

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-56-15>

УДК 004.62

Веремеско Андрій Андрійович, магістрант

<https://orcid.org/0009-0000-1916-3254>

Національний університет «Львівська політехніка», м. Львів, Україна

## ДОСЛІДЖЕННЯ ВИКОРИСТАННЯ ФРАКТАЛЬНИХ АЛГОРИТМІВ КОМПРЕСІЇ В ІНФОРМАЦІЙНИХ СИСТЕМАХ, ЩО ПРАЦЮЮТЬ З ВЕЛИКИМИ ОБСЯГАМИ ДАНИХ ТА ВИМАГАЮТЬ ВИСОКИХ КОЕФІЦІЄНТІВ СТИСНЕННЯ ПРИ МАКСИМАЛЬНОМУ ЗБЕРЕЖЕННІ ЯКОСТІ ЗОБРАЖЕНЬ

Веремеско А.А. Дослідження використання фрактальних алгоритмів компресії в інформаційних системах, що працюють з великими обсягами даних та вимагають високих коефіцієнтів стиснення при максимальному збереженні якості зображень. В роботі досліджено ефективність застосування фрактальних алгоритмів компресії в інформаційних системах, що обробляють великі обсяги даних та вимагають високих коефіцієнтів стиснення при збереженні якості зображень. Описано принципи роботи фрактальних алгоритмів, які використовують математичні моделі для виявлення самоподібних структур у даних, забезпечуючи високе співвідношення стискання без значної втрати якості. Проведено порівняння фрактальних методів з іншими популярними методами стиснення, такими як JPEG та Wavelet Transform, оцінено їх ефективність на основі продуктивності, коефіцієнтів стиснення та якості відновлених зображень. Виявлено, що фрактальні алгоритми є перспективними для систем, що потребують збереження високоякісних зображень при мінімальній втраті якості, особливо в умовах обмежених ресурсів зберігання та пропускну здатності мереж. Розроблено рекомендації щодо впровадження фрактальних алгоритмів у інформаційні системи для підвищення ефективності управління великими обсягами даних.

**Ключові слова:** фрактальний алгоритм, компресія зображень, інформаційна система, афінні перетворення, квадратичне розбиття.

Veremeenko A. Study of the usage of fractal compression algorithms for information systems that handle large volumes of data and require high compression ratios while preserving image quality. This work investigates the effectiveness of applying fractal compression algorithms in information systems that process large volumes of data and require high compression ratios while maintaining image quality. The principles of fractal algorithms, which utilize mathematical models to detect self-similar structures in data, are described, providing high compression ratios without significant loss of quality. A comparison of fractal methods with other popular compression methods, such as JPEG and Wavelet Transform, is conducted, evaluating their effectiveness based on performance, compression ratios, and the quality of restored images. It is found that fractal algorithms are promising for systems that need to preserve high-quality images with minimal quality loss, especially under conditions of limited storage resources and network bandwidth. Recommendations are developed for implementing fractal algorithms in information systems to improve the efficiency of managing large volumes of data.

**Keywords:** fractal algorithm, image compression, information system, affine transformations, quadratic partitioning.

**Постановка наукової проблеми.** Сучасний світ стикається з експоненціальним зростанням обсягів даних, що створюються та зберігаються в різних галузях, таких як телекомунікації, наука, медицина, мультимедіа та індустрія розваг. Ефективне стиснення даних стає критично важливим для зниження вимог до зберігання та покращення швидкості передачі даних. Стиснення даних дозволяє значно зменшити обсяг, що займає інформація на дисках та інших носіях. Це знижує витрати на обладнання для зберігання даних і покращує ефективність використання існуючих ресурсів. Також зменшується навантаження на мережеві канали при передачі великих обсягів даних. Зменшення обсягу даних за допомогою стиснення дозволяє прискорити процес передачі інформації через мережі, особливо в умовах обмеженої пропускну здатності.

Будь-яке зображення може бути збережене як сукупність пікселів. Зображення, збережене в комп'ютері, може мати певну кількість надлишковості. Якщо зображення має високу надлишковість – воно вимагає не тільки більше місця для зберігання, але й високошвидкісних каналів для передачі цього зображення з однієї мережі до іншої. Для мінімізації часу передачі та вартості зберігання необхідне стиснення.

Фрактальні алгоритми стиснення даних можуть забезпечити високу якість відтворення зі значним зменшенням розміру файлі. Фрактальні алгоритми компресії даних базуються на математичних принципах, що дозволяють ефективно виявляти самоподібні структури у даних. Це забезпечує високе співвідношення стискання без значних втрат якості. Фрактальні алгоритми також мають здатність до адаптивного стискання, що дозволяє більш ефективно стискати дані з різними характеристиками.

Фрактальні алгоритми, з їх здатністю до високоякісного стискання, можуть бути інтегровані в системи, що потребують збереження високої роздільної здатності зображень. В ході розробки однієї

з таких систем, а саме – універсальної web-платформи для астрономів аматорів, виникла потреба в збереженні високоякісних світлин космосу, при як найменшій втраті якості. Оскільки така платформа передбачає роботу з великим об'ємом зображень, завантажуваних користувачами, вартість зберігання цих даних може бути суттєвою. Також, це може вплинути на швидкодію платформи.

Таким чином, актуальність теми обумовлена необхідністю ефективного управління великими обсягами даних, зниженням витрат на зберігання та передачу, покращенням якості обслуговування користувачів даної платформи. В результаті аналізу та порівняння існуючих алгоритмів компресії – фрактальний алгоритм був обраний як найбільш оптимальний та відповідний поставленим задачам і цілям.

*Предметом дослідження* є комплекс методів та алгоритмів фрактального стиснення даних, зокрема зображень, з метою досягнення високого рівня стиснення без значної втрати якості для збереження в розробленій інформаційній системі. Особлива увага приділяється порівнянню фрактальних методів з іншими популярними методами стиснення, такими як JPEG та Wavelet Transform, для оцінки їх ефективності та застосування у різних галузях.

*Об'єктом дослідження* є процеси стиснення та декомпресії даних за допомогою фрактальних алгоритмів, які використано у розробленій інформаційній системі. Дослідження охоплює аналіз ефективності фрактальних методів у зменшенні обсягів даних та впливу на якість відновлених зображень, а також порівняння цих методів з іншими технологіями стиснення.

*Завданням даного дослідження* є аналіз існуючих методів фрактального стиснення зображень та їх основних принципів; порівняння ефективності фрактальних алгоритмів з іншими популярними методами стиснення, такими як JPEG та Wavelet Transform, на основі показників продуктивності, коефіцієнтів стиснення та якості відновлених зображень; аргументація доцільності використання фрактальних алгоритмів компресії для систем, що вимагають збереження великих обсягів зображень з мінімальною втратою якості, формування вимог до інформаційної системи, яка ґрунтується на методах фрактального стиснення зображень.

**Аналіз досліджень.** Поняття фракталу було описане математиком ІВМ Бенуа Мандельбротом. Саме він виявив, що традиційна геометрія не може повною мірою використовуватися для опису структури природних об'єктів, які є складними та хаотичними, наприклад, гори, хмари, берегові лінії, дерева, тощо.

Неевклідова геометрія або фрактальна геометрія займається нерегулярними та фрагментними візерунками. Фрактали є складними об'єктами, які мають властивість самоподібності, тобто частина фрактального об'єкта подібна до всього об'єкта, отже, фрактали є повторенням тієї ж самої структурної форми.

Для створення будь-якого фрактала потрібні три речі: набір перетворень (IFS - Iterated Function System), база, з якої починається ітерація, та множина конденсації (можливо, порожня множина).

У фрактальному стисненні зображень застосовуються де-які математичні функції та афінні перетворення, які також називаються контрактними афінними перетвореннями. Афінне перетворення, застосоване до фрактального стиснення зображень, є комбінацією перетворень, тобто: обертання, зміна масштабу, нахил або переміщення. Контрактне афінне перетворення грає значну роль, оскільки відстань між двома точками в оригінальному зображенні стає все ближче і ближче до фіксованої точки, і ця фіксована точка після афінного перетворення є не що інше, як зображення. Зображення, отримане після застосування контрактного афінного перетворення, є наближенням до оригінального зображення.

В основі ітераційної системи функцій лежить обрання точки або фігури та її заміна на декілька інших, ідентичних. Ітераційна система функцій – це колекція контрактних афінних перетворень  $s_1, s_2$  і так далі до  $s_n$ .

Застосування ітераційної системи функцій (IFS) відбувається до тих пір, доки вона не збігається до певної точки, яка є атрактором IFS. М. Барнслі вивів спеціальну форму контрактного відображення перетворення (CMT - Contractive Mapping Transform), що застосовується до IFS та називається теоремою Коледжу. Вона дістала таку назву через те, що М. Барнслі вперше представив свою ідею під час лекцій у коледжі. Ця теорема стверджує, що відстань Гаусдорфа, що є математичною мірою для визначення максимальної відстані між точками двох множин у метричному просторі, зокрема, між двома зображеннями має бути якомога меншою, тобто атрактор IFS має бути близьким наближенням до оригінального зображення. Відстань між оригінальним зображенням та атрактором називається помилкою коледжу, і повинна бути якомога меншою.

На основі напрацювань М. Барнслі, інший відомий дослідник в галузі фракталів, М.Ф. Жакен, запропонував деталізований підхід. У методі Жакена зображення розділяється на підзображення, які називаються "блоками діапазону", і PIFS (Partitioned Iterated Function System, або ж розподілена ітераційна система функцій) застосовується до підзображень, а не до всього зображення, щоб знайти найбільш підходящий блок домену.

У розділеній IFS замість застосування афінного перетворення до всього зображення, контрактні афінні перетворення застосовуються до його частин, і об'єднання афінних перетворень створює кінцеве зображення.

Алгоритм фрактального стиснення зображень зображений на рис. 1.

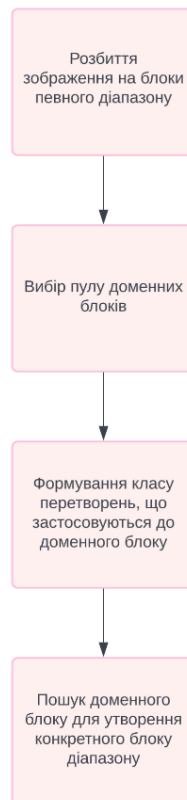


Рис. 1. Алгоритм фрактального стиснення зображень

Основною метою є збільшення коефіцієнта стиснення та мінімізація часу стиснення. Іншим важливим аспектом є розмір стисненого файлу у порівнянні з якістю декомпресованого зображення. Існує два типи стиснення: стиснення без втрат та стиснення з втратами. Стиснення з втратами завжди створює зображення, які не є точною копією оригінальних даних, воно усуває надлишковість і створює наближення до оригіналу. Фрактальний алгоритм компресії зображень є методом стиснення з втратами.

**Мета роботи.** Метою даної статті є аналіз існуючих методів фрактального стиснення зображень, порівняння їх з іншими популярними методами стиснення, такими як JPEG і Wavelet Transform, та аргументація доцільності використання фрактального алгоритму компресії для розроблення інформаційної системи, що сприяє збереженню великих обсягів зображень, з мінімальною втратою якості. Також розглядається проблематика пошуку самоподібних блоків у зображеннях, вплив різних схем розбиття зображення на ефективність стиснення та основні показники якості відновлених зображень.

### 1. Порівняння алгоритмів та результати

Існують різні методи компресії зображень, такі як Joint Photographic Experts Group (JPEG), JPEG2000, Wavelet Transform та фрактальне стиснення зображень. Ці методи були протестовані для реальних зображень розміром  $512 \times 512$  та  $1024 \times 1024$ .

Результати продуктивності наведені в таблицях 1, 2 та 3.

Табл. 1. Продуктивність алгоритмів компресії для зображення 512×512

Алгоритм	Значення PSNR(дБ)	Компресія	Декомпресія
Wavelet	37.65	9.89 сек	6.7 сек
JPEG	33.72	0.49 сек	0.29 сек
JPEG2000	38.27	15.20 сек	10.27 сек
Фрактальний	31.03	1201.45 сек	4.92 сек

Табл. 2. Продуктивність алгоритмів компресії для зображення відбитків пальців 1024×1024

Алгоритм	Значення PSNR(дБ)	Компресія	Декомпресія
Wavelet	39.0	24.5 сек	14.9 сек
JPEG	35.2	1.03 сек	0.57 сек
JPEG2000	40.0	31.2 сек	20.1 сек
Фрактальний	30.2	3019.21 сек	9.7 сек

Табл. 3. Продуктивність на основі коефіцієнта стиснення різних алгоритмів кодування

Алгоритм	Коефіцієнт стиснення
Wavelet	13
JPEG	4
JPEG2000	16
Фрактальний	8

Можна трактувати результати наведені в таблиці 3, наступним чином: фрактальне стиснення забезпечує коефіцієнти стиснення не менше 8:1. Хоча цей метод не досягає таких високих коефіцієнтів, як Wavelet або JPEG2000 і тривалість компресії є значно більшою, проте, він має перевагу у збереженні якості зображення при високих ступенях стиснення та забезпечує відносно швидку декомпресію, що є основною вимогою в розробці платформи.

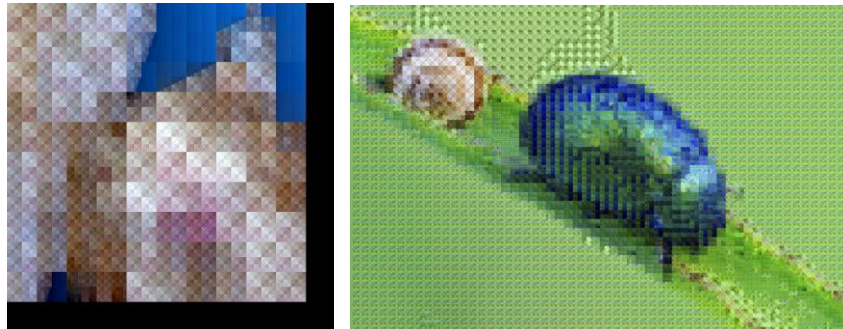
Результати стиснення за допомогою фрактального алгоритму зображені на рис. 2-5 для прикладів світлини «Кіт» та «Жук» та демонструють високе збереження якості навіть при великому ступені компресії (розмір рангового блоку до 64 пікселів, що забезпечує компресію в більш ніж 1000 разів). Проте, висока якість досягається найкраще для зображень, що містять елементи самоподібності, що добре видно в різниці між зображенням «Кіт» та «Жук».



Рис. 2. Оригінальні зображення



Рис. 3. Декомпресійовані зображення при розмірі рангового блоку  $r=2$

Рис. 4. Декомпресійовані зображення при розмірі рангового блоку  $r=8$ Рис. 5. Декомпресійовані зображення при розмірі рангового блоку  $r=64$ 

### 1.1 Переваги фрактального стиснення

1. Незалежність від роздільної здатності та пристроїв.
2. Декомпресія в 6 разів швидша, ніж у JPEG2000.
3. Вища якість при високих коефіцієнтах стиснення.
4. Коефіцієнти стиснення можуть бути покращені шляхом збільшення часу стиснення без збільшення часу декомпресії або зниження якості зображення.

### 1.2 Недоліки фрактального стиснення

1. Час стиснення в значно довший, ніж у інших методів.
2. Непідтверджена технологія.
3. Висока вартість для звичайних користувачів.

### 1.3 Переваги JPEG

1. JPEG підходить тоді, коли потрібні низькі коефіцієнти стиснення.
2. Оскільки це промисловий стандарт, він є безкоштовним.

### 1.4 Недоліки JPEG

1. JPEG не працює з бінарними зображеннями.
2. JPEG не працює добре з нереалістичними зображеннями.
3. Якість зображень значно погіршується при високих коефіцієнтах стиснення.

## 2. Перетворення

### 2.1 Афінне перетворення

Алгоритм фрактальної компресії зображень базується на ідеї самоподібності, тобто частини зображення повторюються в різних масштабах. Це дозволяє представити зображення як набір афінних перетворень, що значно зменшує обсяг даних, необхідних для його збереження.

Афінні перетворення – це лінійні перетворення, які є композицією обертання, трансляцій, масштабувань та зсувів. Воно не зберігає кути або довжини. Два або більше послідовних перетворень можуть бути застосовані до зображення за допомогою афінного перетворення.

Перетворення  $\omega: R^2 \rightarrow R^2$  має вигляд:

$$w(x, y) = (ax + by + e, cx + dy + f),$$

де  $a, b, c, d, e$  і  $f$  – дійсні числа, це називається (двовимірним) афінним перетворенням. Афінні перетворення є ключовим поняттям для фрактальних алгоритмів.

### 2.2 Теорема про контрактні відображення Банаха

Перетворення  $f: X \rightarrow X$  в метричному просторі  $(X, d)$  називається контрактним або контрактним відображенням, якщо існує константа  $0 \leq s < 1$  така, що:

$$d(f(x), f(y)) \leq s \cdot d(x, y), \quad \forall x, y \in X$$

Будь-яке таке число  $s$  називається коефіцієнтом контрактності для  $f$ .

### 2.3 Простір Гаусдорфа

Нехай  $(X, d)$  є повним метричним простором. Тоді відстань Гаусдорфа між точками  $A$  і  $B$  у  $H(X)$  визначається як:

$$h(A, B) = \max\left\{\sup_{a \in A} \inf_{b \in B} d(a, b), \sup_{b \in B} \inf_{a \in A} d(a, b)\right\}$$

Це також називається метричним простором Гаусдорфа.

## 3. Ітераційні системи функцій (IFS)

IFS стали дуже популярними у середині 1980-х років. М. Барнслі та його колеги з Технологічного інституту в Джорджії першими помітили потенціал IFS для застосування в комп'ютерній графіці. М. Барнслі і А. Слоан представили в наукових журналах здатність IFS до стиснення кольорових зображень із відношенням понад 10 000 до 1.

Ітераційна система функцій з ймовірностями складається з IFS разом з упорядкованим набором чисел  $\{p_1, p_2, \dots, p_N\}$ , таких що  $p_1 + p_2 + p_3 + \dots + p_N = 1$  і  $p_i > 0$  для  $(i = 1, 2, \dots, N)$ .

Ймовірності пов'язані з теорією міри атракторів IFS і відіграють роль у обчисленні зображень атрактора IFS за допомогою алгоритму випадкової ітерації.

### 3.1 Недоліки IFS

Основний недолік IFS полягає в складності процесу отримання набору афінних перетворень для представлення даного зображення. Для повністю автоматизованого фрактального стиснення зображень необхідний алгоритм, який не вимагає "користувацького втручання", такого як налаштування коефіцієнтів в IFS.

## 4. Методи під IFS

### 4.1 Детерміністичний метод

У цьому методі, де-які афінні перетворення застосовуються до кожної підмножини зображення і таким чином, шукається наступне зображення, яке повинно бути повною підмножиною простору  $(R^2)$ , де знаходиться зображення. Після багаторазового застосування афінних перетворень ми отримуємо послідовність зображень, які повинні зближуватися до деякої точки, яка буде граничною точкою. Ця гранична точка – і є зображення. Для використання цього підходу в генерації фракталів потрібна велика кількість пам'яті, оскільки на кожній ітерації ми генеруємо деяке зображення, яке повинно бути тимчасово збереженим.

Нехай  $\{X; w_1, w_2, \dots, w_N\}$  є гіперболічним IFS. Виберіть компактну множину  $A_0 \subseteq R^2$ . Потім обчислюйте послідовно  $A_n = W^n(A)$  відповідно до:

$$A_{n+1} = \bigcup_{j=1}^N w_j(A_n) \quad \text{для } n = 1, 2, \dots$$

Таким чином, створюється послідовність  $\{A_n; n = 0, 1, 2, 3, \dots\} \subseteq H(X)$ . Згідно з теоремою IFS, послідовність  $\{A_n\}$  збігається до атрактора IFS у метриці Гаусдорфа. Фрактали можуть бути згенеровані за допомогою ітерації Ішікави. Метою ітерації Ішікави, в свою чергу, є визначенням різних типів орбіт для генерації фракталів.

### 4.2 Алгоритм випадкової ітерації

Підхід випадкової ітерації відрізняється від детерміністичного підходу тим, що початкова множина є одиничною точкою, і на кожному рівні ітерації використовується лише одне з визначальних афінних перетворень для обчислення наступного рівня. На кожному рівні афінне перетворення, що застосовується, обирається випадковим чином. Усі точки, крім ранніх, наносяться на графік і відкидаються після використання для обчислення наступного значення. Випадковий алгоритм уникає потреби у великій пам'яті комп'ютера, та найкраще підходить для простих комп'ютерів, на яких можна обчислювати та відображати одну точку за раз. З іншого боку, для створення результуючого зображення цим способом потрібно тисячі точок, щоб воно не було недостатнім.

Нехай  $\{X; w_1, w_2, \dots, w_N; p_1, p_2, \dots, p_N\}$  є IFS з ймовірностями. Виберіть  $(x_0 \in X)$  і далі вибирайте рекурсивно, незалежно,

$$x_n \in \{w_1(x_{n-1}), w_2(x_{n-1}), \dots, w_N(x_{n-1})\} \quad \text{для } n = 1, 2, 3, \dots,$$

де ймовірність того, що  $x_n = w_i(x_{n-1})$  дорівнює  $p_i$ . Таким чином, створюється послідовність  $\{x_n: n = 0, 1, 2, 3, \dots\} \subseteq X$ .

## 5. Фрактальна компресія зображень

Фрактальна компресія зображень є методом стиснення з втратами, що використовує фрактальну теорію. Основна ідея фрактального стиснення полягає в тому, щоб використовувати властивість самоподібності зображень для їх стиснення. Збереження зображення як набору пікселів вимагає більшої кількості пам'яті. Зберігаючи лише коефіцієнти афінних перетворень, потреба в пам'яті може бути зменшена. Стиснення є більш обчислювально витратним, оскільки інтенсивний пошук використовується для знаходження самоподібних блоків. При високому коефіцієнті компресії, фрактальне стиснення може забезпечити вищу якість.

Стандартні методи стиснення зображень можуть оцінюватися за їх коефіцієнтом стиснення: відношенням пам'яті, необхідної для збереження оригінального зображення, до пам'яті, необхідної для збереження зображення в скомпресійованій формі. Після застосування алгоритму фрактальної компресії, зображення стають незалежними від роздільної здатності. Це означає, що при генерації збільшених зображень стає видимою більша кількість деталей порівняно зі звичайним збільшенням, яке поступово розмивається. Фрактальне стиснення зображень є дуже ефективним для компресії зображень з високим ступенем самоподібності. Проте, процес пошуку відповідних блоків та застосування афінних перетворень може бути дуже обчислювально важким і займати значний час.

Аналіз цих методів дозволив зробити висновок, що для стиснення зображень в інформаційній системі найбільш доцільно використовувати фрактальну компресію зображень з декількох причин. Детерміністичний метод не забезпечує такої високої якості компресії, як фрактальна компресія. Він підходить більше для математичних та теоретичних задач, ніж для практичного використання в інформаційних системах для стиснення зображень. Хоча метод випадкової ітерації може бути корисним у генерації фракталів та деяких графічних застосуваннях, він менш ефективний для стиснення зображень через високу обчислювальну складність і менш передбачувані результати.

Фрактальна компресія здатна забезпечити високий ступінь стиснення без значної втрати якості. Це особливо важливо для великих зображень або для інформаційної системи, яка зберігатиме велику кількість графічних даних. Цей метод дозволяє зберегти дрібні деталі та текстури, що робить його ідеальним для фотографій та складних зображень. Фрактальна компресія добре підходить для зображень, які часто піддаються масштабуванню, оскільки дозволяє зберігати високий рівень деталізації навіть при збільшенні зображення. Отже, для розроблення інформаційної системи, яка займається стисненням зображень, фрактальна компресія є найкращим вибором, забезпечуючи баланс між високим ступенем стиснення та збереженням якості зображень.

### 5.1 Алгоритм фрактальної компресії зображення

Для компресії зображення за допомогою фрактального алгоритму компресії, слід виконати наступні кроки:

1. Розділити зображення на блоки діапазону.

Розділити вихідне зображення на неперекривні блоки однакового розміру, які називаються блоками діапазону (range blocks).

2. Розділити зображення на доменні блоки.

Розділити зображення на перекривні блоки, які є більшими за блоки діапазону, і називаються доменними блоками (domain blocks). Розмір доменних блоків зазвичай вдвічі більший за розмір блоків діапазону.

3. Застосування афінних перетворень.

Для кожного доменного блоку виконується наступне:

- Масштабування доменного блоку до розміру блоку діапазону.
- Обчислення середнього значення яскравості та контрасту.

4. Пошук найкращого співпадіння.

Для кожного блоку діапазону слід знайти найкраще співпадіння серед масштабованих доменних блоків з використанням афінних перетворень. Це здійснюється за допомогою мінімізації різниці між блоками діапазону та доменними блоками.

5. Збереження параметрів перетворення.

Збереження параметрів найкращого афінного перетворення (масштаб  $s$  і зсув  $o$ ) для кожного блоку діапазону.

Для декомпресії зображення необхідно виконати ітераційний процес, що полягає в наступному:

1. Ініціалізація буферу зображення випадковими значеннями.
2. Для кожного блоку діапазону застосовуються збережені афінні перетворення до відповідного доменного блоку, оновлюючи значення пікселів у буфері зображення.
3. Процес повторюється, доки буфер зображення не збіжиться до стабільного стану.

### **5.2 Приклади використання фрактальної компресії зображень**

Алгоритми з даного сімейства не є популярними в побуті, через свої унікальні особливості та специфічні вимоги до використання. Проте, вони успішно застосовуються в сферах, що працюють з великими обсягами зображень та вимагають максимальної збереження якості та можливості масштабування.

1. Медична візуалізація - у медичній візуалізації важливо зберігати високоякісні зображення для діагностики, але обсяг даних може бути дуже великим. Фрактальна компресія дозволяє зберігати медичні зображення (наприклад, МРТ, КТ) з високою якістю, забезпечуючи при цьому значне зменшення розміру файлів.
2. Астрономічні зображення - як було зазначено в статті, фрактальна компресія використовується для збереження високоякісних астрономічних зображень на універсальній web-платформі для астрономів-аматорів. Вона дозволяє зберігати великий обсяг зображень космічних тіл, зменшуючи витрати на зберігання і забезпечуючи швидку передачу даних.
3. Збереження супутникових зображень - супутникові зображення мають високу роздільну здатність і часто використовуються для моніторингу земної поверхні, прогнозування погоди та спостереження за екологічними змінами. Фрактальна компресія допомагає зберігати ці зображення з високою деталізацією, зменшуючи обсяг даних, що передаються та зберігаються.
4. Застосування у наукових дослідженнях - у наукових дослідженнях часто використовуються високоякісні зображення, наприклад, у біології (зображення клітин, тканин) або фізиці (зображення результатів експериментів). Фрактальна компресія дозволяє зберігати такі зображення з мінімальними втратами якості, що є критичним для точності наукових висновків.
5. Географічні інформаційні системи (ГІС) - ГІС використовують великі обсяги географічних даних, включаючи карти, аерофотознімки та супутникові зображення. Фрактальна компресія допомагає зменшити обсяг даних, необхідних для зберігання та обробки в ГІС, забезпечуючи при цьому високу якість зображень для точного аналізу та прийняття рішень.

### **6. Схеми розбиття**

При розробці схеми фрактального кодування необхідно приймати конкретне рішення щодо типу розбиття зображення, яке використовується для формування блоку діапазону. Існує широкий вибір схем розбиття.

#### **6.1 Розбиття на квадрати фіксованого розміру**

Це найпростіший метод розбиття зображення. Такий тип розбиття успішний у перетворювальному кодуванні окремих блоків зображення, наприклад, у реалізації стандарту JPEG.

#### **6.2 Квадродерево**

Цей тип розбиття базується на рекурсивному поділі вибраних квадрантів зображення. Після розбиття зображення буде виглядати як деревоподібна структура, в якій кожен нетермінальний вузол має чотири нащадки.

#### **6.3 Горизонтально-вертикальне розбиття**

У цій схемі розбиття зображення поділяється на дві частини горизонтальною або вертикальною лінією, щоб утворити два нові прямокутники. Розбиття повторюється рекурсивно, доки не буде задоволено критерій покриття, як у схемі квадродерева.

#### **6.4 Трикутне розбиття**

Інший спосіб розбиття зображення базується на трикутниках. Прямокутне зображення ділиться по діагоналі на два трикутники. Кожен з цих трикутників поділяється на 4 трикутники шляхом сегментації трикутника вздовж лінії, яка з'єднує три точки поділу вздовж сторін трикутника, що є краще та гнучкіше, порівняно зі схемою горизонтально-вертикального розбиття.

Для стиснення зображень в інформаційній системі для астрономів-аматорів, найбільш доцільним є використання квадродерева з декількох причин. Метод розбиття на квадрати фіксованого розміру простий у реалізації, але не враховує різноманітність вмісту зображення. Для



областей з високою деталізацією він може бути неефективним, оскільки використовує однаковий розмір блоків незалежно від складності. Метод горизонтально-вертикального розбиття менш гнучкий порівняно з квадродеревом. Він може створювати неприродні блоки, що можуть не відповідати вмісту зображення, що погіршує якість стиснення.

### **6.5 Вибір оптимального розбиття для інформаційної системи**

Хоча метод трикутного розбиття може бути ефективним для певних типів зображень (наприклад, з багатьма діагональними лініями), він складніший в реалізації та обчисленнях. Також він може призводити до створення надлишкових трикутників в однорідних областях, що знижує ефективність стиснення. Квадродерево дозволяє адаптивно розділяти зображення на блоки різних розмірів залежно від вмісту зображення. Це означає, що однорідні області можуть бути представлені великими блоками, тоді як складніші області — меншими. Це підвищує ефективність стиснення. Завдяки фрагментації зображення до необхідного рівня деталізації, Квадродерево добре зберігає важливі деталі зображення, що важливо для високої якості результату після декомпресії. Квадродерево також є досить ефективним з точки зору обчислювальної складності, оскільки дозволяє швидко ідентифікувати однорідні області та оптимізувати процес стиснення. Враховуючи вищезазначені фактори, метод квадродерево є найкращим вибором для стиснення зображень в інформаційній системі, забезпечуючи баланс між ефективністю стиснення, якістю зображення та обчислювальною складністю.

Алгоритм квадродерева на прикладі компресії зображень:

1. Ініціалізація:

- Починаємо з кореня дерева, який представляє все зображення.

2. Рекурсивний поділ:

- Для кожного блоку перевіряємо, чи є він однорідним.
- Якщо блок однорідний, залишаємо його як є.
- Якщо блок не однорідний, ділимо його на чотири квадранти і повторюємо процес

для кожного квадранту.

3. Завершення:

- Процес продовжується, поки не будуть досягнуті однорідні блоки або мінімальний розмір блоків.

### **6.6 Приклади застосування квадродерева**

Тип розбиття квадродерево найбільше підходить для наступних цілей

1. Зберігання картографічних даних - картографічні дані можуть бути збережені у вигляді квадродерев, де кожен вузол представляє певну область карти. Однорідні області (наприклад, великі водойми або пустелі) можуть бути представлені великими блоками, тоді як більш складні області (міста, дороги) розбиваються на дрібніші блоки.

2. Адаптивне зберігання зображень - аерофотознімки та супутникові зображення можна також зберігати у вигляді квадродерев, що дозволяє адаптивно розділяти зображення залежно від його вмісту. Це зменшує обсяг даних, що зберігаються, і забезпечує швидкий доступ до різних частин зображення.

3. Обробка даних - квадродерево дозволяє швидко знаходити та обробляти необхідні області зображення або карти. Наприклад, для визначення ділянок лісу або водних ресурсів можна швидко знаходити відповідні блоки в квадродереві.

## **7. Показники якості у фрактальній компресії**

Якість стиснутих зображень стає важливим фактором разом із значним коефіцієнтом стиснення, що визначає ефективне стиснення зображень. Стиснення зображення зменшує кількість надлишкових даних у ньому. Мета показників якості полягає в тому, щоб відобразити якість декодованих зображень. Об'єктивний показник якості повинен добре відображати спотворення зображення, наприклад, через розмитість, шум або стиснення.

### **7.1 Середньоквадратична помилка (MSE - Mean squared error)**

MSE або відношення сигналу до шуму (SNR - Signal-to-noise ratio) є показником якості, який часто використовується для вимірювання відхилення між оригінальним і закодованим зображенням.

### **7.2 Пікове відношення сигналу до шуму (PSNR - Peak signal-to-noise ratio)**

Більші значення SNR і PSNR вказують на меншу різницю між оригінальним і відновленим зображенням. Це найпоширеніший об'єктивний показник якості/спотворення зображень. Основною

перевагою цього показника є легкість обчислення, але він не відображає сприйняття якості. Важливою властивістю PSNR є те, що невеликий просторовий зсув зображення може спричинити значне числове спотворення без візуального спотворення і навпаки, невелике середнє спотворення може призвести до значного візуального артефакту.

### **7.3 Середня різниця (AD – Average difference)**

Нижче значення середньої різниці дає "чистіше" зображення, оскільки зменшується кількість шуму.

### **7.4 Максимальна різниця (MD – maximum difference)**

Велике значення MD означає, що якість зображення є низькою.

### **7.5 Нормалізована кореляція**

Ці показники вимірюють схожість між двома зображеннями. Всі кореляційні показники наближаються до 1, оскільки різниця між двома зображеннями наближається до нуля. Так як показники різниці та кореляції доповнюють один одного, мінімізація показників відстані означає максимізацію показників кореляції.

### **7.6 Середня абсолютна помилка (MAE – Mean absolute error)**

MAE оцінює середню різницю між відповідними пікселями оригінального та відновленого зображень, що дозволяє визначити рівень спотворення. Велике значення MAE означає, що якість зображення є низькою.

### **7.7 Нормалізована абсолютна помилка (NAE – Normalized absolute error)**

За цим показником оцінюється, наскільки відновлене зображення відрізняється від оригінального зображення, причому значення нуль є ідеальною відповідністю.

### **7.8 Вибір оптимального показника якості для інформаційної системи**

Для оцінювання якості зображення в інформаційній системі найдоцільніше використовувати пікове відношення сигналу до шуму з кількох причин.

Середньоквадратична помилка є основою для розрахунку, але сама по собі не так інтуїтивно зрозуміла і менш зручна для інтерпретації. Високе значення середньоквадратичної помилки вказує на значну помилку, але воно не враховує динамічний діапазон зображення.

Середня різниця є дуже простим методом, який враховує лише середню різницю між відповідними пікселями, що не відображає загальну якість зображення так точно, як інші методи.

Максимальна різниця фокусується на максимальній різниці між відповідними пікселями, що може бути корисним у виявленні суттєвих спотворень, але не дає повної картини про загальну якість зображення.

Метод нормалізованої кореляції більше підходить для оцінки подібності зображень, ніж для оцінки якості після стиснення.

Середня абсолютна помилка є більш інтуїтивно зрозумілим показником, ніж середньоквадратична помилка, але не враховує варіацій яскравості та контрастності, що може бути важливим для візуального сприйняття якості зображення.

Нормалізована абсолютна помилка нормалізує помилки відносно амплітуди зображення, але також не враховує всі аспекти людського сприйняття.

Метод пікового відношення сигналу до шуму добре враховує різні види шуму, які можуть виникнути в процесі стиснення та декомпресії зображення. Він враховує максимальні амплітуди сигналу та шуму, що робить його чутливим до суттєвих спотворень. Значення пікового відношення сигналу до шуму виражається в децибелах, що робить його зручним для інтерпретації. Високе значення пікового відношення сигналу до шуму вказує на високу якість зображення, що полегшує порівняння різних методів стиснення або налаштувань параметрів.

Хоча пікове відношення сигналу до шуму не є ідеальним з точки зору моделювання людського сприйняття, він все ж корелює з тим, як люди сприймають якість зображення. Це робить його практичним вибором у багатьох реальних додатках. Таким чином, пікове відношення сигналу до шуму є найбільш доцільним методом для оцінки якості зображення в інформаційній системі, оскільки він поєднує в собі інтуїтивну зрозумілість, зручність інтерпретації та достатню точність для оцінки різних видів спотворень.

Середньоквадратична помилка та пікове відношення сигналу до шуму є ключовими метриками для оцінки якості зображення після стиснення. Середньоквадратична помилка обчислює середнє значення квадрату різниць між відповідними пікселями двох зображень: оригінального (I) та зображення після стиснення (K). Формула для MSE виглядає наступним чином

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i, j) - K(i, j)]^2,$$

де:

$I(i, j)$  - значення пікселя в координатах  $(i, j)$  оригінального зображення,

$K(i, j)$  - значення пікселя в координатах  $(i, j)$  зображення після стиснення,

$m$  та  $n$  - кількість рядків та стовпців у зображенні відповідно.

Пікове відношення сигналу до шуму виражається в децибелах і обчислюється за формулою:

$$PSNR = 10 \cdot \log_{10} \frac{MAX^2}{MSE}$$

де:

$MAX$  - максимальне можливе значення пікселя в зображенні. Для 8-бітних зображень, це зазвичай 255.

$MSE$  - середньоквадратична помилка, обчислена за попередньою формулою.

Наведемо приклад обчислення. Припустимо, у нас є 2 маленькі зображення розміром 2x2 пікселі:

Оригінальне зображення:  $\begin{bmatrix} 100 & 150 \\ 200 & 250 \end{bmatrix}$

Зображення після стиснення:  $\begin{bmatrix} 98 & 152 \\ 203 & 247 \end{bmatrix}$

Обчислимо  $MSE$ :

$$MSE = \frac{1}{4} ((100 - 98)^2 + (150 - 152)^2 + (200 - 203)^2 + (250 - 247)^2)$$

$$MSE = \frac{1}{4} (4 + 4 + 9 + 9)$$

$$MSE = \frac{26}{4} = 6,5$$

Наведемо приклад обчислення  $PSNR$ . Припустимо, що максимальне можливе значення пікселя ( $MAX$ ) для наших зображень - 255. Використаємо обчислене значення  $MSE$  для обчислення  $PSNR$ :

$$PSNR = 10 \cdot \log_{10} \frac{255^2}{6,5}$$

$$PSNR = 10 \cdot \log_{10} \frac{65025}{6,5}$$

$$PSNR = 10 \cdot \log_{10}(10003.846), PSNR \approx 10 \cdot 4, PSNR \approx 40 \text{ dB}$$

Таким чином, для цього прикладу  $PSNR$  становить приблизно 40 дБ.

## 8. Процес фрактального стиснення зображення

Щоб застосувати фрактальне стиснення для зображення, інформаційна система реалізує такий алгоритм:

1. Зчитати файл з фрактальним кодом
2. Створити буфери пам'яті для доменного і діапазонного екранів
3. Ініціалізувати буфер діапазонного екрана довільним початковим станом
4. Повторити:
  - Для кожного домену:
  - Замінити цей домен відповідним перетвореним блоком діапазону
  - Наступний домен
  - Скопіювати вміст доменного екрана на діапазонний екран
  - Доки не буде досягнуто необхідної кількості ітерацій
5. Зберегти кінцевий файл

### 8.1 Проблема швидкості стиснення

Час, витрачений на стиснення будь-якого зображення, є основним недоліком будь-якого алгоритму фрактального стиснення. Основною важливою частиною алгоритму є зіставлення блоку діапазону з відповідним доменним блоком таким чином, щоб відстань Гаусдорфа між оригінальним зображенням і його атрактором була мінімальною. Алгоритм вимагає багато часу на пошук, коли точний збіг не знайдено. Для великого зображення час стиснення може вимірюватися в годинах, а

не в секундах або навіть хвилина. Однак можна стверджувати, що час декомпресії є важливішим, ніж час стиснення.

**Висновки.** Фрактальне стиснення зображень, забезпечує високу якість відновлених зображень і є незалежним від роздільної здатності та пристроїв. Його головною перевагою є швидка декомпресія, що робить його застосування перспективним для медичних та наукових сфер. В свою чергу, основний недолік методу – тривалий час стиснення, який обмежує практичне застосування в умовах, де важлива швидкість обробки.

Порівняння з методами JPEG, JPEG2000 та Wavelet Transform демонструє, що фрактальне стиснення перевершує інші методи при високих коефіцієнтах стиснення, але поступається JPEG при низьких коефіцієнтах через простоту та доступність останнього. Фрактальне стиснення має перспективи для систем, що вимагають високої якості зображень, і працюють з великим обсягом даних. Особлива ефективність фрактального алгоритму досягається у випадках, коли компресія зображення відбувається всього один раз, а доступ до зображення здійснюється численну кількість разів. Таким чином, фрактальне стиснення зображень є перспективною технологією, яка потребує подальших вдосконалень для зменшення часу обробки, що дозволить його ширше впровадження в різні галузі.

#### Список бібліографічного опису

1. Працьовитий М.В. Фрактальний підхід у дослідженнях сингулярних розподілів. Київ: НПУ імені М.П. Драгоманова, 1998. 296 с.
2. Працьовитий М.В., Свинчук О.В. Розсіювання значень однієї фрактальної неперервної немонотонної функції канторівського типу // Нелінійні коливання. 2018. Том 21, № 1. С. 116 – 130.
3. Лапчик В. В., Олексієнко С. О. Фрактальний аналіз та його застосування в інформаційних технологіях. Київ: Видавничий дім "Слово", 2007. 218 с.
4. Тарасенко Ф. П., Бойченко В. В. Математичні моделі фрактальної геометрії в інформаційних системах. Харків: ХНУРЕ, 2010. 272 с.
5. Зінченко А. В., Білозерцев С. І. Застосування фрактальної геометрії в обробці та стисненні зображень // Вісник КНУ. Серія: Фізико-математичні науки. 2015. № 3. С. 45-54.
6. Барнслі, М. Ф. Фрактали всюди // Академічна преса, 1988. 394 с.
7. Фішер, Й. Фрактальна компресія зображень: теорія та застосування. Спрінгер, 1995. 316 с.
8. Жакен, А. Е. Кодування зображень на основі фрактальної теорії ітераційних контрактних перетворень зображень // IEEE Transactions on Image Processing, 1992. 1(1), 18-30.
9. Слоан, К. Р. Фрактальна компресія зображень: огляд // Proceedings of the IEEE, 1994. 82(4), 647-655.
10. Девіс, Г. М. Вейвлет-аналіз фрактальної компресії зображень // IEEE Transactions on Image Processing, 1994. 3(2), 213-220.
11. Гартенштейн, Г. Оцінка продуктивності алгоритмів фрактальної компресії зображень // Proceedings of the Data Compression Conference, 1996. сс. 123-132. IEEE.
12. Врскай, Е. Ітераційні функціональні системи: теорія, застосування та автоматичне створення фракталів // Journal of Mathematical Imaging and Vision, 1994. 4, 75-100.

#### References

1. Pratsiovyty M.V. Fractal Approach in the Study of Singular Distributions. Kyiv: NPU named after M.P. Drahomanov, 1998. 296 p.
2. Pratsiovyty M.V., Svyinchuk O.V. Scattering of Values of a Fractal Continuous Non-monotonic Function of Cantor Type // Nonlinear Oscillations. 2018. Vol. 21, No. 1. pp. 116-130.
3. Lapchuk V. V., Oleksiienko S. O. Fractal Analysis and Its Application in Information Technologies. Kyiv: Publishing House "Slovo", 2007. 218 p.
4. Tarasenko F. P., Boichenko V. V. Mathematical Models of Fractal Geometry in Information Systems. Kharkiv: KhNURE, 2010. 272 p.
5. Zinchenko A. V., Bilozercev Y. I. Application of Fractal Geometry in Image Processing and Compression // Bulletin of KNU. Series: Physical and Mathematical Sciences. 2015. No. 3. pp. 45-54.
6. Barnsley, M. F. Fractals Everywhere // Academic Press, 1988. 394 p.
7. Fisher, Y. Fractal Image Compression: Theory and Application. Springer, 1995. 316 p.
8. Jacquin, A. E. Image Coding Based on a Fractal Theory of Iterated Contractive Image Transformations // IEEE Transactions on Image Processing, 1992. 1(1), 18-30.
9. Sloan, K. R. Fractal Image Compression: A Review // Proceedings of the IEEE, 1994. 82(4), 647-655.
10. Davis, G. M. A Wavelet-Based Analysis of Fractal Image Compression // IEEE Transactions on Image Processing, 1994. 3(2), 213-220.
11. Hartenstein, H. Performance Evaluation of Fractal Image Compression Algorithms // Proceedings of the Data Compression Conference, 1996. pp. 123-132. IEEE.
12. Vrscay, R. E. Iterated Function Systems: Theory, Applications, and the Automatic Generation of Fractals // Journal of Mathematical Imaging and Vision, 1994. 4, 75-100.