

ADVANCED ALGORITHM FOR EXPEDITING IMAGE PROCESSING

Noor Adnan K. Al-Zeyadi

Department of Electrical and Electronics Engineering, University of Thi-Qar, Thi-Qar, Iraq

Mohammed A. Hussein

Electrical and Electronics Engineering Dept, University of Kerbala

Amean Sh. Al-Safi

Department of Electrical and Electronics Engineering, University of Thi-Qar, Thi-Qar, Iraq

Abstract:

Processing images more effectively is possible when a novel technique based on fractal image compression is applied. Range blocks and domain blocks are the overlapping and non-overlapping portions of an image that are separated to achieve a fast encoding time. Typically, domain blocks are bigger than range blocks. This study proposed a novel rapid full-search algorithm technique to drastically cut down on encoding time, one that expands the search until a suitable match is located or the search is finished. Using the nearest points in the range blocks as a starting point, this method searches the range block for the best matching domain. Despite being straightforward to use, the recommended rapid full-search strategy outperforms the traditional search strategy. The proposed method's search reduction, peak signal-to-noise ratio, compression ratio, and encoding time are all assessed. Test results show that the proposed method may reduce the overall search time by 89% and encode a 512x512 gray scale Lena image in 0.31 seconds.

Introduction

We can view items differently thanks to the geometrical ideas of fractals. Fractal geometry was developed as a result of Mandelbrot's research in 1982. Several iterations of a straightforward mathematical function must be performed before it is either found to be unbounded or to converge on a single result. Fractal geometry was subsequently applied to picture compression. Fractals are a useful tool for converting a 2-D image into a statistically uncorrelated dataset.[1-2]

The demand for digital images, computer animations, and multimedia has made data compression a significant issue. Transmission times remain a major requirement for data compression due to the high cost of storage and large channel bandwidth. Images can be compressed, which is a kind of data compression in which a limited amount of bits are used to encode the original data. In order to store or transmit images more effectively for storage or transmission, compression reduces picture duplication. The Fractal Image Compression (FIC) technique is easy to use and has some distinctive features like high compression rates and resolution independence. One of the best image compression techniques is this one. In addition to standard image compression, FIC has also been employed to compress MRIs [3–4] and hyperspectral images [5–6]. Image compression comes in two flavors: lossy and lossless. While the restored image with lossy compression shows a significant degradation, There is no numerical difference between the input image and the reconstructed image that went through lossless compression. This technique yields higher compression ratios than lossless compression. Fractal image compression is a type of lossy compression used in digital photography [7]. For high compression ratios and quick decompression, it makes use of the inherent affine redundancy seen in the majority of images. An image that has been modeled as a collection of fractals is encoded using the feature of self-similarity. Although fractal image compression requires a high encoding time, it has a number of benefits, including resolution independence, quick decoding, and a high compression ratio [8].

Compared to other techniques, fractal image compression (FIC) yields a greater compression ratio (CR), especially when applied to aerial or satellite photographs [9]. FIC has several uses in other areas of image processing, like character recognition, because of its growing use in digitalization [10] and watermarking [11]. FIC's excellent peak signal-to-noise ratio (PSNR) and simple decoding process are two more alluring features [12-13].

Fractal image reduction can be easily understood when compared to a copier machine, which can describe the fractal property[14]. The source image and the output image seem to be becoming closer, as seen in Figure (1). Every time a scale is used, the output image is detailed and created as copies of the original image. The image can be transformed eight various ways on a smaller copy. An input image can be transformed by skewing, stretching, rotating, scaling, and translating it. [15].

In mathematics, it can be written as:

$$W(x) = W \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix} \dots (1)$$

Where (x,y) is a coordinate point and (a,b,c,d,e, and f) are real numbers.

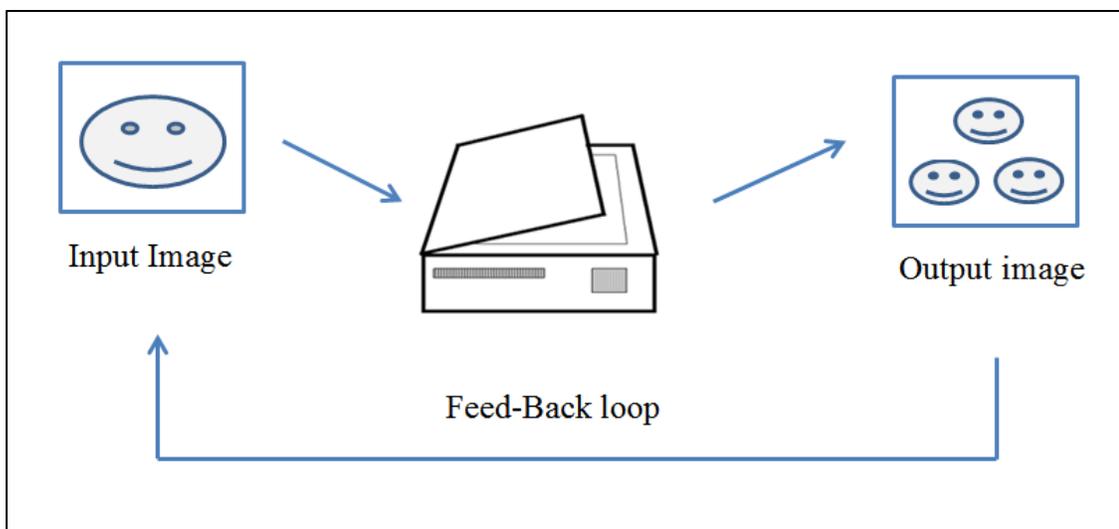


Figure (1): A photocopier creates three tiny copies of the original image.

To create a fractal encoded image, the image must be divided into smaller units. Ranges are the smaller pieces, and domains are the larger pieces. The size of the range block is usually half that of the domain. Based on the Partitioned Iterate Function System (PIFS), A. Jacquin developed an automatic picture encoding method [16]. After partitioning the image using the appropriate partitioning approach, he located the IFS for each of these blocks.

With the use of the contractive mapping theorem, IFS will finally choose a single fixed image following numerous repeats. The range must then be mapped across the domain to find a good matching domain. We effectively save the transformation or IFS code for each block. It would be necessary to conduct a thorough search for an effective fractal image encoding method.

Fractal Image Compression

Splitting an image that can be compressed into smaller images using blocks called range blocks and domain blocks is the first stage in the process of compressing fractal images.[17].

The block-based partitioning used in the fractal compression approach discovers and encodes similarities between different parts of a picture. Due to the low compression speed (long encoding time), which is enabled, the compression ratio is intriguing, but decompression can be quick. The longest part of the encoding process is the search for an efficient domain block for each range block.

The majority of the time required for fractal compression is spent on the matching of several blocks on an image. The Fisher classification method together with other artificial intelligence techniques (evolutionary algorithms, artificial neural networks, etc.) have been suggested to reduce encoding complexity and speed up fractal coding time[18]. Digital image data has been compressed using a variety of methods. These techniques can be classified into two groups: lossless and lossy. Each approach provides a unique set of coding options for the coding phase. Figure (2) illustrates the various types of picture compression algorithms.

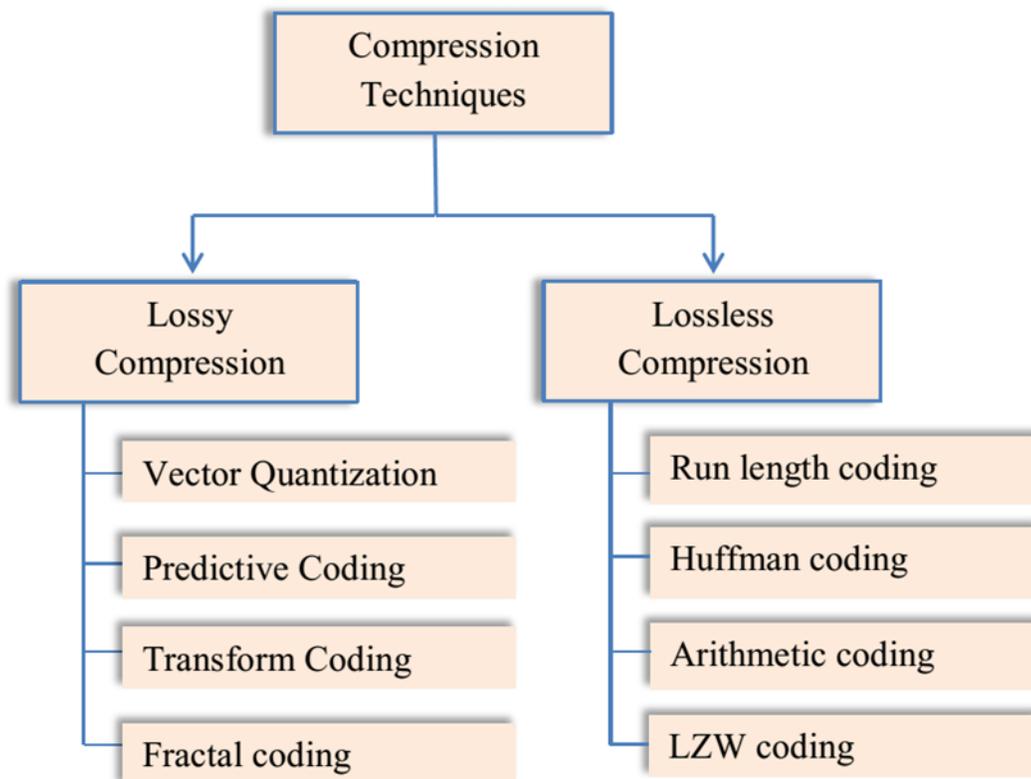


Figure (2): The grouping of methods of compression.

Fractal Image Encoding

Fractal image encoding is the procedure that begins with the image being divided into two pieces using PIFS and finishes with an encoded file attribute. The range of the image is the initial component of partitioning, followed by the domain of the image. Assume that a picture I has to be PIFS encoded. The range of an image would be made up of a group of non-overlapping blocks, and the domains would be made up of a group of overlapping blocks, if it were divided into its range and domains [19], as well as the subsequent;

$B_I = B_1, B_2 \dots B_n$ blocks $\forall B_i \cap B_j = \emptyset, i \neq j$; known as the image pool that I , Where ;

$$I = \bigcup_{i=1}^N B_i \quad \dots (2)$$

Next, we have the image's domain blocks pool, which is the set C , , which is the overlapping domain blocks and $C_I = C_1, C_2 \dots C_n$, is known as the image's domain blocks pool I ,

where;

$$I = \bigcup_{i=1}^N C_i \quad \dots (3)$$

Each range pixel in W is assigned the matching domain blocks by the contractive affine transformation ($W_i; i 1,2,3,\dots N$) configuration:

$$W_i : C_i \rightarrow B_i \text{ for } (i =1,2,3 \dots N) \text{ and } B_i = W_i(C_i) = S_i \phi(C_i) + O_i$$

Translating an image is determined by two values: O_i , also called the offset factor, and S_i , which stand for the actual values of the image's contractions function measured in the range of $(0 S_i 1)$ and represent the real constant value of the spatial contractions function that reduces domain blocks to the size of a range block.

A collection is produced using the image coding technique using contractive affine transformation. The encoding process is difficult, and finding it requires a lot of time. In Figure(3), the image I is divided into its component parts. Firstly, the contractive mapping $X_i: C_I \rightarrow B_I$ is applied to each domain block separately. Next, by translating the image's domain CI geometrically, a range BI smaller than the domain block is produced [20]. The change is then documented to the necessary degree of resemblance once each site has a low degree of deviation from the assigned range blocks.

The partition iterated function system (PIFS) [21] must converge to the image's IFS constant point in order for contractive mapping processes to function. Nevertheless, the computationally expensive method of finding and logging such an attractor.

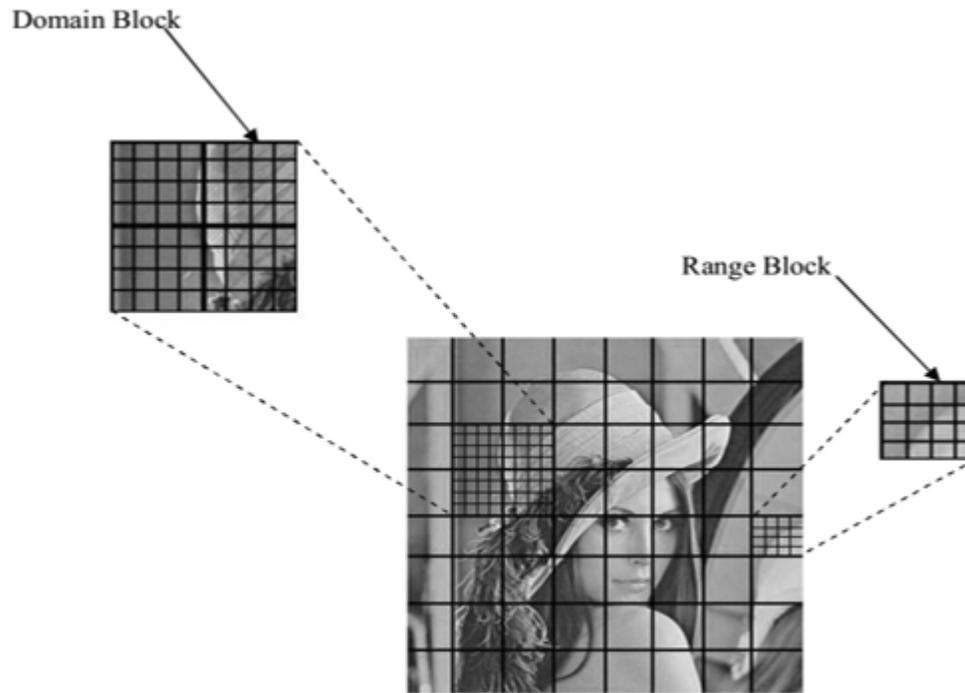


Figure (3): Partitioning mechanism of Range-Domain blocks.

Proposed Algorithm

The proposed **Algorithm** makes use of an adjacency matrix for an efficient data structure. The suggested **Algorithm** aims to decrease the amount of time needed for each iteration of Fractal Image Compress (FIC) encoding in the candidate item set.

First step, separate the inter-image I into non-overlapping sections.

blocks B and to overlapping blocks C

Second step: For every block in the range $B_n \in BI$,

search about the best matching domain blocks where $C_n \in CI$ transform all encoding parameters so that a distance $c(B_i, S_i\phi(C_i) + O_i)$ can be minimized

Third step: compute MSE, S, O

Forth step, Verify, then save the encoded file using the transformation settings (ϕ, S_i, O_i)

Data Set Used in Fractal Image Compression

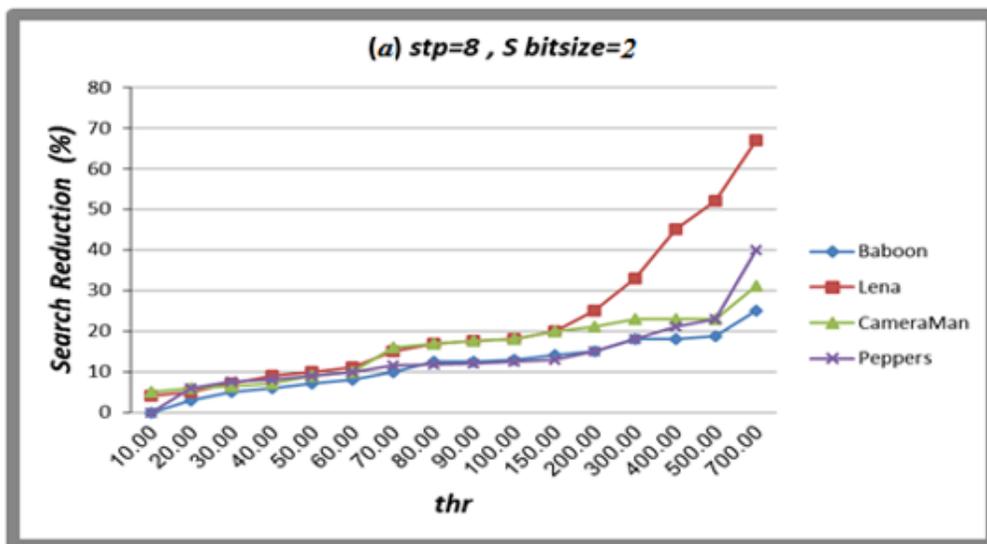
The FIC approach is evaluated using four photos (Lena, CameraMan, Baboon, and Peppers) with varying textures at the grey scale level; each image has 512 x 512 pixels, as can be seen in Figure (4)..



Figure (4): Four typical picture data sets for FIC algorithm evaluation.

Performance Analysis of Proposed FIC Algorithm

After applying Algorithm 1 to the data set in Figure (4), Figure (5) displays the mean enhancement for all the images that were analyzed at various threshold values. a significant improvement over (stp = 8, S bitsize = 2) combined. was found at (stp = 8, S bitsize = 5) when analyzing the search reduction percentage.



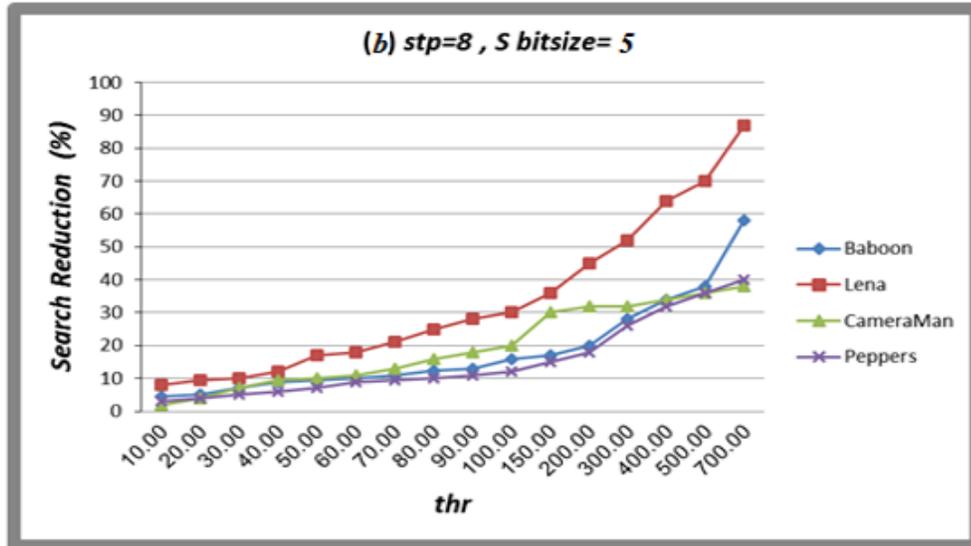


Figure (5): The FIC algorithm's results in terms of percent search reduction with various the-values are demonstrated using four samples of 512×512 greyscale images with an 8×8 range block.

Results of FIC Decoding

The subjective decompression results for the Lena image are shown in Figure (6). This example shows that the decoded image generated by the recommended technique is more visually appealing than the image generated by the traditional search approach. Figure (7) displays the overall average improvement for all assessed image at various thr levels. The amount of progress becomes more obvious as thr is raised. For the ideal combination, the average PSNR improvement at $thr = 700$ is around 1 dB for $r = 8$.



Figure (6): Lena image in grayscale, 512×512 ; (a) original image; (b) image decoding using suggested technique (c) utilizing a common technique to decode an image.

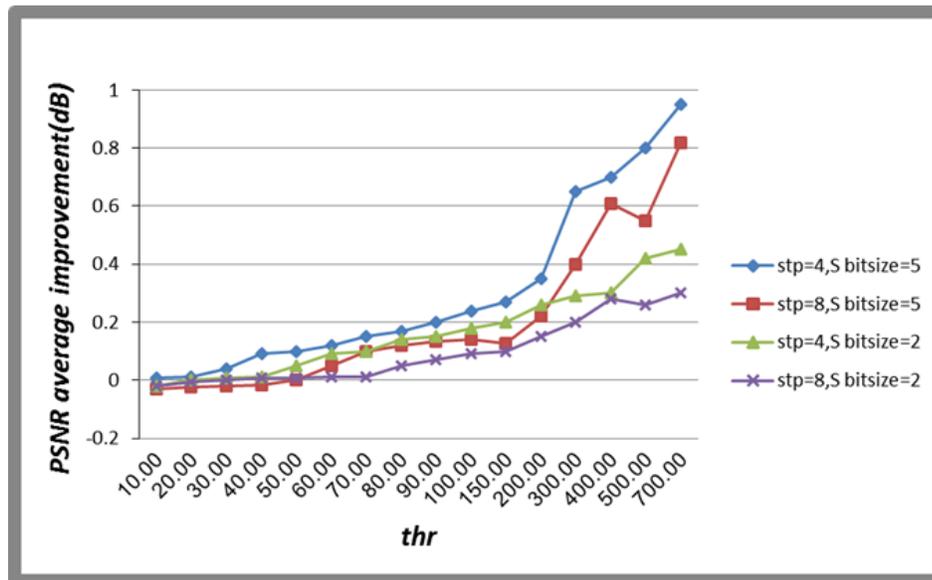


Figure (7): For $r = 8$, the four combinations of S bitsize and stp result in a different average PSNR improvement.

Comparing the Suggested FIC Method's Efficiency with Existing Methods

The parameters of the proposed fractal codes are as follows: $stp = r = 8$, S bitsize = 5, and $thr = 700$. Encoding time, PSNR, and compression ratio (CR) are the three metrics that Table (1) uses to compare the given fractal method to five other existing fractal compression techniques.

Table (1): Comparing the Suggested FIC Method's Performance with Current Approaches

	Proposed Method	[22]	[23]	[24]	[25]	[26]
Platform for Implementation	Intel(R) Core(TM) i5 CPU 1.19GHz	Pentium IV CPU 3GHz	Intel (R) CPU 2.0GHz	Intel(R) Core(TM) i5 CPU 1.19GHz	Celeron 2.66GHz	Intel (R) CPU 2.0GHz
Simulation software	MATLAB program	C#	MATLAB program	MATLAB program	C++	C++
Image information	512 × 512 lena image	512 × 512 lena image	256 × 256 lena image	512 × 512 lena image	256 × 256 lena image	256 × 256 lena image
Method of partitioning	Fixed	Quadtree	Quadtree	Fixed	Quadtree	Fixed
Time Encoding in (sec)	0.309	2.499	56.389	0.359	1.89	6.399
PSNR	30.919	34.89	25.79	30.919	31.199	25.09
Compression ratio (CR)	29.349	5.59	15.59	26.179	11.49	27.099

Enhancement Results

The Lena image was experimented with in the outcomes. The proposed fractal codes included the following parameters: ($thr = 700$, S bitsize = 5, and $stp = r=8$). As demonstrated in Table (2), Enhancing searches leads to a 1.5 dB improvement in image quality and an 89% reduction in the total number of searches needed.

Table (2): Improvements in results compared to the typical greyscale method Image of Lena.

	Typical approach	suggested approach	Amount of improvement
Total Number of Search	285677	31424	89%
PSNR(dB)	29.4	30.92	1.5

Conclusions

Starting at frame compression for the Algorithm's conclusion, the many phases of setting up the FIVC system have been covered. For more effective fractal coding, a modified complete search technique for fast compression has been developed and put into practice. The key requirements for the proposed FIVC system that were taken into account throughout the design phase were the faster encoding speed and smaller memory capacity. This method minimizes the greyscale picture data based on self-similarity and spatial-temporal similarity.

- The following observations are included in the test findings on the modified FIC:
 1. Enhanced FIC performs better than normal FIC, resulting in an 89% reduction in the total number of searches required and a 1.5 dB improvement in image quality. With an encoding time of 0.31 seconds, the obtained compression ratio is around 29.35 for PSNR = 30.92 dB..
 2. The optimal set of parameters selected to yield a favorable outcome is (stp = 8 and S bitsize = 5). Since the search reduction ratio regulates the increase in encoding speed, it serves as the foundation for the comparison.

Ideas for Further Work

From the numerous thoughts generated by the discussion of the test results, the following recommendations are put forth as possible topics of investigation:

1. The results show how well the recommended algorithms work to speed up the compression of photos and movies. Another kind of fractal compress algorithm based on the intelligent system may be used to enhance the compress problem and compare it with the recommended strategies.
2. Real-time image compression coding could benefit from additional development of the suggested techniques.
3. The suggested approach could be broadened to use high-performance FPGA in fractal coding of color images and video sequences, as a response to the increasing need for superior coder/decoder techniques.

References:

1. M. F. Barnsley and L. P. Hurd, "Fractal image compression."
2. Y. Fisher, "Fractal image compression," *Fractals*, vol. 2, no. 03, pp. 347–361, 1994.
3. B. Wohlberg and G. De Jager, "A review of the fractal image coding literature," *IEEE Trans. Image Process.*, vol. 8, no. 12, pp. 1716–1729, 1999.
4. C. E. Martin and S. A. Curtis, "Fractal image compression," *J. Funct. Program.*, vol. 23, no. 6, pp. 629–657, 2013.
5. S. Liu, W. Bai, N. Zeng, and S. Wang, "A fast fractal based compression for MRI images," *Ieee Access*, vol. 7, pp. 62412–62420, 2019.
6. S. Kadam and V. R. Rathod, "Medical image compression using wavelet-based fractal quad tree combined with Huffman coding," in *third international congress on information and communication technology*, 2019, pp. 929–936.

7. Abood, Baydaa Sh Z., and Hanan AR Akkar. "Review on Fractal Video and Image Compression Techniques." *IRAQI JOURNAL OF COMPUTERS, COMMUNICATIONS, CONTROL AND SYSTEMS ENGINEERING* 23.1 (2023).
8. W. Pan, Y. Zou, and L. Ao, "A compression algorithm of hyperspectral remote sensing image based on 3-D wavelet transform and fractal," in *2008 3rd International Conference on Intelligent System and Knowledge Engineering*, 2008, vol. 1, pp. 1237–1241.
9. A. E. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations," *IEEE Trans. image Process.*, vol. 1, no. 1, pp. 18–30, 1992.
10. S. Mozaffari, K. Faez, and M. Ziaratban, "Character representation and recognition using quad tree-based fractal encoding scheme," in *Eighth International Conference on Document Analysis and Recognition (ICDAR'05)*, 2005, pp. 819–823.
11. S. Kiani and M. E. Moghaddam, "A multi-purpose digital image watermarking using fractal block coding," *J. Syst. Softw.*, vol. 84, no. 9, pp. 1550–1562, 2011.
12. Shah, Tawheed Jan, and M. Tariq Bandy. "A review of contemporary image compression techniques and standards." *Examining Fractal Image Processing and Analysis* (2020): 121-157.
13. M. Polvere and M. Nappi, "Speed-up in fractal image coding: comparison of methods," *IEEE Trans. Image Process.*, vol. 9, no. 6, pp. 1002–1009, 2000.
14. R. Hamzaoui and D. Saupe, "Fractal image compression," *Doc. Image Compression*, no. January, pp. 145–175, 2006, doi: 10.1142/s0218348x94000442.
15. Y. Fisher and A. Lawrence, "Fractal image encoding," NETROLOGIC INC SAN DIEGO CA, 1992.
16. A. E. Jacquin, "Fractal image coding based on a theory of iterated contractive image transformations," in *Visual Communications and Image Processing'90: Fifth in a Series*, 1990, vol. 1360, pp. 227–239.
17. M. Ruhl, H. Hartenstein, and D. Saupe, "Adaptive partitionings for fractal image compression," in *Proceedings of International Conference on Image Processing*, 1997, vol. 2, pp. 310–313.
18. M. Mokhtari, A. Boukelif, A. Gabis, and A. Naimi, "Optimization of fractal image compression based on Kohonen neural networks," 2006.
19. A. Thesis, "Design of Intelligent Controller for Solar Tracking System Based on," no. July, 2014.
20. G. R. Sinha, "USING DISTRIBUTED ARCHITECTURE AND SIMD APPROACH," no. January, 2016, doi: 10.13140/RG.2.1.3162.1202.
21. M. F. Barnsley, *Fractals everywhere*. Academic press, 2014.
22. T. Kovács, "A fast classification based method for fractal image encoding," *Image Vis. Comput.*, vol. 26, no. 8, pp. 1129–1136, 2008.
23. X.-Y. Wang and D.-D. Zhang, "Discrete wavelet transform-based simple range classification strategies for fractal image coding," *Nonlinear Dyn.*, vol. 75, no. 3, pp. 439–448, 2014.
24. Abood, Baydaa Sh Z., Hanan AR Akkar, and Amean Sh Al-Safi. "Fast Full-Search Algorithm of Fractal Image Compression for Acceleration Image Processing." *Science Journal of University of Zakho* 11.1 (2023): 117-124.
25. X.-Y. Wang, F.-P. Li, and Z.-F. Chen, "An improved fractal image coding method," *Fractals*, vol. 17, no. 04, pp. 451–457, 2009.
26. W. Xing-Yuan, Z. Dou-Dou, and W. Na, "Fractal image coding algorithm using particle swarm optimisation and hybrid quadtree partition scheme," *IET image Process.*, vol. 9, no. 2, pp. 153–161, 2015.