



ISSN: 0067-2904

## Fractal Image Compression Using Block Indexing Technique: A Review

Zainab J. Ahmed<sup>1\*</sup>, Dr. Loay E. George<sup>2</sup>, Zinah S. Abduljabbar<sup>3</sup>

<sup>1</sup>Department of Biology Science, College of Science, University of Baghdad, Baghdad, Iraq

<sup>2</sup>University of Information Technology and Communications, Baghdad, Iraq

<sup>3</sup>Computer Science Department, College of Science, Mustansiriyah University, Baghdad, Iraq

Received:7/1/2020

Accepted: 15/3/2020

### Abstract

Fractal image compression depends on representing an image using affine transformations. The main concern for researches in the discipline of fractal image compression (FIC) algorithm is to decrease encoding time needed to compress image data. The basic technique is that each portion of the image is similar to other portions of the same image. In this process, there are many models that were developed. The presence of fractals was initially noticed and handled using Iterated Function System (IFS); that is used for encoding images. In this paper, a review of fractal image compression is discussed with its variants along with other techniques. A summarized review of contributions is achieved to determine the fulfillment of fractal image compression, specifically for the block indexing methods based on the moment descriptor. Block indexing method depends on classifying the domain and range blocks using moments to generate an invariant descriptor that reduces the long encoding time. A comparison is performed between the blocked indexing technology and other fractal image techniques to determine the importance of block indexing in saving encoding time and achieving better compression ratio while maintaining image quality on Lena image.

**Keywords:** Fractal, Compression, Iterated Function System, Encoding, Block Indexing, Moment Descriptor.

### ضغط الصورة باستخدام الكسوريات بالاعتماد على تقنية ترميز المقاطع: دراسة مراجعة

زينب جواد احمد<sup>1\*</sup>، لؤي ادور جورج<sup>2</sup>، زينة صادق عبد الجبار<sup>3</sup>

<sup>1</sup>قسم علوم الحياة، كلية العلوم، جامعة بغداد، بغداد، العراق.

<sup>2</sup>جامعة تكنولوجيا المعلومات والاتصالات، جامعة بغداد، بغداد، العراق.

<sup>3</sup>قسم علوم الحاسوب، كلية العلوم، جامعة المستنصرية، بغداد، العراق.

### الخلاصة:

ضغط الصورة باستخدام الكسوريات هي طريقة ضغط مناسبة تعتمد على تمثيل صورة باستخدام تحويلات الاقايين affine. الاهتمام الرئيسي للبحوث المنشورة حديثاً في مجال استخدام طريقة "ضغط الصور باستخدام الكسوريات (FIC)" يتركز على تقليل الوقت المستغرق لترميز (ضغط) بيانات الصورة. ان الفكرة الأساسية تتركز على كون جميع أجزاء الصورة تماثل بشكل متكرر (والى حد ما) بعض الأجزاء الأخرى من نفس الصورة. وفي هذه العملية، هناك العديد من النماذج المتقدمة لهذا النوع من الضغط. تم ملاحظة وجود كسوريات في البداية والتعامل معها باستخدام نظام وظيفة تكرارات الذي يرمز له (IFS)؛ التي تستخدم لترميز الصور. في هذه

\*Email: zainabjahmed83@gmail.com

الدراسة ، تم استعراض العديد من نماذج ضغط صورة باستخدام الكسوريات. وكذلك تم إجراء مراجعة شاملة للمساهمات في هذا المجال لتحديد مدى كفاءة تحقيق ضغط للصورة باستخدام مفهوم الكسوريات، وقد تم التركيز على طريقة ترميز المقاطع باستخدام واصفات العزوم. حيث تعتمد هذه الطريقة على تصنيف مقاطع الصور domain.range (النطاق والمدى) باستخدام واصفات العزوم لتصنيف اجزاء الصورة، الامر الذي بدوره سيسهم في تقليل وقت الترميز الى حد كبير حيث انه طويل نسبياً. تم إجراء مقارنة بين تقنية ترميز المقاطع وغيرها من تقنيات الاخرى لضغط الصورة باستخدام الكسوريات لتحديد أيهما أفضل في نسبة الضغط (CR) وجودة الصورة المتكونة ووقت الترميز على صورة لينا (Lena).

## Introduction

In this section, an introduction is provided to data compression and the essential concept behind fractal image compression in order to clarify the most important methods involved.

### A. Data Compression

To minimize the number of bits wanted to show an image, a video sequence or music, the data compression algorithms are applied. It is the science or technique to represent the information in a compact structure. Data compression is necessary because the users produce and utilize more and more of the information that is in digital form. The data compression function consists of two components: (i) an encoding algorithm that takes an image and produces a "compressed" representation, and (ii) a decoding algorithm that rebuilds, from the compressed representation, the original message or some approximation of it [1]. A variety of data compression methods are used to compact images including the lossy and lossless methods. In the lossy compression, the compressed signal does not look like the original actual signal and there are disfigurements of the actual signal with the decompressed signal. The lossless compression results in a compressed signal that can return the same original actual signal [2]. Recent compression methods can be classified into several categories, such as (a) Wavelet, (b) DCT, (c) Vector quantization, (d) Genetic algorithm and (e) Fractal methods.

One benefit of fractal compression over the conventional transforms coding approaches is that it occupies merits of local scale invariance". Many natural image features, such the contrast regions and straight edges, are not changed by re-scaling. This kind of redundancy is difficult to exploit, and this is the major impulse of fractal image coding [3].

### B. Fractal Image Compression

A new image compression technique, called fractal compression, has become rapidly popular. Nowadays it is being utilized in many of implementations, including image compression, feature extraction, image signatures, texture segmentation, and image watermarking [4]. Fractal image compression is an encouraging technique for compact storing of images. It depends on fractals and utilizes self-symmetry which can be observed in all images. It is a lossy technique that poses no troubles for applications, such as using the internet to submit and move the images [5]. The first researcher who used the concept of fractals in image compression was Michael Barnsley. He proposed that by utilizing refined repetitions of the affine transformation of the plane, one could result from a fractal like an image via saving the image as a combination of transformations instead of a combination of pixels [6]. The merging of many affine mappings on the whole image composing these transforms that are recognized as Iterated Function Systems (IFS). The fractal image compression that depends on IFS was difficult and yet not workable. Jacquin's introduction of the partitioned IFS (PIFS) makes the fractal image compression a workable fact. The difference between PIFS and IFS is that Jacquin divided the image into groups named as the range blocks, while the temporary images utilized to make the range blocks are named as the domain blocks. The PIFS were performed on the groups instead of being performed on all images [6, 7, 8]. The method of fractal image encoding included dividing the image to create the range blocks and domain blocks, identification of domain blocks, and appropriate implementation of affine transforms. On every range block, the set of transformations taken is performed to guide the domain blocks, as the elapsed time (encoding) is longer than the decoding time, which makes the fractal coding asymmetric. The decoding operation affects the specific IFS transformations on any preliminary image, and the operation is reduplicated numerous times until arriving at the attractor [9].

Fractal compression, compared to other image compression techniques, makes an acceptable balance between the compression ratio (CR) and the quality of image. The advantage of fractal coding is the ability to rebuild the compressed image to any size without the loss of details [10]. Fractal compression is one of the efficient codec methods. Many researchers paid attention to the process of further planning of the models that could reach properties such as more efficient compression ratio (CR), independent resolutions, and high grade of decoding speed, etc. [11, 12, 13]. The technique of fractal image compression is proper in the operation of image signature solutions [14, 15], texture segmentation process [16, 17], image retrieval [18, 19] and the distinct methods like MRI- and ECG-based image processing [20, 21]. In addition, it focuses on other types of image processing applications, image denoising [22, 23], and digital watermarking solutions [24, 25]. A major setback in the model is its complexity, long encoding time, and time taken to find the best block matching in a large pool of domain blocks [26]. Thus, speeding up the encoding is necessary and many attempts were conducted to speed FIC using different methods. Numerous techniques targeted the aspect of addressing the conditions of encoding time [10, 27, 28, 29]. Decreasing the size of a domain pool could be another techniques adapted in many ways [30, 31, 32].

### **Related Work**

The fractal image compression (FIC) has been the centric point in numerous research studies. Various techniques for generating fractals have been developed and used to produce images, such as fractal with feature extraction, DCT, wavelet, Quadtree, genetic algorithms, and neural networks, etc. Table-1 presents a comparison between these works in terms of the technique used and results.

#### **1. Fractal image with features extraction**

Saad [33] described the range of images by extracting a few number of features to improve the basic fractal image compression. The mean and variance features were extracted of all the range blocks. The mean (Mr) value of a range offers the amount of the average gray level of that range, whereas the variance (Vr) value of a range explains the dispersion of its gray level from the mean; it is utilized to determine if the range area is a homogenous region or it has details. Several values of the variance are used to control the areas of homogeneity; these values are called Homogenous Permittivity (HP), which demonstrate the level of homogeneity accepted. This can (i) decrease the number of complex calculations needed by the FIC compression method and (ii) lead to reducing the encoding time. Baviskar and Pawale [34] proposed an effective domain search method by feature extraction for fractal image compression that reduces coding time and enhances the form of the image that is compressed. For every image block, a small number of features are computed; these features are: (i) mean, (ii) standard deviation, (iii) skewness, and (iv) kurtosis. These features are used to distinguish the range and domain images and to compare the domain and range blocks rather than using separate pixels to distinguish and compare.

#### **2. Fractal image with DCT**

Rawat and Meher [35] combined the DCT with the fractal image compression. The color image is compressed using the DCT and, to avoid the repetitive coding of similar blocks, the fractal image compression is used. Equivalent blocks are discovered by using the Euclidean distance measure. The image is partitioned into  $8 \times 8$  non-overlapping blocks. For all blocks of the image, the DCT is used and then the DCT coefficients are quantized. The zero values are removed by looking into the block values in a zig-zag technique; this technique enhances the compression quality and extracts the nonzero values. In this stage, the process of fractal image compression is applied. Lastly, the image is compressed by Huffman encoding. Gupta et al. [36] exercised a DCT for the optimization of image encoding. Two simple adaptive techniques, namely the variance-based selection of image blocks and the adaptive threshold, are applied for gaining the goal of image compression. The first technique, the range block exclusion adaptive technique eliminates similar range blocks before FIC compression. This exclusion decreases the number of domains and range blocks of matching operations. The hybrid compression algorithm localizes the position of the domain block around the range block without checking the whole domain pool. In the second technique, a decrease in the domain pool size is performed. Differences between domain and range blocks are matched. Each domain block that is within the threshold value is preserved in the pool. Lastly, quadtree partitioning is applied to optimize the achievement of encoding adaptive threshold with variance domain selection process.

### 3. Fractal image with wavelet

Feng et al. [37] proposed a fractal image compression technique which depends on multi-wavelet transform. This technique is performed by decomposing the original image into sub-images at different spatial frequencies through multi-wavelet transformation. Then, on the high-level wavelet coefficients, fractal coding is performed only by using the correlation between the wavelet coefficients at various scales and estimating the fractal code of the low-level wavelet coefficients from that of the upper-tiered wavelet coefficients. Rather than just building on the foundation of the neighborhood gray-scale correlation, it transforms the image model into a collection of relevant elements in the frequency domain. So that a larger image block can be coded, the possibility of global correlation in the image can be removed, and the coding speed of the existing fractal image compression algorithm can be enhanced. Hien [38] proposed a four-fork tree with a novel wavelet and fractal image coding technique, by examining the restrictions of the present wavelet-based image compression processes. Initial errors happen at various levels of significance, depending on the frequencies of sublevel-band wavelet coefficients. Higher frequency sublevel bands may end up with larger initial errors. Consequently, the initial errors of the sizes of sublevel blocks and superblocks vary. Also, because of the permitted errors and compression rates, the matching sizes between sublevel blocks and superblocks vary. Finally, the Haar wavelet was analyzed for 5-level decomposition for image compression.

### 4. Fractal image with QuadTree and Huffman coding

Veenadevi and Ananth [39] divided the original grey image into non-overlapped blocks, according to a threshold value and Quadtree decomposition, to achieve fractal image compression. These methods were applied for the compression of satellite imageries by using a threshold value and Huffman coding for encoding and decoding of the image. Pandey and Seth [40] performed a method of fractal image compression using quadtree decomposition. The block size in the Quadtree method varies according to the features of the image and the partition into different blocks. Later, the Huffman coding was applied in the compressed fractal image leading to the improvement its performance. Various quantitative computing methods are established by passing images of various formats and dimensions and, based on those quantitative measures, a decision is made as to which image format is better for compression.

### 5. Fractal image with genetic algorithm

Pandey and Singh [41] suggested a fractal image compression technique with the investigation of a novel image compression scheme that depends on genetic algorithm. In the tested image compression, a global technique was applied that can integrate genetic algorithm and other image compression schemes to achieve parameter alleviation and to perform application requirements. This is performed by applying the genetic algorithm process for finding best possible optimization through the compression process utilizing the different crossover methods. The goal is to quantitatively analyze the existing and suggested image compression schemes on the basis of few parameters, including various crossovers. The mechanism is performed and simulated using (i) Fractal image compression with different images and (ii) Same fractal compression technique with suggested adaptive genetic algorithm method utilizing variants of Crossover. After that, the effectiveness of these two methods was compared on few famous images in different scenarios and parameters. The outcomes showed that the suggested methods have better performance. Saxena and Singh [42] improved the genetic algorithm to raise the CR and quality of the digital image. This algorithm process involves the parallel selection of transform blocks in multiple sequences. The fractal transforms function is used for the transformation of image data. The fractal transforms provide the non-overlapping blocks. The second dimension is used to map the dimension and transform the image. The improved genetic algorithm process of data relies on the reference direction of the image. The direction of the image was divided into three sections: horizontal, vertical and diagonal. The modified algorithm is a combination of FIC and genetic algorithms. The fractal transforms function is used for the decomposition of data in different layers and the different layers precede the data for the input of parallel genetic algorithm. The parallel genetic algorithm precedes the data for searching for the best set of pixels or transforms value for the aims of input of the head code matrix (HCC matrix).

## 6. Fractal image with neural network

Zhou et al. [43] utilized a neural network to alter the variance depending on a fractal image compression algorithm. The original image has different subblocks and, depending on these blocks, some mapping structures are selected automatically by neural network. If the pixels of subblocks have few differences, the affine mapping had a perfect outcome between the matched subblocks. If the pixel values of subblocks differ highly, the affine mapping turns into inefficient for the mapping and the outcomes are in a large matching error. This problem is solved by using a neural network to structure the mapping for the subblocks, when the pixels differ highly. Each pixel in the subblocks is considered as a neuron in the network; the pixel in its matched domain blocks as input neurons and the pixel in range blocks as output neurons. Chakrapani and Rajan [44] classified the domain pool objects of a gray image using the developed neural network, so that the encoding time and quality of the image were improved. The feature extraction reduced the volume of the issue and also allowed the neural network to be trained on an image detached from the test image and, therefore, reduced the computations required during the search. The prime benefit of the neural network is adapting itself from the training data depending on the distribution of feature space noticed in the training stage.

**Table 1-** Summarized previous research work.

References	Technique Used	Results
[33]	Feature Extraction mean and variance features	<ol style="list-style-type: none"> <li>1. Reduced the time spent in the coding process by almost 70%, access to a higher CR, and good image quality when using a homogeneous permittivity HP.</li> <li>2. The test on Lena image (256x256) with block size= 4x4, CR=5.81, PSNR=28.69, and time=9 sec.</li> </ol>
[34]	Feature Extraction Mean(M), standard deviation(sd), skewness(S), and kurtosis(k).	<ol style="list-style-type: none"> <li>1. Good PSNR at the cost of less CR provided when least sized range blocks were used.</li> <li>2. Enhanced compressed image quality and reduced encoding decoding time.</li> <li>3. The results for monkey.jpg (8-bit/pixel) image with block size = 4x4 are CR=6.8705215, PSNR=24.890816, and encoding time=66.5 sec.</li> </ol>
[35]	A Hybrid mechanism using DCT, Fractal Image and Huffman encoding.	<ol style="list-style-type: none"> <li>1. The suggested method gave better visuality than JPEG.</li> <li>2. The method showed success in coding the color image.</li> <li>3. Higher PSNR, Universal Image Quality Index (UIQI) and Structural Similarity (SSIM) when matched to JPEG with certain image quality elements.</li> <li>4. Good PSNR and image quality.</li> <li>5. The results on color Lena image with block size= 8x8 are CR=11.1544, and PSNR= 31.5739.</li> </ol>
[36]	Discrete Cosine Transform (DCT) and quadtree partitioning	<ol style="list-style-type: none"> <li>1. The test images used are 8 bit Lena, Pepper and Baboon of size 512 x 512.</li> <li>2. CR= 25.54, ET=12.95 sec, PSNR=29.13 on Lena image.</li> <li>3. CR= 23.62, ET=15.23 sec, PSNR=28.16 on Pepper image.</li> </ol> <p>CR=08.64, ET=35.88 sec, PSNR= 21.07 on Baboon image.</p>
[37]	Multi-wavelet transform	<ol style="list-style-type: none"> <li>1. Multiwavelet-FIC improved the compression time by reducing it by 22.23% on average. The CR was reduced to 1.347 and the PSNR was decreased slightly.</li> <li>2. Multiwavelet-FIC reduced the coding complication in comparison with FIC, it shortened the matching search time, and accelerated the coding speed.</li> <li>3. The results on gray peppers image with block size=4x4 are CR=10.241, PSNR=30.92, and time=33.668 sec. The results on pears image with block size=4x4 are CR=12.209,</li> </ol>

		PSNR=28.87, and time=19.861 sec.
[38]	Wavelet transform with the four-fork tree	<ol style="list-style-type: none"> <li>1. This algorithm increased the CR, while the speed of encoding time and the PSNR were decreased; the distortion evaluation was not high.</li> <li>2. The results on gray Lena image are CR=25.6, PSNR=29.3, and time=36sec.</li> </ol>
[39]	Quadtree decomposition and Huffman coding	<ol style="list-style-type: none"> <li>1. Very significant advancement in the CR and PSNR values derived.</li> <li>2. The results on gray Lena image (256 X 256) are CR=2.0, PSNR=29.92, and Time=20.20 sec.</li> </ol>
[40]	Quadtree decomposition and Huffman coding	<ol style="list-style-type: none"> <li>1. The png and jpeg image works well with this suggested method but this method is not appropriate for bitmap image format.</li> <li>2. This method is performed for color images that take more encoding time but achieve high CR and better PSNR values.</li> <li>3. The results on color Lena.jpg are CR=27.693, PSNR=29.912, and time=86.612 sec.</li> </ol>
[41]	Genetic algorithm	<ol style="list-style-type: none"> <li>1. Compression ratio (CR) in lena.jpg is 63.35% in one-point crossover and 77.6% in two-point crossover, while in the image of Cameraman.jpg the values were 1.4 times and 1.5 times, respectively.</li> <li>2. Encoding and decoding times were reduced in both crossovers and test images.</li> <li>3. PSNR needs to be optimized with the increased compression ratio.</li> <li>4. In all the test images and in both types of crossovers, the decrease in MSE was observed to a perceptible level.</li> <li>5. In both the crossovers and test images, an increase trend in PSNR was noticed.</li> </ol>
[42]	Fractal transform function and parallel genetic algorithm	<ol style="list-style-type: none"> <li>1. Good image quality and PSNR value.</li> <li>2. The results on gray Barbara.png image (512× 512) are CR=8.4262, PSNR=19.1998.</li> </ol>
[43]	Neural network	<ol style="list-style-type: none"> <li>1. The PSNR is greater than that in the traditional variance-based method. The coding time and the CR are nearly similar to those in the traditional variance-based method.</li> <li>2. The variance-based method increased the coding time by 22 times, but the rebuild image quality is decreased clearly.</li> <li>3. The rebuild image quality is similar or greatly better than that in the baseline encoding algorithm.</li> <li>4. The results on gray peppers image (256×256) are PSNR=33.41, time=0.64sec.</li> </ol>
[44]	Neural network (Back-propagation algorithm)	<ol style="list-style-type: none"> <li>1. The network is properly trained and the domains are classified correctly with a minimum deviation which helps in encoding the image using FIC.</li> <li>2. The PSNR is less for the grayscale image by neural network method depends on FIC than that of FIC utilizing full search method.</li> <li>3. The encoding time is good.</li> <li>4. The results on gray Lena image (128 × 128) are block size=4 × 4, CR=5, PSNR=34.434, and time=2400.</li> </ol>

## 7. Fractal image compression using block indexing technique

A novel fractal compression technique is presented; this technique depends on indicating the zero-mean range blocks by implementing moment descriptors. The principal concerns for the moment descriptor search scheme are:

1. When the blocks are identical, they must have roughly a similar descriptor amount, but they may not be similar to the same descriptor amounts.
2. Candidate blocks are specified using the descriptor search scheme. These blocks must be identical in their descriptor values utilizing single identical for matching. These values must be tested utilizing block based matching process.
3. It is applied to specify the image blocks in the range and domain pool.
4. It performs the IFS-mapping stage as a selective process, rather than that each domain block is tested to detect the best approximate for all range block.
5. It is used to speed up the range-domain seeking task, which consequently speeds up the matching stage.
6. The total of the matching blocks in the search window is reduced by exchanging it with identical block descriptors matching.
7. Adopting the asymmetry assessment method to achieve extra time reduction and to estimate the near-optimal isometric transform is needed to obtain the perfect matching with any two blocks. This predictor is employed and thus will reduce the eight isometric transformation trials for the domain block to one.
8. The range pool partition is achieved for any chosen block size.

The most complex task in the layout of the descriptor search is how to determine the perfect special descriptor for all blocks. Several types of research were published within block indexing mechanisms for fractal image compression, which increased the encoding time with a high quality of the reconstructed image. The published works are listed below [45-53]:

- A novel fractal coding planner which depends on IFS-transform for zero-mean to domain and range blocks was implemented by George [45]. Several enhancements were accomplished on the matching process, which utilize the moment indicating as a standard to classify the blocks of the domain in an appropriate manner that they look like the blocks of the range. Also, the use of finish search condition depends on controlling the minimum error for matching. The fractal coding long time will be minimized by these extra coding proceedings. The encoder includes a group of transforms on parts of the image, which are the domain or range blocks. The domain blocks are overlapped, whereas the range blocks are non-overlapped. The group of transformations is composed of a spatial contraction (such as the mean of every four adjacent pixels) to build  $k \times k$  blocks from blocks of  $2k \times 2$  size. Then one of the eight square symmetry process (four rotations and four reflections) is performed, followed by a contractive affine transformation on the grey pixel values.
- A new fractal compression technique to index the zero-mean range on domain blocks by moment criteria was proposed by Al-Hilo and George [46]. The block descriptor "moment's ratio index" was determined using moment criteria; this descriptor was performed to determine the blocks in domain and range pools. In the encoding stage, the block moment ratio descriptor of each range block is used to filter the domain blocks and take only those blocks whose moment descriptor is appropriate to be IFS-matched with the tested range block. Initially, the loaded RGB is converted to YUV and the (U, V) component is down-sampled to obtain a reliable compression. FIC method is used to code individually each component of YUV and the image is divided to obtain non-overlapping range blocks. An identical but larger domain block is presented to each range block. The method used in this job is a fixed size dividing method to divide images, because it needs little calculation times compared with the other methods used. The domain blocks in which moment ratio factors are comparable to that of the tested range block are achieved using the IFS method instead of the full search method.
- George and Al-Hilo [47] depended on the zero-mean process to suggest a mechanism for speeding-up the fractal compression for color images utilizing a symmetry predictor. This predictor was used to decrease the total of isometric trails from eight to only one. IFS compression using the zero-mean process implies that the offset values of the block are changing with the mean values of brightness. Depending on this change, the equation of IFS mapping was implemented. For all domain and range blocks, the first order centralized moments are implemented to index onto 1 of 8 potential isometric

states. These blocks indexes are passed via the predictor that produces the index of the desired isometric transform to obtain a good matching of range and domain. Also, the indexes of range average and scale are coded by implementing Differential Pluses Coding Modules (DPCM) followed by shift coding.

- George and Minas [48] suggested an improved PIFS scheme; the image in the PIFS scheme is transformed to YUV, then the U, V components are down-sampled. A block descriptor is implemented to arrange the zero-mean of domain and range blocks. This descriptor is computed by a pair of low-frequency DCT coefficients because of the energy packing features of DCT. Two kinds of R-descriptors have been tested (i.e., the ratio descriptor, and the relative difference). For each kind, the two sets of DCT coefficients were taken into consideration. Hence, the isometric state of blocks of both range and domain are addressed using these descriptors. The quantization is performed on optimal affine approximation coefficients; they are coded by implemented DPCM, and finally the shift coding is performed.
- George and Kadim [49] depended on block indexing and proposed a vector quantization compression mechanism to compress color images. An affine transform is used to constitute the codebook using the blocks of the color image. To nearly clarify the blocks of the compressed color image by using a template image (domain pool), it must have more contrast areas like a codebook. The image blocks are indexed using block descriptors, which are obtained from the DCT coefficients. The DCT-based descriptor is an affine transform invariant. The DCT descriptor is utilized to classify the blocks of the domain and matching is performed only with comparable indexed blocks. DCT shows the image by its elementary frequency ingredients; by which the more related information of the image is saved or compressed in the top left corner of the color image; this characteristic of DCT is named energy compaction, where it is logical to easily drop out the high frequency coefficients, like the approximate mean, to classify the block characteristics.
- Mahmoud [50] applied the IFS using double descriptors and symmetry prediction. These descriptors are dependent on the quantity of the central moments of order-1 and order-3. They are incorporated to obtain block descriptors, classified like the affine transform invariant descriptors, to mark the domain and range blocks and increase the IFS matching. The isometric predictor used the order-1 moment status of the block matching to decrease the total time of matching process, because, rather than 8 matches, only one match will be performed to determine the comparability for both range and domain blocks. The coefficients of the best affine approximation are quantized to implement high CR and saved as a compressed file. The moment features are utilized for block indexing to obtain double block descriptors such that the affine invariant is accomplished for each one.
- Al Hilo and Al-khafaji [51] introduced the impact of quantization on the rebuilt image in the FIC technique. This method depends on the zero-mean block for the block matching process; an average of the range blocks is used as a substitute for the traditional offset parameter. The uniform quantization was implemented because it can work faster than the other methods. To increase the compression, IFS coefficients should be quantized before prior saving because the IFS coefficients are real-valued. The quantization process affects the quality of the rebuilt image. The conversion of the real numbers to integers causes the forgetting of additional information. When the information missing is large, the quality of the rebuilt image will be poor. If the missing information is little, better quality and little compression ratio of the rebuilt image will be achieved.
- Al Hilo and Al-khafaji [52] investigated the impact of image size on the compression coefficients of the (FIC) method. In this work, the same encoding technique used in an earlier work [46] and three variant sizes of the gray image (100 x 100, 150 x 150, 256 x 256) were utilized.
- Sultan et al. [53] proposed an enhanced fractal image compression system (FIC); it speeds up the blocks matching process by utilizing the symmetry prediction and indexing of blocks. The proposed system used a quadtree as a variable range block dividing method. The partitioning decision has two criteria: the Sobel-based edge magnitude and the contrast of block. The proposed system introduced a new set of moment descriptors which can emphasize the weights of different parts of each block. The performance of all possible combinations between double moment descriptors was investigated and a fast computation mechanism was introduced to calculate the moments attended to enhance the overall computation cost. The enhanced algorithm of FIC based on the partition IFS method was represented in: (i) The scheme of range pool partitioning, (ii) IFS matching with low computation redundancy, and

(iii) using a new set of centralized moments which are complementary and more informative. Table-2 presents a comparison of the results of previous works on the Lena image in terms of compression ratio (CR), Peak Signal to Noise Ratio (PSNR), and coding time.

**Table 2-** Comparison of Compression ratio (CR), PSNR, and encoding time on Lena image in block indexing techniques.

Reference	Image size	Color depth	Block size	CR	PSNR	Time (sec)	Results
[45]	256x256	8	4x4	5.09	32.6	236.3	<ol style="list-style-type: none"> <li>The IFS-method provides better compression results than the conventional fractal coding technique, and the offset coefficient value is best encoded by utilizing DPCM.</li> <li>Encoding time decreased (more than 3.5 times), with no high distortion in image quality.</li> </ol>
[46]	256x256 color image	24	4x4	10.97	30	4.82	<ol style="list-style-type: none"> <li>Reduction in time and no notable distortion in the quality of the image.</li> <li>There is a 96% reduction (or 25 times) in the time when comparing the results with those of the conventional techniques.</li> </ol>
[47]	256x256 color image	24	4x4	10.34	33.28	4.81	<ol style="list-style-type: none"> <li>FCI compression method is more efficient than the conventional technique (Zero-Mean) in the coding color image because they make proportional reduction in time and no notable distortion in quality.</li> <li>Reduction in time (around 84%) PSNR (around 1.7%), and CR (around 0.1%).</li> </ol>
[48]	256x256 color image	24	5x5	13.533	31.66	1.02	<ol style="list-style-type: none"> <li>The encoding time is decreased to 1 sec, image quality is preserved more than the passable level, and no considerable lowering in the CR.</li> <li>Using the color model YUV leads to good CR when it is compared with the case of using RGB.</li> </ol>
[49]	256x256 gray image	24	4x4	7.31	30.09	1.13	<ol style="list-style-type: none"> <li>Decreased coding time to about 17.2 % in comparison to the conventional scheme.</li> <li>The matching process becomes faster when utilizing DCT coefficients to filter the image blocks.</li> </ol>

[50]	256x256 gray image	24	4x4	7.99	30.56	0.31	The double moment descriptors used with symmetry predictors reducing the coding time to about 50% of the coding time using a single moment descriptor, with no considerable distortion in the quality of the image.
[51]	256x256 color image	24	4x4	10.3 3	33.85	34.64	<ol style="list-style-type: none"> <li>1. The coding time decreased to 54.51%, and an increment in the CR of 5.24%.</li> <li>2. Quality was almost conserved in comparison with the traditional methods.</li> </ol>
[52]	100x100 gray image	8	4x4	5.27	29.77	1.42	<ol style="list-style-type: none"> <li>1. The PSNR and encoding time are higher but the CR is reduced with the increment of image size.</li> <li>2. The quality of the reconstructed image is acceptable when using any size of an image.</li> </ol>
	150x150 gray image	8	4x4	5.17	21.07	6.81	
	256x256 gray image	8	4x4	4.74	34.75	59.86	
[53]	256x256 color image	24	4x4	7.97 7	27.81 3	0.209	<ol style="list-style-type: none"> <li>1. The variable partitioning scheme can reach better results than the fixed partitioning one (4 x 4 blocks), the CR is faster and PSNR is not significantly reduced.</li> <li>2. The quadtree partitioning is considered as a useful partitioning mechanism which is guided by Sobel or variance.</li> <li>3. The proposed moments are appropriate to perform double block descriptions that can increase the IFS encoding method.</li> </ol>
		24	8x8	32.1 15	24.51 5	0.064	

When comparing the fractal using block indexing technique with other techniques, using color and gray Lena bitmap image (i.e. articles [33,35, 38, 39,44]) on block size of [4x4] and [8x8], the proposed block indexing method is better in CR and rebuilt image quality, while the coding time of this method is about 45 times faster.

### Conclusions

There were many studies of different fractal image compression techniques; these techniques have many merits and demerits. The block indexing process can provide more optimal results than other

techniques. The main concept of the block indexing process is that each block is classified by a descriptor value to the candidate domain blocks and to reduce the encoding time.

## References

1. Sayood, K. **2017**. *Introduction to Data Compression*. Fifth edition. Elsevier.
2. Havaladar, P. and Medioni G. **2010**. *Multimedia Systems: Algorithms, Standards, and Industry Practices, Course Technology*. Cengage Learning. Canada.
3. AL-Ali, H. B. T. **2013**. Speeding-up Fractal Audio Compression Using Double Moment Descriptor. Ph.D. dissertation, College of Science, University of Baghdad, Baghdad, Iraq.
4. Ozturk, A. and Gungor, C. **2011**. A Fast Fractal Image Compression Algorithm Based on a Simple Similarity Measure, *Journal of Combinatorics, Information and System Sciences*, **36**(1): 159-178.
5. Barnsley, M. **1988**. *Fractals Everywhere*. Academic Press. New York.
6. Jacquin, A.E. **1992**. Image coding based on a fractal theory of iterated contractive image transformation, *IEEE Transactions on Image Processing*, **1**(1): 18-30.
7. Jacquin, A. E. **1993**. Fractal image coding, *Proceedings of IEEE*, **81**(10): 1451-1465.
8. Fisher, Y. **1994**. *Fractal Image Compression: Theory and Application*. Springer-Verlag. New York.
9. Padmashree, S. and Nagapadma, R. **2012**. Comparative Analysis of Quad-tree Partitioned Iterated Function Systems for Medical Images, *International Journal of Computer Application*, **59**(14): 25-27.
10. Revathy, K. and Jayamohan, M. **2012**. Dynamic Domain Classification for Fractal Image Compression, *International Journal of Computer Science & Information Technology (IJCSIT)*, **4**(2): 95-102.
11. Kamal, N. B. **2015**. Iteration Free Fractal Image Compression for Color Images Using Vector Quantization, Genetic Algorithm and Simulated Annealing, *The Online Journal of Science and Technology*, **5**(1): 39-48.
12. Xi, L. and Zhang, L. **2007**. A Study of Fractal Image Compression Based on an Improved Genetic Algorithm, *International Journal of Nonlinear Science*, **3**(2): 116-124.
13. Sheeba, K. and Abdul Rahiman, M. **2016**. Iteration less Wavelet-Fractal Image Compression Applicable in Cellular Mobile Communication System, *International Journal of Computer Applications*, **136**(8): 40-44.
14. Al-Saidi, N. M. G. **2014**. An efficient signcryption method using fractal image coding scheme, *International Journal of Circuits, Systems and Signal Processing*, **8**: 68-76.
15. Nikolaou, N. and Papamarkos, N. **2002**. Color image retrieval using a fractal signature extraction technique, *Engineering Applications of Artificial Intelligence*, **15**(1): 81-96.
16. Sarafrazi, K., Yazdi, M., and Abedini, M. J. **2013**. A New Image Texture Segmentation Based on Contourlet Fractal Features, *Arabian Journal for Science and Engineering, Springer*, **38**(12): 3437–3449.
17. Marrón, B. **2018**. Texture Filters and Fractal Dimension on Image Segmentation, *Journal of Signal and Information Processing*, **9**(3): 229-238.
18. Qin, Z., Qinghua, L. and Xin, K. **2017**. An Effective Image Retrieval Method Based on Fractal Dimension Using Kernel Density Estimation, 4th International Conference on Information Science and Control Engineering (ICISCE), IEEE, pp: 159-163, Changsha, China.
19. Zhang, L., Wang, Y., Xi, L., Gao, K. and Hu, T. **2008**. New Method of Image Retrieval Using Fractal Code on the Compression Domain, *Wseas Transactions on Systems*, **12**(7): 1484-1493.
20. Oien, G.E., Krakeland, H. and Narstad, G. **1996**. Fast fractal-based ECG Signal Compression, Proceedings of 18th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, IEEE, pp: 1378- 1379, Amsterdam, Netherlands.
21. Marusina, M.Y., Mochalina, A.P., Frolova, E.P., Satikov, V.I., Barchuk, A.A., Kuznetsov, V.I., Gaidukov, V.S. and Tarakanov, S.A. **2017**. MRI Image Processing Based on Fractal Analysis, *Asian Pacific Journal of Cancer Prevention*, **18**(1):51-55.
22. Ghazel, M., Freeman, G.H. and Vrscey, E. R. **2003**. Fractal Image Denoising, *IEEE Transactions on Image Processing*, **12**(12): 1560-1578.

23. Barthel, K.U., Cycon, H.L. and Marpe, D. **2003**. Image Denoising Using Fractal and Wavelet-Based Methods, *Proc. SPIE*, **5266**: 10–18.
24. Shrivatava, A. and Sharma, S. **2017**. A Modified Approach in Watermarked Image using Fractal Technology, *International Journal of Computer Trends and Technology (IJCTT)*, **47**(3): 156-160.
25. Channapragada, R. S.R. and Prasad, M. V. N. K. **2015**. Digital Watermarking Using Fractal Coding, *Advances in Signal Processing and Intelligent Recognition Systems, Springer*, pp:109-118.
26. Belloulata, K., Zhu, S. and Wang, Z. **2011**. A Fast Fractal Video Coding Algorithm Using Cross-Hexagon Search for Block Motion Estimation, *International Scholarly Research Network ISRN Signal Processing*, 2011, pp: 1-10.
27. Salih, I. and Smith, S. **1999**. Encoding Time Reduction in Fractal Image Compression, Proceedings DCC'99 Data Compression Conference, IEEE, Snowbird, UT, USA.
28. Al-A'mri, J. H., Al-Amawee, H. A. and Al-Jumaily, K. J. **2004**. Adaptive Algorithm to Enhance the Fast Method of Fractal Image Compression, *Iraqi Journal of Science*, **45**(1): 221-229.
29. RaoD, S. and Chandra N, S. **2016**. Fast Search Approaches for Fractal Image Coding: Review of Contemporary Literature, *Journal of Computer Science and Technology*, **16**(3): 36-46.
30. Roy, S.K., Bandyopadhyay, S. K., Mahato, A. and Kim, T.H. **2015**. A Speeding Up Fractal Image Compression Using Fixed Size Partition and Hierarchical Classification of Sub-images, *Advances in Mathematics and Statistical Sciences, Proceedings of the 3rd International Conference on Mathematical, Computational and Statistical Sciences (MCSS '15)*, pp: 326-332, Dubai, United Arab Emirates.
31. Hassaballah, M., Makky, M.M. and Mahdy, Y. B. **2005**. A Fast Fractal Image Compression Method Based Entropy, *Electronic Letters on Computer Vision and Image Analysis*, **5**(1): 30-40.
32. Harbi, J. **2013**. Speeding-Up Fractal Image Compression by Using Classification Range Blocks, *Engineering and Technology Journal*, **31**(6): 770-780.
33. Saad, E. A. **2011**. Fractal Image Coding Based on the Features Extraction, *Electronic Computer Center, Al-Mustansiriyah University*, **22**(3): 85-92.
34. Baviskar, A. G. and Pawale, S.S. **2012**. Efficient Domain Search for Fractal Image Compression Using Feature Extraction Technique, *Advances in Computer Science, Engineering and Applications*, Springer, **166**: 353–365, Springer-Verlag Berlin Heidelberg.
35. Rawat, Ch. S. and Meher, S. **2013**. A Hybrid Image Compression Scheme using DCT and Fractal Image Compression, *The International Arab Journal of Information Technology*, **10**(6): 1-10.
36. Gupta, R., Mehrotra, D. and Tyagi, R. **2016**. Adaptive Search Less Fractal Image Compression in DCT Domain, *The Imaging Science Journal*, **64**(7): 374-380.
37. Feng, Y., Lu, H. and Zeng, X. **2015**. A Fractal Image Compression Method Based on Multi-Wavelet, *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, **13**(3): 996-1005.
38. Hien, V. T. **2009**. A New Wavelet–Fractal Image Compression Method, 13th International Conference, KES 2009, Santiago, Chile, Part I, LNAI 5711, pp: 161–168, Springer-Verlag Berlin Heidelberg.
39. Veenadevi, S.V. and Ananth, A.G. **2012**. Fractal Image Compression Using Quadtree Decomposition and Huffman Coding, *Signal & Image Processing: An International Journal (SIPIJ)*, **3**(2): 207-212.
40. Pandey, Sh. and Seth, M. **2014**. Hybrid Fractal Image Compression Using Quadtree Decomposition with Huffman Coding, *International Journal of Science and Research (IJSR)*, **3**(6): 943 -948.
41. Pandey, A. and Singh, A. **2015**. Fractal Image Compression using Genetic Algorithm with Variants of Crossover, *International Journal of Electrical, Electronics and Computer Engineering*, **4**(1): 73-81.
42. Saxena, Sh. and Singh, R. P. **2016**. Digital Image Compression Using Improved Genetic Algorithm, *IOSR Journal of Computer Engineering (IOSR-JCE)*, **18**(6): 77-83.
43. Zhou, Y., Zhang, C. and Zhang, Z. **2006**. Improved Variance-Based Fractal Image Compression Using Neural Networks, *Lecture Notes in Computer Science, Springer*, pp: 575 – 580.
44. Chakrapani, Y. and Rajan, K. S. **2008**. Implementation of Fractal Image Compression Employing Artificial Neural Networks, *World Journal of Modelling and Simulation*, **4**(4): 287-295.

45. George, L. E. **2006**. IFS Coding for Zero-Mean Image Blocks, *Iraqi Journal of Science*, **47**(1): 190-194.
46. Al-Hilo, E. and George, L.E. **2008**. Speeding-up Fractal Colored Image Compression Using Moments Features. In: *Digital Image Computing: Techniques and Applications (DICTA)*, IEEE. Pp: 486–490, Canberra, ACT, Australia.
47. George, L.E. and Al-Hilo, E.A. **2009**. Speeding-Up Color FIC Using Isometric Process Based on Moment Predictor, *International Conference on Future Computer and Communication*, IEEE Computer Society, pp: 607-611, Kuala Lumpur, Malaysia.
48. George, L. E. and Minas N. A. **2011**. Speeding Up Fractal Image Compression Using DCT Descriptors, *Journal of Information and Computing Science*, **6**(4): 287- 204.
49. George, L. E. and Kadim, A. M. **2011**. Color Image Compression Using Fast VQ with DCT Based Block Indexing Method, *Springer*, 6754, pp: 253–263, 2011.
50. Mahmoud, S. L. **2012**. The Use of Double Moment Based Descriptors to Speed up FIC, *Journal of Al-Nahrain University, Science*, **15**(4): 200-202.
51. Al-Hilo, E.A. and Al-khafaji, K.H. **2013**. Effect Of Quantization Process On FIC For Gray Image by Using Zero-Mean Method, *Journal of Kufa – Physics*, **5**(2): 8-14.
52. Al-Hilo, E. A. and Al-khafaji, K. H. **2015**. Study the Performance of Fractal Gray Image Compression Using Different Images Size, *Journal of Kufa – Physics*, **7**(2): 1-7.
53. Sultan, B. A., George, L. E. and Hassan, N. F. **2018**. The Use of Quadtree Range Domain Partitioning with Fast Double Moment Descriptors to Enhance FIC of Colored Image, *ARO-The Scientific Journal of Koya University*, **VI**(1): 13-22, 2018.