Watermarking in Farsi/Arabic binary document images using fractal coding

Fatemeh Daraee
Electrical and Computer Department
Semnan University
Semnan, Iran
daraee@semnan.ac.ir

Saeed Mozaffari
Electrical and Computer Department
Semnan University
Semnan, Iran
mozaffari@semnan.ac.ir

Abstract—This paper presents a novel watermarking method based on fractal theory. In the proposed method, information is embedded into binary document images. First, host image is coded by the proposed fractal coding method which is designed particularly for binary images. To insert the watermark uniformly over the entire host image, only some of Range segments with certain conditions are selected. Then, the watermark is added to the number of ones in the fractal code of selected Range segments. Finally, the output image is obtained by the fractal decoding procedure. Experimental results show that the output image quality of the proposed methods is acceptable to human eyes. In addition, empirical results show that the proposed fractal based watermarking is robust to the common attacks.

Keywords—fractal image coding; binary document image; watermarking; information hiding.

I. INTRODUCTION

As the internet becomes ubiquitous as a fast and easy environment for information transmission, individuals can shares their resources such as text, audio, images, videos and etc. In addition to numerous advantages, such information transmission threatens to ruin ownership and copyright protection [1]. Watermarking has emerged as a tool to covert communication, Broadcast monitoring, Proof of ownership and etc.

Regarding the ways of information hiding, watermarking methods can be categorized into two main groups, namely spatial domain and transform domain [2]. Figure 1 shows these two main methods and their subsequent approaches. In spatial domain methods, gray level values of the host image pixels are manipulated and information is inserted directly into it. Although spatial domain methods can be easily implemented and have high capacity, they are vulnerable to different kinds of attacks such as cropping, filtering and compression. On the other hand, transform domain methods can tolerate noise and attacks, and have good-looking output. In the transform domain methods, first, the input image is converted by a predefined transformation. Then, the watermark is embedded in the output image or in the transform coefficient. Finally, the inverse transform is performed. Since the watermark is distributed over the whole pixels of the host image, rather than local parts, transform domain methods are more robust to attacks.

Over the past decade, fractal coding has mainly been exploited for image compression. Recently, it is utilized for different tasks such as pattern recognition, watermarking and etc. Based on the fractal codes, several watermarking techniques have been proposed [1]. Most of the previous efforts focused on gray level image watermarking in which the watermark (binary image/code) is housed into a host image with gray level values.

In this paper, we proposed a new method for information hiding in the binary images based on fractal coding. For this purpose, a novel binary fractal coding with high quality output, suits for binary documents, was developed. After coding the host image with the proposed binary fractal coder, binary sequence of the watermark is inserted into fractal codes. By decoding the obtained fractal codes, watermark information is spread globally over the host image.

The main reason behind choosing binary host image, rather than grayscale, is its prevalence and compactness. Among different types of information such as text, video, and audio, documents are the most common format in our daily life. Documents are usually converted from grayscale into binary format for archiving in the databases and distributing over the internet.

The main challenges of watermarking of binary images are sensitivity and capacity issues. Changing a pixel’s value from 0 to 1 or vice versa means 50% alteration. Moreover, characters and digits are almost familiar to human. So, small changes in binary document images can be easily recognized by human’s eye. Therefore, information hiding capacity of binary images is much less than gray level images.

Best to our knowledge, few researches have been focused on binary fractal coding. In [3], a smoothness class is defined for each block in the binary image. If the smoothness class of a range-domain pair is similar to a predefined parameter, μ, then these blocks can be similar to each other. So, it suffices to compare smoothness class values of image blocks under analysis. This strategy restricts the domain blocks pool and accelerates the coding process. Range blocks of three distinct types are singled out at the encoding stage, namely: absolutely black, absolutely white and non-monochrome (mixed). After finding the best match for each
range and domain blocks, four parameters called \( o, t, f, \) and \( e \) are calculated. Where \( e \) and \( f \) specify coordinates of the domain block \( D \) for a current range block \( R \), \( t \) determines the type of the transformation, and parameter \( o \) characterizes the type of the range block \( R \). This fractal coding is suitable for binary silhouette images which have low pixel's value distribution.

The rest of the paper is organized as follows. Section 2 describes the baseline fractal coding originally developed for gray level images. The proposed fractal coding for binary image presented in section 3. Sections 4 and 5 explain watermark insertion and detection, respectively. Experimental results are exhibited in section 6. Finally, conclusion remarks are presented in section 7.

II. BASELINE FRACTAL CODING METHOD

Fractal image coding is based on self-similar sets and Iterated Function System (IFS) [4]. It is inspired from the fact that our natural environment generally shows self similarity on different scales and has considerable amount of redundancy. By the means IFS, there would be a contractive transformation for each image that has the fixed-point resemble to the original image. In other words, applying that transform \( (T) \) iteratively on an arbitrary starting image, the result will converge to the original image. In practice, after less than 10 iterations, the fixed-point image can be obtained. The following equations describe IFS:

\[
I_{n+1} = T(I) \\
I = \lim_{n \to \infty} T(I) \\
T(I) = \bigcup_{i=1}^{n} T(I)
\]

For fractal coding an image with the size of \( 2m \times 2n \), the entire image is first partitioned into \( N \) blocks, where \( N = (2m/2b) \times (2n/2b) = 2m+n-2b \). These non-overlapping square are called range blocks having size of \( 2b \times 2b \). A domain block pool is then obtained from the original image by sliding a window of size \( 2b+1 \times 2b+1 \) within the image, starting at the top left corner of the image, in step-size of \( \delta \), along the horizontal or vertical direction. Figure 2 shows rang and domain blocks in the input image. For each range block, we search the domain block pool to find the most similar domain block and an affine transformation \( \tau \) which relates these blocks.

Finally, the fractal coding can be formulated as:

\[
R^{(o)} = s \times \sigma(D^{(n-1)}) + gU
\]

Where \( s \) and \( g \) are the contrast scaling and luminous offset, respectively. \( \sigma(.) \) is a contractive operator to shrink the domain blocks with the size of \( 2b+1 \times 2b+1 \) into the same size as the \( 2b \times 2b \) range blocks. \( U \) is a unit matrix.

III. THE PROPOSED METHOD FOR BINARY IMAGE FRACTAL CODING

Fractal coding of gray level images is a time consuming task. Because large numbers of sequential search through a list of domains are needed to find the best match for a given range block. After finding the best match for each range block, several parameters should be determined according to equation 1.

On the other hand, fractal coding of binary images is fast enough. This is mainly due to limited numbers of pixel’s values in the range and domain blocks. Furthermore, some of parameters in the fractal code, such as contrast scaling and luminous offset, are not meaningful for binary images. By the use of look-up table and discarding redundant parameters, coding process can be speed up.

As mentioned before, binary fractal coder, presented in [3], is only appropriate for those binary images with monolithic regions. To deal with binary document images which have several black regions (text) over a white region (background), a new fractal coding is needed. Moreover, the binary fractal coder in [3] cannot be utilized for watermarking applications. Because, only the type of the range block, coordinates of the domain block and type of the transformation are recorded in the fractal code. So, the watermark can only be inserted into the type of the transformation which is rotated 90, rotated 180, rotated 270, reflected diagonally, reflected anti-diagonally, reflected horizontally, and reflected vertically. It is obvious that changing the type of the transformation result an undesirable output image.

This paper proposes a new fractal coding method which can be used for general binary images, such as binary documents. This method is also capable to be employed for watermarking tasks. In the following, details of coding and decoding are presented.

---

Figure 1. Review of watermarking methods.

Figure 2. Range and domain blocks representation.
A. Coding

In this paper, instead of using two-dimensional range blocks of size $2^b \times 2^b$, range segments of size $1 \times 2^b$ are utilized. Block diagram of the overall fractal coder is shown in Figure 3. Each row of the input image is divided into range segments of size $1 \times 2^b$ ($b=2$ in this paper). To make the domain pool, rows of the input image are concatenated to form domain segments with the size of $1 \times 2^{b+1}$. Each range segment can be categorized into three groups: absolutely black, absolutely white, and non-monochrome. Absolutely black, absolutely white, and non-monochrome segments are represented by 0, 1 and 2 in the fractal code. These blocks are not considered in the fractal coding process. For each range segment of non-monochrome type, domain segments of non-monochrome type are searched to find the most similar one (domain segments of absolutely black and absolutely white are discarded).

Block diagram of the overall fractal coder is shown in Figure 3. Each row of the input image is divided into range segments of size $1 \times 2^b$ ($b=2$ in this paper). To make the domain pool, rows of the input image are concatenated to form domain segments with the size of $1 \times 2^{b+1}$. Each range segment can be categorized into three groups: absolutely black, absolutely white, and non-monochrome. Absolutely black, absolutely white, and non-monochrome segments are represented by 0, 1 and 2 in the fractal code. These blocks are not considered in the fractal coding process. For each range segment of non-monochrome type, domain segments of non-monochrome type are searched to find the most similar one (domain segments of absolutely black and absolutely white are discarded).

$$
\text{Hamming distance is used to find best range-domain segment pairs. In binary images, Hamming distance shows the number of dissimilar pixels. This distance can be easily obtained by XOR of range and domain segment’s pixels. After performing the above process for each row of the input image, the fractal code is stored as follows:}
$$
\{NR, R1, D\} = \min\{\text{number of ones } (RXOR \sigma(D))\} \quad (5)

where \(NR\) is the range segment type, \(R1\) indicates number of 1s in the range segment, and \(D\) is index of corresponding domain segment for current range block \(R\).

B. Decoding

Having fractal codes and an arbitrary initial image, the original input image can be reconstructed by the decoding stage. First, range segment type \((NR)\) is extracted from fractal code. If \(NR=0\) or \(NR=1\), then a range segment with the size of $1 \times 2^b$ containing zeros and ones is built, respectively. But, for \(NR=2\), a range segment with \(R1\) (the number of 1s in the range segment) ones is produced. For instance, if \(R1=3\), four range segment candidates are considered: 0111, 1011, 1101, 1110. The most resemble candidate is replaced by the domain segment at \(D\) position (index of best domain segment) is compared with each of the four candidates via hamming distance criterion. The most resemble candidate is replaced by the domain segment at \(D\) position. Performing the above process for all range segments, one iteration is completed. Repeating the decoding process for several iterations, the algorithm converges and the original input image is achieved. Figure 4 shows the decoding results after 1st, 2nd, and 5th iterations. According to Figure 4, the decoded image is constant after three iterations.

Figure 3. Flowchart of the proposed fractal image coding method.

Figure 4. Decoded image after one, two and five iterations.

IV. WATERMARK INSERTION

For watermark insertion, first the host image is coded by the proposed fractal coder. The watermark can be either a
binary string or a binary image. In this paper, the watermark is IEEE logo which is added to the number of ones in the range segment. Range segments have size of $1 \times 4$. To embed the watermark uniformly over the host image, only range segments having 1 or 2 ones are selected. Therefore, the capacity of the host image equals the number of such blocks.

Absolutely black range segments are not suitable for watermark insertion. Because, changing pixels of such a range segment would be obvious. On the other hand, absolutely white range segments do not have any capacity to house the watermark.

Inserting the watermark $w$ into the host image fractal code $\{NR, R1, D\}$, yields the $\{NR, R1+w, D\}$ as the watermarked image fractal code.

V. WATERMARK DETECTION

To detect the watermark, both host image and watermarked image are needed. First, range segments of both host and watermarked images are constructed. After subtracting corresponding range segments in host and watermarked images, the watermark can be easily extracted without the need of having fractal codes.

VI. EXPERIMENTAL RESULTS

To evaluate performance of the proposed fractal based watermarking approach, some experiments are conducted. First, the visualization issue is assessed. Information hiding methods should accommodate the watermark in an intangible manner. Figure 5 shows the watermarked output image.

Another important concern in watermark methods is their resistance to noise and attacks. During transformation, the watermarked image may be affected by noise. In addition to random changes, such as noise, the watermarked image may be deliberately altered by attacks. Attacks are intended to destroy the watermark by applying operations like cropping, compression and etc. Figures 6, 7 and 8 demonstrate the robustness of the proposed method.

VII. CONCLUSION REMARKS

This paper presents a new fractal coding method for binary images. Experimental results show that our method is able to reconstruct the original input image with high quality even after 3 iterations. The new fractal coding method is also utilized for watermarking. Binary sequence is added to the number of ones in the fractal code of those range segments with 1 or 2 ones. The robustness of proposed method is tested under different attacks and the results were promising.
REFERENCES


