

## **ДОСЛІДЖЕННЯ МЕТОДУ ФРАКТАЛЬНОГО СТИСНЕННЯ ЗОБРАЖЕНЬ З МЕТОЮ ПОКРАЩЕННЯ ЯКОСТІ СТИСНЕННЯ**

*Анотація. Розвиток мережі Internet, поряд з доступністю все більш потужних комп'ютерів та інших цифрових пристроїв, камер, сканерів та принтерів, привели до широкого використання цифрових зображень. У зв'язку з цим зростає інтерес до поліпшення алгоритмів стиснення даних, наприклад зображень.*

*Стиснення даних є актуальним як для швидкості передачі, так і ефективності зберігання. Крім багатьох видів комерційного використання, технології стиснення становлять також інтерес у військовій галузі, наприклад, програми обробки даних телеметрії, отриманих від перехоплювачів ракет, або для архівного зберігання даних про зображення місцевості для моделювання оборонних дій.*

*Вирішення проблеми стиснення зображення або, в загальному сенсі, кодування зображення, використовувало досягнення та стимулювало розвиток багатьох галузей техніки та математики.*

*В статті досліджується фрактальне стиснення зображень — метод стиснення даних, що ґрунтується на використанні самоподібних патернів у зображенні. Цей метод дозволяє досягти високого ступеня стиснення за умови збереження деталей зображення.*

*Ключові слова: фрактал, фрактальне стиснення, самоподібність, афінні перетворення, коефіцієнт стиснення, декомпресія.*

**Постановка задачі.** Фрактальне стиснення зображень є унікальним і ефективним підходом до стиснення даних, засноване на математичній теорії фракталів. У сучасності, це має важливі сфери застосування та переваги, які роблять його цінним інструментом у галузі обробки зображень. До основних переваг відносять:

1. Збереження деталей при стисканні. Однією з ключових переваг фрактального стиснення є його здатність зберігати високий ступінь деталізації зображень за відносно низького обсягу зберігання. Це особливо важливо для ситуацій, де потрібно зберегти якість зображення за обмежених ресурсів для зберігання та передачі даних.

2. Ефективність передачі через мережу. Фрактальне стиснення дозволяє створювати компактні зображення, що робить його придатним для передачі зображень через мережу з низькою пропускнуою здатністю. Це особливо актуально для мобільних пристроїв, Інтернету речей та інших сценаріїв, де широка смуга пропускання не завжди доступна.

3. Адаптивне стиснення для різних роздільних здатностей. Фрактальне стиснення дозволяє адаптувати рівень стиснення залежно від роздільної здатності та деталей зображення. Це означає, що воно може бути використане для стиснення різних розмірів зображень без значної втрати якості.

4. Архівування та зберігання даних. Фрактальне стиснення може бути корисним для архівування та довгострокового зберігання зображень, так як воно дозволяє ефективно зменшити обсяг даних, не втрачаючи важливої інформації. Це особливо важливо для бібліотек, архівів, дослідницьких баз даних та інших сховищ даних.

Фрактальне стиснення зображень залишається актуальним і цінним інструментом у сучасних умовах, завдяки своїм здібностям до ефективного стиснення, збереження деталей та адаптивності до різних сценаріїв використання. Тому дослідження його ефективності, оптимізація програмного коду для отримання більш швидкого та якісного результату стиснення, є актуальною задачею.

**Аналіз останніх досліджень і публікацій.** Фрактальне стиснення є інноваційним підходом до стиснення даних, заснований на використанні самоподібних фрактальних структур. Концепція фракталів, вперше запропонована Бенуа Мандельбротом у 1975 році, застосовується у різних галузях, включаючи геометрію, фізику та комп'ютерну графіку [1]. В останні десятиліття фрактальне стиснення стало однією з найважливіших технік стиснення зображень, що надає високий рівень стиснення з мінімальною втратою інформації.

Фрактальне стиснення відрізняється від традиційних методів стиснення, таких як JPEG або MPEG, які ґрунтуються на зменшенні надмірності даних. Натомість, фрактальне стиснення досліджує і витягує самоподібні візерунки зображення, що дозволяє представити його з використанням відносно невеликої кількості даних.

Фрактальне стиснення зображень має чудові властивості, які роблять його привабливим для дослідження і використання в обробці зображень. В багатьох наукових роботах досліджені головні властивості цього методу [2-5].

1. Однією з головних переваг фрактального стиснення є здатність зберігати дрібні деталі зображення. Традиційні методи стиснення, як правило, призводять до втрати високочастотних компонентів зображення, що призводить до втрати деталей і розмиття зображення. Фрактальне стиснення дозволяє відновити ці деталі, що робить його особливо корисним для додатків, що вимагають високого ступеня деталізації.

2. Фрактальне стиснення зображень ґрунтується на понятті самоподібності, де фрагменти зображення можуть бути точно або приблизно повторені в інших частинах зображення. Ця властивість забезпечує ефективне кодування та подання зображення з використанням обмеженого набору фрактальних операторів.

3. Фрактальне стиснення також знайшло застосування в цифруванні та передачі зображень за низької швидкості передачі даних. Його здатність представляти зображення з високим ступенем стиснення робить його зручним для використання в системах з обмеженою пропускнуою здатністю.

**Мета дослідження** - дослідження методу фрактального стиснення зображень з використанням варіативності вхідних даних, оптимізація програмного коду для покращення якості стиснення.

#### **Викладення основного матеріалу дослідження.**

Фрактальний метод стиснення зображень з використанням афінних перетворень - це один із способів стиснення зображень, заснований на ідеях фракталів та самоподібності [6]. Цей метод дозволяє ефективно кодувати зображення шляхом опису їх структурних властивостей за допомогою набору афінних перетворень.

#### *Логіка роботи методу*

1. Декомпозиція зображення: початкове зображення поділяється на невеликі блоки - "блоки пікселів" (наприклад, прямокутник розміром 2×2 або 4×4 пікселя). Кожен блок сприймається як набір точок, які мають деяку локальну структуру.

2. Вибір базових перетворень: для кожного блоку пікселів шукаються афінні перетворення (масштабування, зсув, поворот), які найкраще підходять для наближення цього блоку. Ці перетворення обираються так, щоб мінімізувати помилку наближення.

3. Кодування: кожен блок пікселів кодується з використанням вибраного перетворення та коефіцієнтів, що описують різницю між вихідними значеннями пікселів і значеннями, отриманими в результаті перетворення. Таким чи-

ном, інформація про структуру зображення кодується у вигляді набору афінних перетворень та відповідних різниць.

4. Рекурсивна обробка: процес повторюється рекурсивно для кожного блоку, включаючи різницеві дані, отримані на попередньому етапі. Це дозволяє створювати більш детальні описи зображення за допомогою набору афінних перетворень і різниць кожному рівні рекурсії.

5. Кодування структури: створюється код, що містить інформацію про структуру зображення, афінні перетворення і коефіцієнти різниць. Цей код може бути стиснутий далі з використанням стандартних методів стиснення даних.

6. Декодування: афінні перетворення застосовуються у зворотному порядку, починаючи з внутрішнього рівня рекурсії, і коефіцієнти різниць додаються назад до значень пікселів.

Перед стисненням, зображення перетворюється на чорно-біле. Це необхідно для:

1. Спрощення алгоритму. Зображення у відтінках сірого (чорно-білі) містять лише один канал яскравості замість трьох каналів кольору (червоного, зеленого та синього). Це спрощує алгоритми стиснення та зменшує кількість даних, з якими потрібно працювати.

2. Збереження деталей. Чорно-білі зображення мають менше інформації, ніж кольорові зображення, і, отже, можуть бути більш ефективно стиснуті з мінімальною втратою деталей.

#### *Приклад роботи методу та аналіз результатів*

На рисунку 1 наведено приклад роботи фрактального методу стиснення зображень. У якості вхідного зображення взято зображення формату jpg з роздільною здатністю 555x370 пікселів (програмно змінене на 256x256) та розміром 113 КБ.

У результаті роботи фрактального методу стиснення зображень доступна наступна інформація:

Time taken for compression = 0.291021 seconds

Compression ratio = 3.329912

Time taken for decompression = 0.457792

PSNR = 26.724437



Рисунок 1 – Результат роботи програми стиснення зображення: вхідне зображення (ліворуч), розкладання за допомогою квадродерева (по центру), відновлене зображення (праворуч)

Time taken for compression - це час, витрачений на стиснення зображення методом афінних перетворень. Значення визначається в секундах (з точністю до мілісекунд) і показує скільки часу зайняло виконання алгоритму стиснення.

Compression ratio – це коефіцієнт стиснення, який показує, наскільки зменшився розмір зображення після стиснення. Він розраховується як відношення кількості бітів у вихідному зображенні до кількості бітів після стиснення. У даному випадку значення коефіцієнта стиснення дорівнює 3.329912, що означає, що розмір стисненого зображення зменшився приблизно в 3.32 разів у порівнянні з оригінальним зображенням.

Time taken for decompression – це час, витрачений на декомпресію (відновлення зображення після стиснення). Також вимірюється в секундах.

PSNR (Peak Signal to Noise Ratio) – використовується для вимірювання якості відновленого зображення в порівнянні з оригіналом. Він розраховується як відношення максимально можливого значення сигнал-шум до квадрата середньоквадратичної помилки між оригінальним та відновленим зображеннями. Чим вище PSNR, тим краще якість відновленого зображення. У даному випадку значення PSNR дорівнює 26.724437, що вказує на задовільну якість відновленого зображення.

#### *Оптимізація програмного коду для покращення якості стиснення*

Проведено дослідження, наскільки сильно вплине на швидкодію програмного продукту експериментальне втручання та зміна програмного коду для досягнення кращої або гіршої якості зображення.

1. Погіршення якості зображення. У кодї, щоб отримати зображення найгіршої якості, були внесені наступні зміни. У рядку  $s = \text{qtdecomp}(I, 0.1, [216])$ ; було зменшено граничне значення до 0.1. Це означає, що алгоритм агресивніше розбиватиме зображення на блоки. У аргументі [2 16] у функції `qtdecomp`, було зменшено розмір блоків, щоб вони були меншими. Це призведе до більшого розбиття зображення. Інші частини коду залишилися без змін. У результаті ці зміни роблять розбиття зображення грубішим і сильно стискають дані, що призводить до погіршення якості зображення.

Вхідні дані: зображення формату `jpg` з роздільною здатністю 555x370 пікселів (програмно змінене на 256x256) та розміром 113 КБ.

Отримані результати наведено у таблиці 1.

Аналіз результату. Коефіцієнт стиснення зріс удесятеро, що говорить про значне погіршення якості зображення. Також спостерігається істотна зміна таких показників, як `Time taken for compression`, `Time taken for decompression` і `PSNR`. Значення показників `Time taken for compression` і `Time taken for decompression` вказують на непомітну оптимізацію коду. Адже швидкість виконання операцій та отримання достовірного результату є основою будь-якого хорошого програмного продукту. Змінений код вирішує питання продуктивності всього програмного продукту. Але в той же час варто врахувати: саме в цьому випадку з помітним підвищенням продуктивності так само помітно падає і якість одержуваного результату. Показник `PSNR` (який є показником у свою чергу якості відновленого зображення порівняно з оригіналом) дорівнює 19.539598, що свідчить (залежно від переваги користувача або кінцевого споживача) про відносно погану якість отриманого зображення.

2. Покращення якості зображення. Для отримання зображення кращої якості було внесено наступні зміни. У рядку  $s = \text{qtdecomp}(I, 0.6, [4128])$ ; було збільшено граничне значення до 0.6. Це означає, що алгоритм більш консервативно розбиватиме зображення на блоки. У аргументі [4 128] у функції `qtdecomp` було збільшено розмір блоків. Блоки будуть більшими і міститимуть більше деталей. Інші частини коду залишилися без змін.

Вхідні дані: зображення формату `jpg` з роздільною здатністю 555x370 пікселів (програмно змінене на 256x256) та розміром 113 КБ.

Отримані результати наведено у таблиці 1.

Аналіз результату. Результат надає абсолютно протилежні дані, порівняно з попереднім результатом. З візуально помітним покращенням якості зобра-

ження значно підвищилося споживання ресурсів обладнанням, і водночас час обробки.

На рисунку 2 наведено порівняння результатів експериментів для початкового експерименту, при грубому розбитті (при погіршенні якості) та при консервативному розбитті (при покращенні якості).

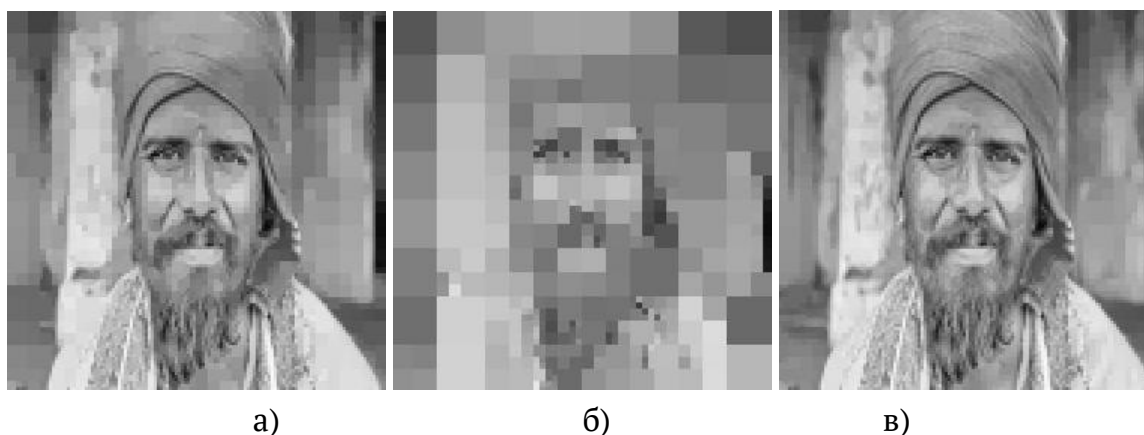


Рисунок 2 – Порівняння результатів експериментів: а) результат початковий; б) результат погіршення якості; в) результат покращення якості

У таблиці 1 наведено порівняння результатів експериментів. Наведено чотири показники алгоритму стиснення зображення для початкового експерименту, при грубому розбитті (при погіршенні якості) та при консервативному розбитті (при покращенні якості).

Таблиця 1

Порівняння результатів з покращення/  
погіршення якості стиснення зображення

Показники алгоритму стиснення зображення	Експерименти		
	Початковий варіант	Погіршення якості (грубе розбиття)	Покращення якості (консервативне розбиття)
	Значення показника		
Time taken for compression	0.291021	0.166994	1.195880
Compression ratio	3.329912	46.516547	1.987957
Time taken for decompression	0.457792	0.161126	0.774696
PSNR	26.724437	19.539598	28.145178

**Висновки.** Фрактальний метод стиснення зображень з використанням афінних перетворень дозволяє досягти високого ступеня стиснення, зберігаючи структурні особливості зображення. Однак цей метод вимагає немало обчислювальних ресурсів та складних алгоритмів для пошуку оптимальних перетворень та кодування даних.

В ході дослідження була детально проаналізована базова версія створеного програмного коду для внесення необхідних до завдання змін, щоб зберігати базове призначення програмного продукту. Крім того, були проаналізовані результати налагодження, які свідчать про підвищення та зниження продуктивності коду. Порівнюючи вихідні дані, також можна зробити висновки, що внесені зміни добре демонструють гнучкість і розширюють програмні можливості продукту.

#### ЛІТЕРАТУРА

1. Mandelbrot, B.B. (1982) *The Fractal Geometry of Nature*. Freeman Press, New York.
2. Barnsley M. *Fractals Everywhere* / M. Barnsley. – London: Academic Press Inc., 1988. – 370 p.
3. Zhang Aihua, 2014. Fractal image coding combined with discrete cosine transform complement. *Computer Technology and Development*, Vol. 24, No. 1, pp. 61–68.
4. Ватолін, Д. *Методи стиснення даних* / Д. Ватолін, А. Ратушняк, М. Смирнов, В. Юкин. – М.: Діалог-МИФИ, 2002. – 381 с.
5. Карпов, П. М. Швидкий фрактальний алгоритм стиснення зображень [Текст] / П. М. Карпов. – Наукова сесія МИФИ, 2006. – Т. 15.
6. Шашин К.В., Журба А.О. Дослідження фрактального методу кодування зображень // Молодь: наука та інновації: матеріали XI Міжнародної науково-технічної конференції студентів, аспірантів та молодих вчених, Дніпро, 22–24 листопада 2023 року: у 2-х т. / Національний технічний університет «Дніпровська політехніка» – Дніпро : НТУ «ДП», 2023. Том 2. С. 12-13.

#### REFERENCES

1. Mandelbrot, B.B. (1982) *The Fractal Geometry of Nature*. Freeman Press, New York.
2. Barnsley M. *Fractals Everywhere* / M. Barnsley. – London: Academic Press Inc., 1988. – 370 p.
3. Zhang Aihua, 2014. Fractal image coding combined with discrete cosine transform complement. *Computer Technology and Development*, Vol. 24, No. 1, pp. 61–68.



4. Vatolin, D. Data compression methods / D. Vatolin, A. Ratushnyak, M. Smirnov, V. Yukin. - M.: Dialog-MYFI, 2002. - 381 p.
5. Karpov, P. M. Fast fractal image compression algorithm [Text] / P. M. Karpov. – Scientific session of MYFI, 2006. – Vol. 15.
6. Shashin K.V., Zhurba A.O. Study of the fractal image coding method // Youth: science and innovation: materials of the 11th International scientific and technical conference of students, postgraduates and young scientists, Dnipro, November 22-24, 2023: in 2 volumes / National Technical University "Dnipro Polytechnic" – Dnipro: NTU "DP", 2023. Volume 2. P. 12-13.

Received 25.04.2024.  
Accepted 26.04.2024.

### ***Study of fractal image compression method with the purpose of improving compression quality***

*The development of the Internet, along with the availability of increasingly powerful computers and other digital devices, cameras, scanners and printers, has led to the widespread use of digital images. In this regard, interest in improving data compression algorithms, such as images, is growing.*

*Data compression is important for both transfer speed and storage efficiency. In addition to many commercial uses, compression technologies are also of interest in the military industry, such as applications for processing telemetry data from missile interceptors or for archiving terrain image data for defense simulations.*

*Solving the problem of image compression, or, more generally, image coding, has used advances and stimulated the development of many fields of engineering and mathematics.*

*The article examines fractal image compression — a data compression method based on the use of self-similar patterns in an image. This method allows you to achieve a high degree of compression while preserving image details.*

*Fractal image compression is a unique and efficient approach to data compression based on the mathematical theory of fractals. Nowadays, it has important applications and advantages that make it a valuable tool in image processing. The main advantages include:*

- 1. Preservation of details during compression. One of the key advantages of fractal compression is its ability to preserve a high degree of image detail in a relatively small amount of storage. This is especially important in situations where image quality must be preserved with limited storage and data transfer resources.*

2. *Efficiency of transmission through the network. Fractal compression allows for compact images, making it suitable for image transmission over a low-bandwidth network. This is especially true for mobile devices, the Internet of Things, and other scenarios where high bandwidth is not always available.*

3. *Adaptive compression for different resolutions. Fractal compression allows you to adapt the level of compression depending on the resolution and details of the image. This means that it can be used to compress various image sizes without significant loss of quality.*

4. *Data archiving and storage. Fractal compression can be useful for archiving and long-term storage of images, as it allows you to effectively reduce the amount of data without losing important information. This is especially important for libraries, archives, research databases and other data repositories.*

*Fractal image compression remains a relevant and valuable tool in today's environment, thanks to its ability to efficiently compress, preserve details, and adapt to different usage scenarios. Therefore, the study of its efficiency, the optimization of the software code to obtain a faster and better compression result, is an urgent task.*

**Журба Анна Олексіївна** - доцент кафедри Інформаційних технологій і систем ІІБТ УДУНТ.

**Zhurba Anna** - assistant professor, Department of information technology and systems, Institute of industrial and business technologies, Ukrainian state university of science and technologies.