD and DLA can only be applied on binary images. Binarization method is an important matter for these FDs [18]. Binarization algorithms can be divided into two categories namely global and local (adaptive) approaches [19]. Our experiences show that adaptive approaches are better than global methods for OFR purpose, so we applied Niblack's binarization method in our simulations.

4.3 Radon-Higuchi

One of the most important differences between Higuchi and other FD methods is that, Higuchi is just applicable on a 1D signal. Since images are 2D signals, for extracting 1D directional signals from text images we utilized Radon transform.

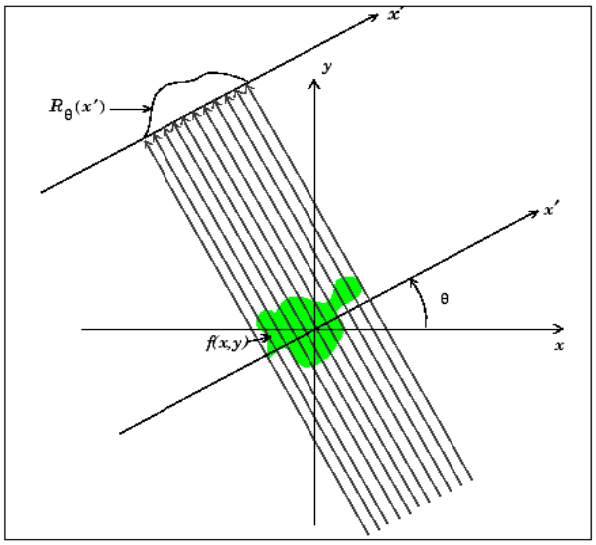

Fig. 6 Applying Radon transform on an image [Matlab 2010].

Radon transform projects a 2D image along a radial line oriented at a specific angle (Fig. 6). For each direction a 1D signal is obtained. Some of the 1D directional signals obtained from a text image are plotted in Fig. 7.

By applying Higuchi method on each obtained signal, one FD is calculated. In this case, length of feature vector depends on the number of directions.

Radon-Higuchi feature extraction algorithm is as follows:

1. Apply Radon transform on each text image in the specified directions. We quantized 0 to π interval into 16 directions.
2. Calculate FDs from obtained 1D signals via Higuchi method. In this case the obtained feature vector is 16D.

4.4 Two dimensional Variogram

Variogram is a directional FD in which the dimension of feature vector depends on the number of selected directions. We chose vertical, horizontal and diagonal directions for font recognition. Calculating these FDs according to Eq. (9) and Eq. (10) is as follows:

1. In vertical direction $Z_i \sim im(x, y)$ and $Z_{i+h} \sim im(x + h, y)$.
2. In horizontal direction $Z_i \sim im(x, y)$ and $Z_{i+h} \sim im(x, y + h)$.
3. In diagonal direction $Z_i \sim im(x, y)$ and $Z_{i+h} \sim im(x + h, y + h)$.
4. According to Eq. (9) for every h one $\gamma(h)$ is obtained, for estimating Variogram dimensions we utilized $h = 1,2,\ldots,6$.

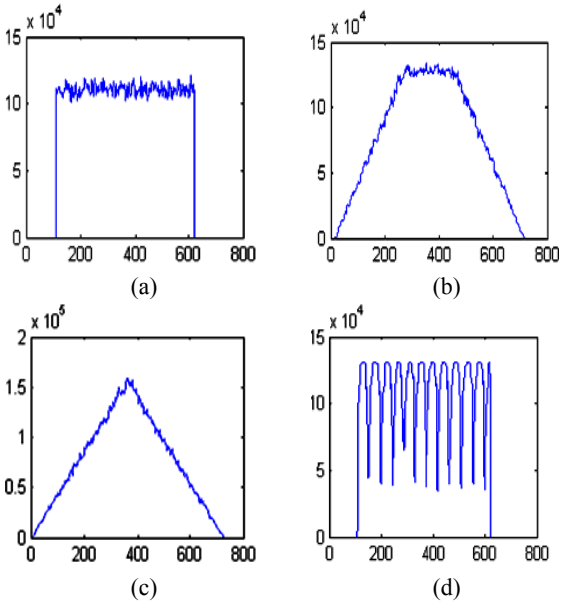

Fig. 7 Four 1D signals obtained from applying Radon transform through different angles on a text image: (a) 0°, (b) 30°, (c) 45°, (d) 90°.

5. A linear regression between log[h] and log[γ(h)] for each direction is produced. A graphical figure based on these two parameters is shown in Fig. 8.
6. For each direction, one slop and one intercept are produced.
7. For computing FDs through Eq. (10), $\log(\gamma(h))/\log(h)$ is replaced with above obtained slopes.
8. Moreover we used the obtained intercept of the linear regression as the second feature. Whereas 3 directional FDs were computed our feature vector is 6D.

4.5 Wavelet-Variogram

Wavelet transform is a suitable tool for font recognition [9]. Applying wavelet transform on an image decompose it into four components which are approximation (cA_1), and the details in horizontal (cH_1), vertical (cV_1), and diagonal (cD_1) directions. Our experiments show that combining these directional details with a directional FD algorithm such as Variogram enhance the results of font recognition. In this case, we applied Variogram FD on the obtained components from wavelet transform instead of applying on the original image:

1. Apply Wavelet transform on each text image.
2. Apply directional Variogram on the obtained directional components with the same direction. In the other words, vertical Variogram is applied on cV1, horizontal Variogram on cH1, diagonal Variogram on cD1, and vertical and horizontal Variogram on cA1 ssimultaneously.
3. Computing 4 FDs through Eq. (9) and Eq. (10) as mentioned in previous section.

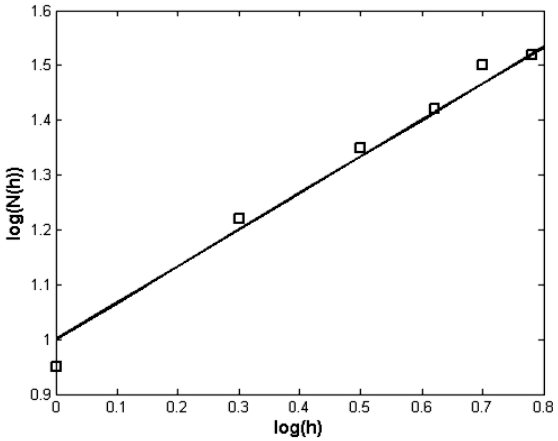

Fig. 8 Graphical figure based on log(h) and log(γ(h)).

In this case, the intercept of the linear regression is not utilized and the obtained feature vector is 4D. Combining wavelet and Variogram not only can decreases feature extraction dimension but also can improve the accuracy of discrimination.

4.6 Fourier Transform-Polar Histogram

In this section an innovative FD method is presented and it will be utilized for font recognition. All FD algorithms obey three stages [20]:

1. Select a measuring step such as $S_1, S_2, S_3, \ldots, S_N$.
2. Calculate a special quantity (function) based on measuring step as shown in Eq. (11) and Eq. (12).

$$F(S_1), F(S_2), F(S_3), \ldots, F(S_N) \tag{11}$$

$$\begin{array}{ccc} S_1 & \rightarrow & F(S_1) \\ S_2 & \rightarrow & F(S_2) \\ S_3 & \rightarrow & F(S_3) \\ \vdots & & \vdots \\ S_N & \rightarrow & F(S_N) \end{array} \tag{12}$$

3. Estimate FD based on the slope of the regression line between $log(F(S_i))$ versus $log(S_i)$ obtained from Eq. (12).

According to the above steps, we proposed a new FD method based on combination of Fourier Transform Coefficients and Polar Histogram (FTCPH) as follows:

1. Fourier transform is applied on the input image. For every pixel, one coefficient is obtained. Every coefficient has real (a) and imaginary (b) parts. Radius (R) and angle (θ) parts are calculated according to Eq. (13) from real and imaginary components.

$$\begin{cases} \theta = \tan^{-1}(\frac{b}{a}) \\ R = \sqrt{a^2 + b^2} \end{cases} \quad (13)$$

2. Quantize the whole area into n beans (see Fig. 9).
3. Quantize each bean into *k* radius $0 < R < \mathrm{R_{max}}$.
4. Build a polar histogram for the radiuses and angles obtained from step 1. The process is shown in Eq. (14) for the radius $R = R_n$.

Area	*Scale* $\rightarrow \theta$ *(number of bean)*	*histogram*
$\theta_1 < \theta < \theta_2$	1	h_{n1}
$\theta_2 < \theta < \theta_3$	2	h_{n2}
$\theta_3 < \theta < \theta_4$	3	h_{n3}
$\vdots$	$\vdots$	$\vdots$
$\theta_m < \theta < \theta_{m+1}$	m	h_{nm}

(14)

In the other words, for each radius located between two quantized values $R_n < R < R_{n+1}$ and each phase located between two quantized values $\theta_m < \theta < \theta_{m+1}$, the nth row and mth column is increased one unit.

5. FTCPH FD is obtained through Eq. (15).

$$D = \frac{log\ (log(h_{nm}))}{log(m)} \quad (15)$$

where $\mathrm{h_{nm}}$ is polar histogram of nth radius and m is number of beans. We examined various amounts of n and k and the best results obtained for $n = 32$ and $k = 6$.

Using described algorithm for each R_n(row) there are 32 numbers (columns) which will be utilized for FD calculations. For example, D_1 is estimated through the slop of a linear regression between $log[log\ (h_{1i})]$ and $log[i]$ as shown in Eq. (16).

Area	*Scale* $\rightarrow \theta$	*histogram*
$\theta_1 < \theta < \theta_2$	1	$h_{1,1}$
$\theta_2 < \theta < \theta_3$	2	$h_{1,2}$
$\theta_3 < \theta < \theta_4$	3	$h_{1,3}$
$\vdots$	$\vdots$	$\vdots$
$\theta_{32} < \theta < \theta_1$	32	$h_{1,32}$

(16)

$D_2, D_3, \cdots, D_n$ can be estimated with the same algorithm. The FTCPH feature vector is 6D.

5 Data Sets and Data Reconstruction

Most of the previous OFR algorithms recognize font type in a text block, consisting several text lines. Based to our knowledge, the only line level OFR algorithm and also the only existent Farsi OFR dataset was created by H. Khosravi et al [5]. Since our proposed algorithms also deal with font recognition in text blocks, Khosravi's dataset cannot be used directly in this paper. Since Farsi and Arabic languages have common alphabets, we utilized ALPH-REGIM dataset which S. B. Moussa et al utilized in their paper [11]. Moreover, Latin ALPH-REGIM dataset was used to compare the results with previous Latin OFR algorithms.

Since size of primary text blocks in these two datasets were different, for achieving better results, we build 512×512 text blocks from them. The utilized reconstruction algorithm is similar to that described by H. Khosravi et al in [5].

These steps are as follows:

- Find all lines in the input text image and separate them.
- Align separated lines in a straight arrangement.
- Segment the obtained lines into 512 pixel width.
- Concatenate the segmented lines vertically to construct 512×512 image blocks.

Due to lack of enough space, some samples with the size of 128×128 text block obtained from Fig. 10 and Fig. 12 are respectively shown in Fig. 11 and Fig. 13.

6 Experimental Results

In this section the performance of some recent published font recognition systems, [5, 10, 11], are compared with the proposed methods. The comparison includes length of feature vector, speed, robustness, recognition rate and etc.

We used fractal dimension feature extraction methods, and RBF and KNN classifiers for font recognition. All experiments are performed on ALPH-REGIM Arabic and English datasets. After reconstructing database (as explained in section 5) we used $\frac{2}{3}$ samples for training and the rest for testing.

6.1 Feature Vector Length

High-dimensional feature set is one of the problems that many pattern recognition algorithms suffer from it. Such redundant feature vectors make the identification

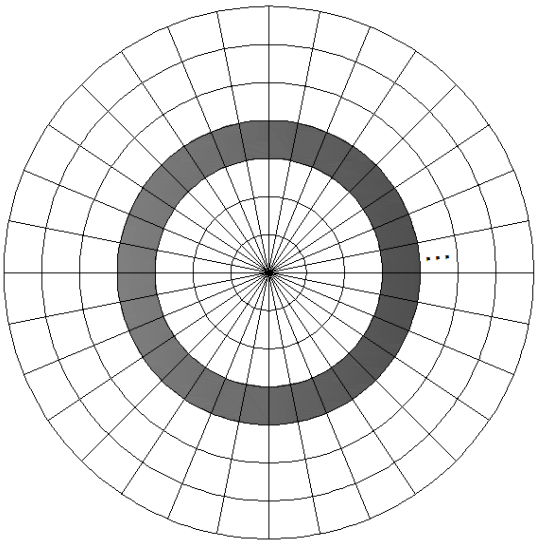

Fig. 9 The polar histogram diagram.

process more complicated. The most important problem with high-dimensional feature vector is that, in many cases, not all the measured variables are important for recognition phenomena. In these situations, dimensionality reduction algorithms are essential. One of the most important advantages of FD algorithms is their ability to produce concise feature vectors. Table 1 compares the proposed feature vector's lengths.

As mentioned in section 4, in order to enhance performance of font recognition system, these fractal based methods are combined with some other algorithms.

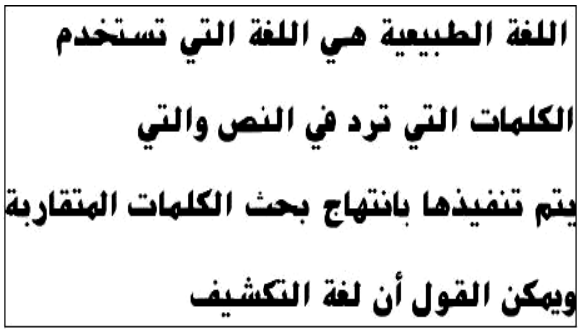

Fig. 10 An Arabic text image with several lines.

Fig. 11 Three 128×128 text images obtained from Fig 10.

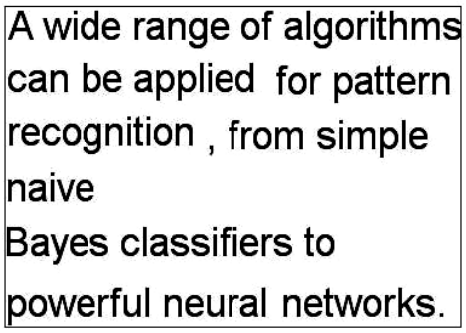

Fig. 12 An English text image with several ines.

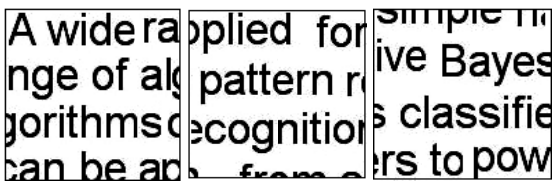

Fig. 13 Three 128×128 English text images obtained from Fig 12.

As shown in Table 1, the length of feature vectors in all the proposed algorithms don't exceed from 21D, while SRF feature vector length in [5] is 512D. Moreover Table 1 shows that the shortest feature vector lengths are related to BCD-DCD [11] and Wavelet-Variogram proposed algorithms which are 4D.

6.2 Discrimination Rates

Discrimination rate is one of the most important criterions in every pattern recognition algorithm. The obtained discrimination rates are presented in Tables 2 and 3 for Farsi/Arabic and English scripts, respectively. Maximum recognition rates belong to Wavelet-Variogram method for both Farsi/Arabic and English fonts which are 99.14% and 100%, respectively.

It should be noted that comparing the obtained recognition rates with the reported one [5] for SRF feature is not fair. Because utilized datasets are not the same as mentioned in section 5. However, in other cases, the used datasets are the same and comparison is meaningful.

Table 1 Feature vector length of different FDs.

Technique	Length
SRF [5]	512
BCD-DCD [11]	4
Gabor [10]	32
Directional Filter Bank-BCD	12
DLA	21
Radon-Higuchi	16
Variogram	6
Wavelet-Variogram	4
FTCPH	6

Table 2 Recognition rates of different Methods for Farsi/Arabic fonts recognition.

Technique	Classifier	Recognition Rates [%]
SRF [5]	MLP	94.16
BCD-DCD [11]	KNN	96.2
	RBF	98
Directional Filter Bank-BCD	KNN	93.56
	RBF	96.97
DLA	KNN	82.35
	RBF	83.86
Radon-Higuchi	KNN	98.86
	RBF	98.12
Variogram	KNN	98.84
	RBF	96.9
Wavelet-Variogram	KNN	98.91
	RBF	99.14
FTCPH	KNN	97.69
	RBF	98.41

Table 3 Recognition rates of different Methods for English fonts recognition.

Technique	Classifier	Recognition Rates [%]
Gabor [10]	WED	99.1
Directional Filter Bank-BCD	RBF	98.49
	KNN	98.03
DLA	RBF	88.97
	KNN	89.14
Radon-Higuchi	RBF	99.68
	KNN	99.52
Variogram	RBF	98.21
	KNN	99.6
Wavelet-Variogram	RBF	100
	KNN	100
FTCPH	RBF	99.75
	KNN	99.46

6.3 Computational Time

Speed of OFR is in the secondary importance because a time consuming OFR decreases the overall speed of OCR system. Except H. Khosravi et al who reported speed issue [5], others neglected this important topic. There is a comparative study for the average computational time in each OFR algorithm in Table 4. These computational times are for feature extraction from 512×512 text images. SRF [5] is the fastest OFR algorithm and among our proposed methods, Variogram and Wavelet-Variogram have the least computational time. Our experimental results show that SRF algorithm is about 2 times faster than the mentioned algorithms.

DLA and BCD-DCD are the most time consuming algorithms mainly due to applying an adaptive binarization algorithms during their process.

6.4 Robustness Test

Generally, recognition rate degrades with unavoidable distortions which are prevalent in real applications. In this section robustness of the proposed algorithms against skew and noise are studied and compared with [5] and [11].

6.4.1 Skew

Document skew is a distortion that every OCR system may encounter. Lack of robustness against skew may severely influences the system's performance. Nowadays skew correction algorithms become an integral part in OCR systems.

Most of the FD algorithms are robust against skew. Rotating objects does not influence their Euclidean dimension. This rule is true in most of the FD methods except directional FDs. All utilized FDs except Variogram are skew independent.

Similar to [11], the font recognition system is tested with five rotations presented in Fig. 14.

Results in Table 5 confirm that among proposed approaches DLA and FTCPH are more robust against skew. The average errors in these cases are 0.19 % and 1.51% respectively. BCD-DCD algorithm is also robust against skew and the average error rate is about 0.86% [11]. So, DLA is the most robust algorithm agains skew distortion.

According to previous definitions, DLA, BCD, DCD and FTCPH algorithms are related to some counting information which are relatively independent of object direction.

Fig. 15 shows some comparative study among the average recognition rates before and after applying skew. High errors are due to combination of FDs with directional algorithms. Although SRF [5] and Gabor [10] features are not robust against skew, the skew effect was ignored in them.

6.4.2 Noise

Performance of every recognition system degrades in the presence of environmental noise.

To verify the performance of the proposed algorithms, a range of noise from SNR = 20 to SNR = 50 was added to all data set (Fig. 16). According to our experimental results, most of the proposed methods are robust to noise when noise intensity is low. According to Fig. 17, Directional Filter Bank and Radon-Higuchi are the most robust algorithms against noise. In fact directional decomposing algorithms works as a filter and decrease the propagation of noise to the next stages. Table 6 shows the skew robustness of different algorithms.

Table 4 Time of different feature extraction methods.

Technique	Time [Sec]
SRF [5]	0.06
BCD-DCD [11]	40.47
Gabor [10]	2.848
Directional Filter Bank-BCD	7.196
DLA	45.638
Radon-Higuchi	0.538
Variogram	0.159
Wavelet-Variogram	0.138
FTCPH	4.706

Table 5 Recognition error for skewed images.

Methods	Average Recognition Error	Skew Robustness
BCD-DCD [11]	% 0.86	Yes
Directional Filter Bank-BCD	% 60.43	No
DLA	% 0.19	Yes
Radon-Higuchi	% 73.71	No
Variogram	% 77.1	No
Wavelet-Variogram	% 85.62	No
FTCPH	% 1.51	Yes

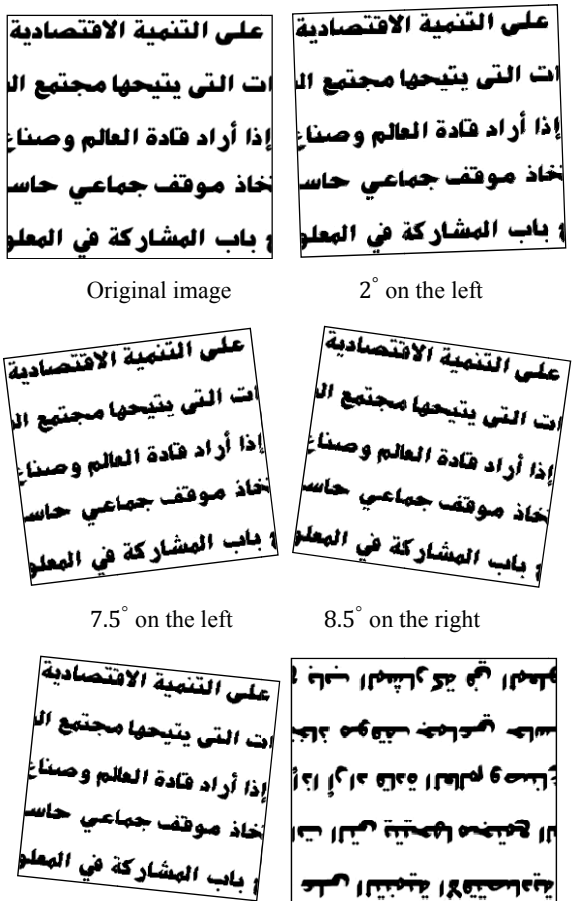

Fig. 14 Some skew and rotation effects applied on text images.

Table 6 Comparative study among the skwe robustness.

Technique	Dirctional / Nondirectional	Skew Robustness
SRF [5]	Dirctional Algorithm	No
BCD-DCD[11]	Nondirctional Algorithms	Yes
Gabor [10]	Dirctional Algorithm	No
Directional Filter Bank-BCD	Dirctional Algorithm (Combination of nondirctional FD with a dirctional algorithm)	No
DLA	Nondirctional Algorithm (FD)	Yes
Radon-Higuchi	Dirctional Algorithm (Combination of nondirctional FD with a dirctional algorithm)	No
Variogram	Dirctional Algorithm (Dirctional FD)	No
Wavelet-Variogram	Dirctional Algorithm (Combination of dirctional FD with a dirctional algorithm)	No
FTCPH	Nondirctional Algorithm	Yes

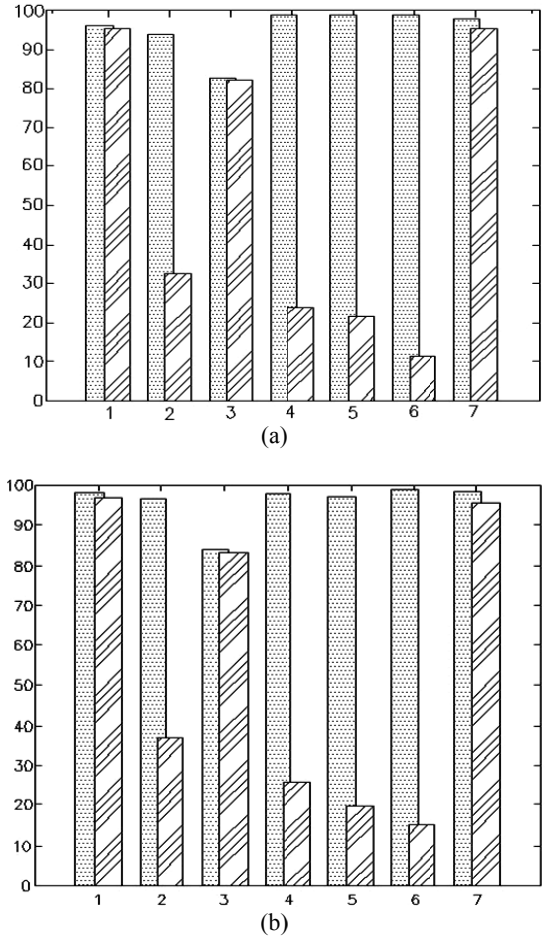

Fig. 15 Comparative study among the algorithms robustness against skew, (a):KNN (b):RBF. 1: BCD-DCD, 2: Directional Filter Bank-BCD, 3: DLA, 4: Radon-Higuchi, 5: Variogram, 6: Wavelet-Variogram, 7: FTCPH.

Fig.16 Example of noising images with various values of SNR: (a) SNR=50, (b) SNR=40, (c) SNR=30, (d) SNR=20.

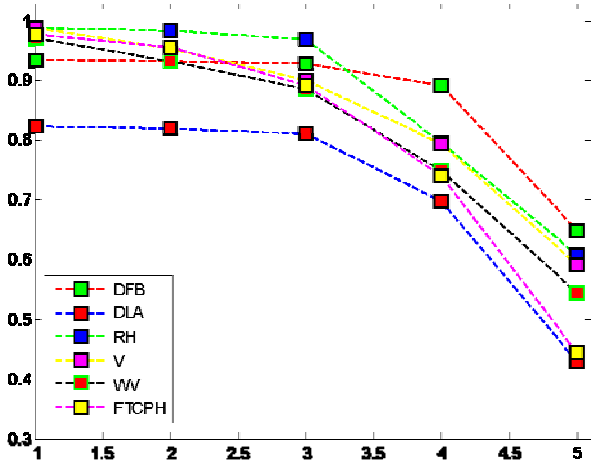

Fig. 17 Recognition rates of different FDs in the presence of noise. DFB:Directional Filter Bank-BCD, RH:Radon-Higuchi, V:Variogram, V:Wavelet-Variogram.

7 Conclusion

In this paper, we proposed some new fractal dimensional feature extraction methods for font identification. Our experimental results confirmed the previous findings about FDs ability for font discrimination. The most important advantages of the proposed feature extraction methods are:

- Low dimensionality.
- Low computational complexity.
- Being content-independent, which makes it easier to adapt for other applications.
- Being robust against document skew and rotation in some cases.
- Being applicable to other languages font recognition systems.

According to our experiments, minimum feature vector length, minimum runtime and maximum recognition rates belong to the proposed Wavelet-Variogram algorithm. The only drawback of this feature is lack of robustness against skew. Among skew robust algorithms, FTCPH is the fastest algorithm which has the highest recognition rate.

References

[1] M. Shams Esfand Abadi and S. Nikbakht, "Image Denoising with Two-Dimensional Adaptive Filter Algorithms", *Iranian Journal of Electrical & Electronic Engineering*, Vol. 7, No. 2, pp. 84-105, June. 2011.

[2] S. Mohammadi, S. Talebi and A. Hakimi, "Two Novel Two Novel Chaos-Based Algorithms for Image and Video Watermarking", *Iranian Journal of Electrical & Electronic Engineering*, Vol. 8, No. 2, pp. 97-107, June. 2012.

[3] E. Shabani Nia and Sh. Kasaei, "Moving Vehicle Tracking Using Disjoint-View Multicameras", *Iranian Journal of Electrical & Electronic Engineering*, Vol. 7, No. 3, pp. 168-178, Sep. 2011.

[4] R. Ebrahimpour, S. Sarhangi and F. Sharifizadeh, "Mixture of Experts for Persian Handwritten Word Recognition", *Iranian Journal of Electrical & Electronic Engineering*, Vol. 7, No. 4, pp. 217-224, Dec. 2011.

[5] H. Khosravi and E. Kabir "Farsi font recognition based on Sobel–Roberts features", *Elsevier, Pattern Recognition Letters*, Vol. 31, No. 1, pp. 75-82, Janu. 2010.

[6] BB. Chaudhuri and U. Garain, "Automatic detection of italic, bold and all-capital words in document images", *14th International Conf. on Pattern Recognition*. Vol. 1, No. 1, pp. 610-612, Aug. 1998.

[7] A. Zramdini and R. Ingold, "Optical font recognition using typographical features", *IEEE Transaction on Pattern analysis and Machine Intelligence*. Vol. 20, No. 8, pp. 877-882, Aug. 1998.

[8] C. B. Jeong, H. K. Kwag, S. H. Kim, J. S. Kim and S. C. Park , "Identification of font styles and typefaces in printed Korean documents", *ICADL*, Vol. 2911, No. 1, pp. 666-669, July. 2003.

[9] X. Ding, L. Chen and T. Wu, "Character Independent Font Recognition on a Single Chinese Character", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 29, No. 2, pp. 195-204, Feb. 2007.

[10] Y. Zhu, T. Tan and Y. Wang, "Font Recognition Based on Global Texture Analysis", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 23, No. 10, pp. 1192-1200, Sep. 2002.

[11] S. B. Moussa, A. Zahour, A. Benabdelhafid and A. Alimi , "New features using fractal multi-dimensions for generalized Arabic font recognition", *Elsevier Pattern Recognition*, Vol. 31, No. 5, pp. 361-371, April. 2010.

[12] O. Shenker, "Fractal geometry is not the geometry of nature", *Elsevier Studies In History and Philosophy of Science Part A*, Vol. 25, No. 6, pp. 967-981, Dec. 1994.

[13] Z. Yang, L. Yang, D. Qic and C. Suene, "An EMD-based recognition method for Chinese fonts and styles", *Elsevier Pattern Recognition*, Vol. 27, No. 14, pp. 1692-1701, Oct. 2006.

[14] A. Chaabouni, H. Boubaker, M. Kherallah, A. Alimi and H. Abed, "Fractal and Multi-Fractal for Arabic Offline Writer Identification", *IEEE Trans International Conference on Pattern* Recognition, Vol. 1, No. 1, pp. 3793-3796, Aug. 2010.

[15] R. Valdez and D. Hern Andez-Ram Irez, "Fractality of Monthly Extreme Minimum Temperature", *Fractals*, Vol. 11, No. 2, pp.269-273, June 2003.

[16] T. Babadagli and K. Develi, "Fractal characteristics of rocks fractured under tension",

Elsevier, Theoretical and Applied Fracture Mechanics, Vol. 39, No. 1, pp. 73-88, Feb. 2003.

[17] A. Hajian Nezhad and S. Mozaffari "Font Recognition Using Variogram Fractal Dimension", *ICEE Conference*, Vol. 1, No. 1, pp. 634-639, May. 2012.

[18] A. Hajian Nezhad and S. Mozaffari "Fractal and Multi-Fractal Dimensions For Farsi/Arabic Font Type and Size Recognition", *IEEE The 7th MVIP Conference*, Vol. 1, No. 1, pp. 1-4, Nov. 2011.

[19] M. Valizadeh and E. Kabir, "Binarization of degraded document image based on feature space partitioning and classification", *International Journal on Document Analysis and Recognition (IJDAR)*, Vol. 15, No. 1, pp. 57-69, March. 2012.

[20] R. Lopes and N. Betrouni, "Fractal and multifractal analysis: A review", *Elsevier, Medical Image Analysis*, Vol. 13, No. 4, pp. 634-649, Aug. 2009.

Akram-Alsadat Hajian Nezhad received the B.Sc. degree in Electrical Engineering (Electronics) from Babol University of Technology, Babol, Iran, in 2007 and the M.Sc. degree in Electrical Engineering (Electronics) from Semnan University, Semnan, Iran, in 2012. Her research interests include Image Processing, Pattern Recognition, Medical Imaging Processing and Neural Networks.

Saeed Mozaffari received his B.Sc., M.Sc and Ph.D. degrees in Electronic Engineering from Amirkabir University of Technology, Tehran, Iran. Since 2006, he is a faculty member in Electrical and Computer Department of Semnan University. His research interests include digital image processing, computer vision, and Pattern Recognition.