Comparison of Fractal Compression Methods of Images

Tolaniddin Nurmukhamedov[1,2], Bakhodir Achilov[1,2] and Odilzhan Turdiev[1,2]
[1] *Tashkent State Transport University, Temiryolchilar Str. 1, 100167 Tashkent, Uzbekistan*
[2]*University of Diyala, 32009 Baqubah, Diyala, Iraq*
ntolaniddin@mail.ru, borya19861804@gmail.com, odiljan.turdiev@mail.ru

Keywords: Integrated Functional Systems, Wavelet Image Compression, Rank Block, Domain Block, Self-Similarity, Wavelet Compression, Wavelet Transformation, Quadtree.

Abstract: This article is a report on the results of a comparative analysis of various most commonly used methods of fractal image compression: actually, fractal compression, compression using iterated functional systems, compression based on quad trees. Nowadays, when huge amounts of data are generated daily, efficient image compression techniques play an important role in reducing the required storage space and transmission bandwidth. Fractal compression, a relatively new approach, attracts attention due to its ability to compress images with minimal loss of quality. Therefore, when comparing the above compression methods, the following criteria were used: compression ratio, processing efficiency (productivity), and the quality of the images obtained. The article also discusses the basic principles of fractal compression, its advantages and disadvantages compared to traditional methods such as JPEG and PNG. Special attention is paid to the analysis of various fractal compression algorithms, their application and performance. The authors of the article strive to identify the most effective methods that provide a high degree of compression while maintaining the maximum amount of information about the image. This analysis can be useful for developers, engineers and researchers involved in image and data processing, as well as for a wide range of readers interested in advanced technologies in the field of digital data processing.

1 INTRODUCTION

Nowadays, when a huge number of images are created and processed, effective compression methods play an important role in reducing file size and improving data transfer. Fractal image compression is a relatively new approach that is attracting increasing attention for its ability to compress images with minimal loss of quality [1].
Fractal compression is based on the principle of self-similarity, which is found in many natural images. Fractal compression encodes similar patterns and structures in images. This approach allows you to achieve high compression ratios while preserving the details of the original images. Traditional methods such as JPEG and PNG are widely used for image compression, but they have their limitations, especially in terms of quality and compression ratio [2]. Based on the mathematical theory of fractals, they offer an alternative approach to this problem. These methods use self-similarity of image structures to represent it in a compressed form. The basic idea is to find small sections of the image that can be approximated by scaled and rotated copies of other sections of the same image. Thus, instead of storing all information about each pixel, only information about transformations is stored, which significantly reduces the amount of data [3].

Fractal compression methods have a number of unique advantages, such as a high compression ratio and the ability to infinitely enlarge an image without loss of quality. However, they also have their drawbacks, including high computational complexity and long compression times. The introduction to this topic allows you to understand more deeply the principle of fractal methods, their advantages and disadvantages, as well as compare them with traditional image compression methods.

As a result, the studying comparison of fractal image compression methods opens up new prospects for more efficient image storage and transmission.

2 METHODS AND RESULTS

In this article, we will focus on comparing different fractal image compression methods. We will consider fractal image compression based on partitioning the image into rank and domain blocks, based on Iterative Functional Systems (IFS), based on quadtrees (Quadtree) and fractal wavelet image compression. In the comparison process, we will evaluate their effectiveness in achieving high compression ratios and maintaining image quality.

Next, we will present a detailed analysis of each method, evaluating their performance, including compression ratio, computational complexity, and image quality. The results of the analysis will help determine the most effective fractal compression method for various tasks and applications.

2.1 Fractal Image Compression Based on Rank and Domain Blocks

In general terms, fractal compression can be divided into two stages:

- Splitting an image into a set of non-overlapping rank blocks and into a set of domain blocks (which may overlap each other);
- Applying transformations for each domain–rank block pair.

A large area is called a domain block, and a smaller one is called a rank block. The conversion of a domain block to a rank block is carried out by affine transformations, which, in the case of a grayscale image, can be represented by the following system of (1) [4]:

$$\begin{bmatrix} x^* \\ y^* \\ z^* \end{bmatrix} = \begin{bmatrix} a & b & 0 \\ a & d & 0 \\ 0 & 0 & u \end{bmatrix} * \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} e \\ f \\ v \end{bmatrix}, \quad (1)$$

were,

- x, y, z – are the coordinates and brightness of the pixel of the domain block, respectively;
- x^*, y^*, z^* – are respectively the coordinates and brightness of the pixel of the rank block;
- a, b, c, d, e, f – coefficients of coordinate transformations on the plane;
- u – brightness compression ratio;
- v – is the brightness shift.

The quality of compression depends on the partitioning scheme used in the first stage. The more domain blocks there are, the greater the chance of finding the most similar ranked blocks. At the second stage, it is necessary to transform the domain block so that it is as similar as possible to the rank block. The general formula for converting pixel values of a domain block is as follows [5]:

$$D_i^* = sD_i + o, \quad (2)$$

where, D_i^* and D_i - are the transformed and original i-th domain block, respectively; o – is the brightness shift coefficient; $|s| < 1$ is the contrast coefficient, $o \in [-255; 255]$ – is the brightness coefficient. In addition to the direct conversion of pixel values according to formula (1), the domain block can also be subjected to general scaling (reducing the size to the size of a rank block, for example, by interpolation or simple thinning), rotation and other affine transformations [6].

The transformed domain block should match the rank block as strongly as possible, since this is how the rank block will be restored during decoding. To estimate the discrepancy (distance) between the transformed domain and the given rank blocks, it is necessary to enter the appropriate metric. The standard deviation function is usually used:

$$Q = \sum_{i=1}^{N}(D_i^* - R_i)^2 = \sum_{i=1}^{N}((sD_i + o) - R_i)^2, \quad (3)$$

where, R_i- i th rank block, D_i^* and D_i-are the transformed and original i th and original with domain block, respectively, N-is the number of pixels in the rank block.

Obviously, the smaller the distance (2) between the blocks, the more similar they are. The coefficients and o can be found from (2) by taking partial derivatives of these variables.

Let's open the square in expression (4):

$$Q = \sum_{i=1}^{N}(s^2D_i^2 + 2soD_i + o^2 - 2sR_iD_i - 2oR_i + R_i^i). \quad (4)$$

We have the following condition:

$$\begin{cases} \frac{\partial Q}{\partial s} = \sum_{i=1}^{N}(2sD_i^2 + 2D_io - 2R_iD_i) \\ \frac{\partial Q}{\partial o} = \sum_{i=1}^{N}(sD_i + o - R_i) \end{cases}. \quad (5)$$

From (5) above, we obtain (6). Both equations are correct and necessary to find the extremum points. Thus, we solve the equation:

$$\begin{cases} s\sum_{i=1}^{N} D_i^2 + o\sum_{i=1}^{N} D - \sum_{i=1}^{N} R_iD_i = 0 \\ \sum_{i=1}^{N}(sD_i + o - R_i) = 0 \end{cases}. \quad (6)$$

Let's express the shift in brightness:

$$o = \frac{1}{N}\sum_{i=1}^{N} R_i - \frac{1}{N}s\sum_{i=1}^{N} D_i. \quad (7)$$

Substitute (6) into the (7) of the partial derivative and obtain the following formulas for finding the coefficients:

$$\begin{cases} s = \frac{N\sum_{i=1}^{N} R_iD_i - \sum_{i=1}^{N} R_i \sum_{i=1}^{N} D_i}{N\sum_{i=1}^{N} D_i^2 - \left(\sum_{i=1}^{N} D_i\right)^2}; \\ o = \frac{1}{N}\left(\sum_{i=1}^{N} R_i - s\sum_{i=1}^{N} D_i\right). \end{cases} \quad (8)$$

By converting (4), we obtain an expression for finding the distance:

$$Q = s^2 \sum_{i=1}^{N} D_i^2 + No^2 + \sum_{i=1}^{N} R_i^2 - 2s \sum_{i=1}^{N} R_i D_i + 2so \sum_{i=1}^{N} R_i - 2o \sum_{i=1}^{N} R_i. \quad (9)$$

Formulas (6), (7) simplify the computational load, since the sums $\sum_{i=1}^{N} R_i$, $\sum_{i=1}^{N} R_i^2$, $\sum_{i=1}^{N} D_i$, $\sum_{i=1}^{N} D_i^2$ can be calculated even before the enumeration begins, when sets of rank and domain blocks have already been formed [7]. Then, at the matching stage, you will need to calculate only the sum of $\sum_{i=1}^{N} R_i D_i$ and find the coefficients [8].

2.2 Integrated Functional Systems

IFS is one of the first and most well—known methods of fractal compression. This method represents an image using a set of compressive transformations that are applied iteratively to create self-similar patterns. By finding suitable compression transformations for each block, IFS can efficiently encode images. However, IFS-based methods often suffer from high encoding and decoding complexity, which limits their practical use [9].

Hutchinson (1981) showed that for the metric space R^n or, more generally, for the entire metric space X such a system of functions has a single nonempty compact (closed and bounded) fixed set S. One way to construct a fixed set is to start with the initial non–empty closed and bounded set S_0 and repeat the steps f_i, taking S_{n+1} as the union of images S_n under f_i; then taking S as the closure of the union S as the closure of the union S_n. A symbolically unique fixed (nonempty compact) set $S \subseteq X$ has the properties [10].

$$S = \bigcup_{i=1}^{N} f_i(A). \quad (10)$$

Thus, the set S is a fixed set of the Hutchinson operator $F: 2^X \to 2^X$ is defined for $A \subseteq X$ by formula (11)

$$F(A) = \bigcup_{i=1}^{N} f_i(A). \quad (11)$$

The existence and uniqueness of S is a consequence of the compression mapping principle, as is the fact that

$$\lim_{n \to \infty} F^{on}(A) = S, \quad (12)$$

for any nonempty compact A in X. (For compressive IFS, this convergence holds even for any nonempty closed bounded set A). Random elements arbitrarily close to S can be obtained using the "chaos game" described below [11].

It has recently been shown that IFS of a non-compressive type (i.e., composed of maps that are not compressions with respect to any typologically equivalent metric in X) can produce attractors. They arise naturally in projective spaces, although the classical irrational rotation on a circle can also be adapted [12]

2.3 Fractal Compression Based on Quadtree

Fractal compression algorithms based on a quadrant tree divide an image into smaller rectangular blocks, creating a hierarchical structure known as a quadrant tree [13]. The blocks are then recursively encoded and decoded based on their self-similarity. This method offers a good compromise between compression efficiency and computational complexity. Some variants of quadrant tree-based methods include fractal coding with block classification and adaptive subdivision of the quadrant tree [14].

Quadrant trees that store information about lines (English: edge quadtree) are used to describe straight lines and curves. The curves are described by exact approximations by dividing the cells into very small ones [15]. This can lead to unbalancing of the tree, which will mean problems with indexing.

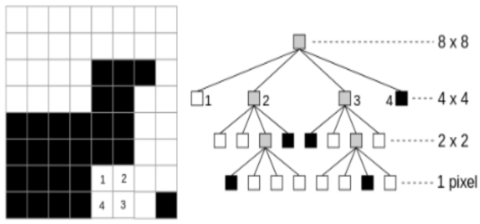

Figure 1: Quadrant trees divide the image into smaller rectangular blocks.

2.4 Fractal Wavelet Image Compression

Fractal wavelet compression combines the principles of wavelet transform and fractal image compression based on rank and domain blocks. The method uses wavelet decomposition to capture image details at various scales, and then applies fractal coding techniques for residual or low-frequency components. This hybrid approach often yields better compression ratios than purely fractal compression methods, while maintaining good image quality [16].

Using the proposed methods of wavelet and fractal analysis in relation to a time series represented by numerical values, it is necessary to obtain quantitative and qualitative estimates characterizing the unsteadiness or randomness of the process under study.

The main processing algorithm is based on the condition of the existence of a time series, that is, statistical material collected at different points in time about the value of any parameters of the process under study, which takes into account the relationship of measurements with time [17].

The first stage is the processing of the time series by the method of wavelet and fractal analysis. The combined application of these two analyses will provide more complete information about the process under study, described by time series [18].

In the work, the wavelet transform is performed using direct WT (Wavelet transformation) [19]:

$$WT_s(a,x) = \frac{1}{a}\int_{-\infty}^{\infty} \psi\left(\frac{t-x}{a}\right) s(t)dt \quad (13)$$

$$\int_{-\infty}^{\infty} \psi(t)dt\, 0, \int_{-\infty}^{\infty} t^m \psi(t)dt\, 0;\ 0 \le m \le n \quad (14)$$

where a – is the scale parameter; x – is the shift the parent wavelet; $s(t)$ – the original signal; $\psi(t)$ – the wavelet function [20].

The calculation of the fractal dimension in this study is performed by the point-by-point method [21]. The algorithm for calculating the pointwise dimension is currently well-known and is a classic one (Fig. 1) [22].

The obtained values of the fractal dimensions of the time series under study are used as a criterion for the stability of the process under study. By the magnitude of the fractal dimension, it is possible to obtain a quantitative assessment of the randomness of the process under study, as well as the multifactorial and "saturation" of the prerequisites that caused the non-stationarity of the process [23].

The second and final stage of the model algorithm is the construction of a predictive assessment based on the allocation of indicators tending to an unstable state and determining the unsatisfactory state of the process. The results can be visualized and presented in the form of constructed wavelet diagrams, which give a visual picture of the most dramatically changing components of the signal. Thus, it becomes possible to identify the most non-stationary areas and, according to the selected areas, analyze in more detail the causes of these non-stationary events in the signal in relation to various applied areas [24].

2.5 Comparative Analysis

When comparing fractal compression methods, compliance with the requirements and limitations of applied tasks plays an important role. In particular, to solve the problems of image compression used in remote sensing (remote sensing of the Earth) [25], important factors are: the degree of compression [26], computational complexity, image quality and noise resistance and minimal information loss during transmission.

As a result of the research, it was found that the most suitable methods of fractal image compression for solving remote sensing problems are fractal compression methods based on rank and domain blocks and fractal wavelet compression.

The following is an illustration of the results of studies on all four methods confirming this conclusion.

2.6 Results

This section evaluates four key fractal image compression techniques:

- Fractal image compression based on rank and domain blocks;
- Iterated Functional Systems (IFS);
- Fractal compression based on Quadtree;
- Compression of the fractal wavelet image.

Fractal image compression based on rank and domain blocks.

Fractal image compression based on rank and domain blocks turned out to be one of the satisfactory results.

From the presented Figure 2 and Figure 3 you can see the preservation of image quality and reduction of compression time.

Integrated Functional Systems (IFS). Fractal compression of real images based on the method (IFS) is less suitable than fractal image compression of L-systems.

In these Figure 4 and Figure 5, the reduction in compression time is not recorded.

Figure 6 and Figure 7 show a significant loss of quality (the compressed image became "blurred"), and the time reduction turned out to be insignificant.

Fractal wavelet image compression. Fractal wavelet compression of the image also gave satisfactory results.

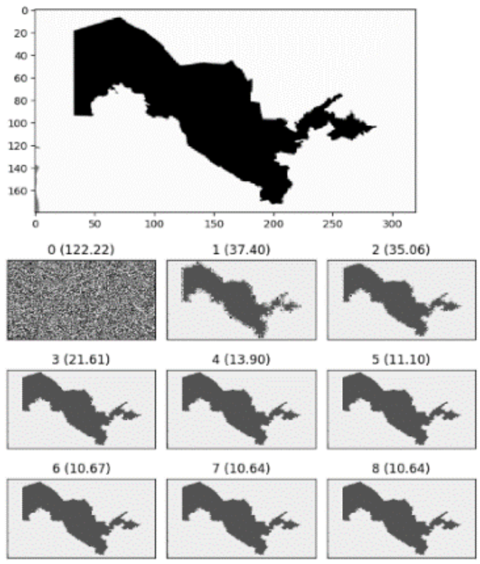

Figure 2: Image compression based on rank and domain blocks.

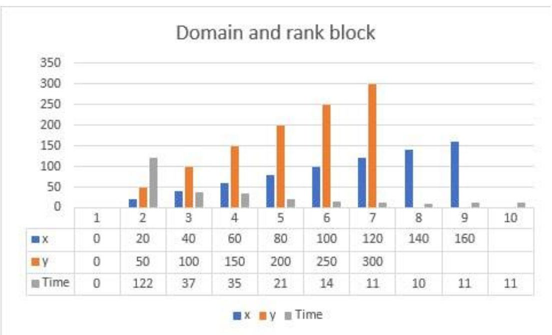

Figure 3: Graph ratio over time compression

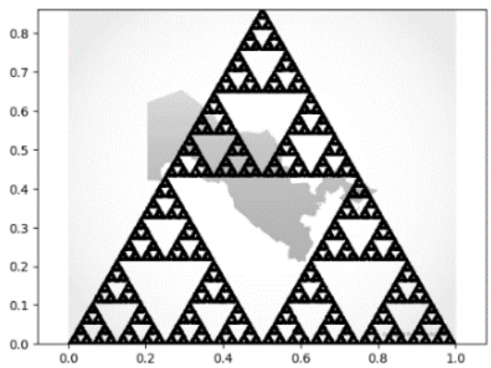

Figure 4: Image Compression Functional Systems (IFS).

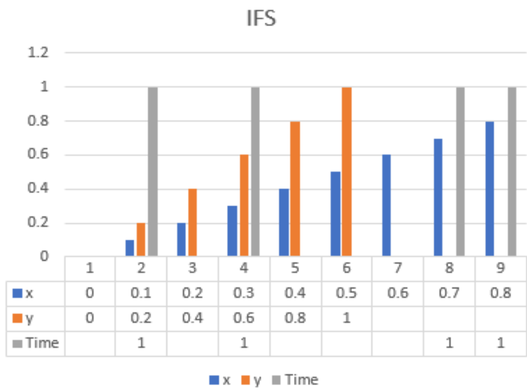

Figure 5: Graph the ratio over time compression.

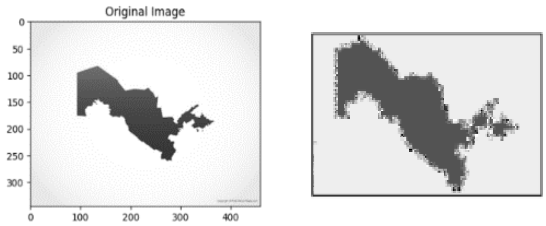

Figure 6: Quadtree-based image compression.

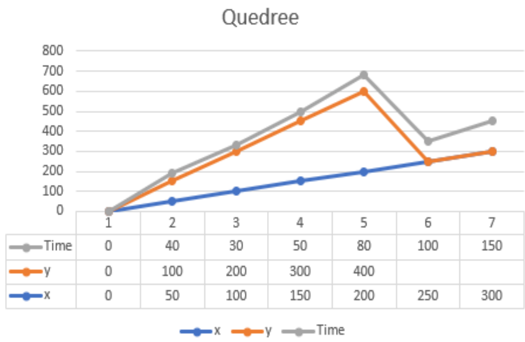

Figure 7: Graph ratio over time compression.

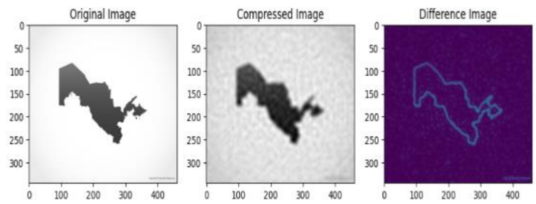

Figure 8: Fractal wavelet image compression.

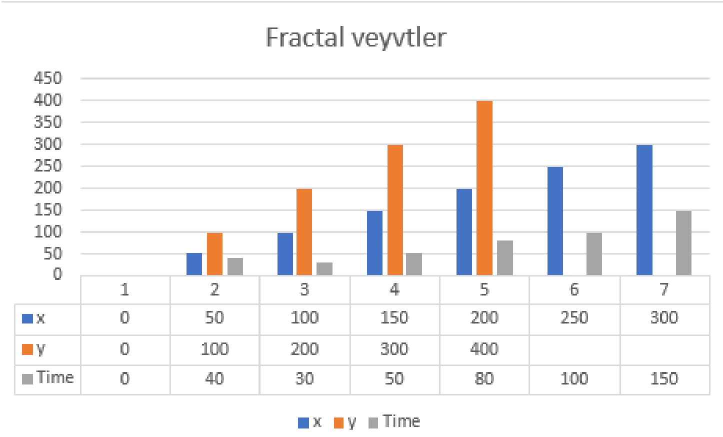

Figure 9: Graph ratio over time compression.

In Figure 8 and Figure 9, as well as in the case of fractal compression based on rank and domain blocks, preservation of image quality and reduction of compression time are recorded.

3 CONCLUSIONS

In conclusion, fractal image compression methods represent a promising direction in the field of processing and storing visual information. Each of the considered methods - based on the quadrant tree, rank and domain blocks, the IFS method and fractal wavelet compression - has its own unique advantages and disadvantages, which makes them suitable for various applications and conditions. The high compression ratios achieved by these methods make them particularly attractive for use in conditions of limited data storage and transmission resources.

Nevertheless, further research aimed at improving algorithms and reducing computational complexity is needed for wider implementation of fractal methods. Optimizing the compression time and improving the quality of the restored image are key areas that require the attention of researchers. It is also important to take into account the specific requirements of various applications, such as medical imaging, satellite imagery or multimedia, in order to adapt compression methods to specific needs.

In the future, fractal compression methods can become an integral part of new technologies, providing efficient and high-quality image compression. With the development of computing power and algorithmic improvements, it can be expected that these methods will find wider and wider application, contributing to progress in various fields of science and technology.

ACKNOWLEDGMENTS

The results of this research work were obtained within the framework of the practical project No. IL-4821091604. We will express our gratitude to the Agency for Innovative Development under the Ministry of Higher Education, Science and Innovation, which supports the project.

REFERENCE

[1] B. Bondžulić, N. Stojanović, V. Lukin, S.A. Stankevich, D. Bujaković, and S. Kryvenko, "Target acquisition performance in the presence of JPEG image compression," Defence Technology, vol. 33, pp. 30-41, 2024, [Online]. Available: https://doi.org/10.1016/j.dt.2023.12.006.

[2] G. Roelofs, "PNG Lossless Image Compression," in Communications, Networking and Multimedia, pp. 371-390, 2003, [Online]. Available: https://doi.org/10.1016/B978-012620861-0/50020-6.

[3] R. Gonzalez, R. Woods, and S. Eddins, Digital Image Processing in the MATLAB Environment. Technosphere, 2006.

[4] S.R. Nayak, J. Nayak, K. Muhammad, and Y. Karaca, "Intelligent Fractal-Based Image Analysis," in Cognitive Data Science in Sustainable Computing, Academic Press, pp. 167-193, 2024, [Online]. Available: https://doi.org/10.1016/B978-0-44-318468-0.00015-5.

[5] H. Wen, Y. Huang, and Y. Lin, "High-quality color image compression-encryption using chaos and block permutation," Journal of King Saud University - Computer and Information Sciences, vol. 35, no. 8, p. 101660, 2023.

[6] S. Liu, L. Chen, and B. Feng, "A theoretical method for scaling the dynamic response of steel beams subjected to blast loads," International Journal of Impact Engineering, vol. 187, p. 104918, 2024, [Online]. Available: https://doi.org/10.1016/j.ijimpeng.2024.104918.

[7] X. Li et al., "A self-supervised feature-standardization-block for cross-domain lung disease classification," Methods, vol. 202, pp. 70-77, 2022, [Online]. Available: https://doi.org/10.1016/j.ymeth.2021.05.007.

[8] M. Khudaiberdiev, T. Nurmukhamedov, and B. Achilov, "Fractal image analysis of tomato leaf texture in the context of viral tomato mosaic disease," Scientific and Technical Journal of FerPI, vol. 28, no. 2, 2024.

[9] Q.-R. Deng, M.-T. Li, and Y.-H. Yao, "On the connected components of IFS fractals," Journal of Mathematical Analysis and Applications, vol. 518, no. 2, p. 126785, 2023.

[10] R. Anjum, M. Din, and M. Zhou, "Fractals of two types of enriched (q,θ)-Hutchinson–Barnsley operators," Chaos, Solitons & Fractals, vol. 181, p. 114589, 2024, [Online]. Available: https://doi.org/10.1016/j.chaos.2024.114589.