

УДК 514.18

DOI: 10.32347/0131-579x.2021.100.118-125

к. т. н., **Залевська О.В.**,
o.zalevska@kpi.ua, ORCID:0000-0002-3163-1695

к. т. н., **Яблонський П.М.**,
upn@ukr.net, 0000-0002-1971-5140

Сидоренко Ю.В., к.т.н.
sulico5@gmail.com, ORCID:0000-0002-1953-0410

Мірошниченко І.В.
goodgod@ukr.net, ORCID: 0000-0001-7383-8013

Ситник А.Ю. *
sytnik.akim@gmail.com, ORCID: 0000-0001-8085-2163

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»

УДОСКОНАЛЕННЯ МЕТОДУ ТА АЛГОРИТМУ ФРАКТАЛЬНОГО СТИСНЕННЯ ГРАФІЧНОГО ЗОБРАЖЕННЯ

*У час інтенсивного розвитку інформаційних систем зростають об'єми даних які необхідні для збереження та обробки інформації, зокрема графічних зображень. Одна з причин цього явища полягає у прагненні до постійного покращення якості контенту, що ми отримуємо. Постійно зростаючий попит на якість зображення вимагає розробки нових та удосконалення існуючих підходів до стиснення інформації. Алгоритми стискування використовують наявність так званого надлишку у даних, який можна усунути при зберіганні даних і відновити при їх відтворенні. Затребувані в даний час методи базуються на зберіганні лише низькочастотних компонентів. Такі методи використовуються у алгоритмах стискування JPEG, MPEG. Недоліком таких алгоритмів є не великий коефіцієнт стискування. В зв'язку з цим виникли методи, що базуються на фрактальній компресії даних. Головною ідеєю методу є зберігання зображення у якості афінних перетворень, що призводить до його стискування. В роботі запропоновано попередню обробляти графічні дані для зберігання їх в вигляді файлу з розширенням *.json. Застосування подальшого алгоритму фрактального стиснення вже до отриманого файлу дозволяє скоротити час необхідний для обробки даних, обчислювальні підрахунки. Отриманий файл матиме переваги фрактального стиснення такі, як швидкість декомпресії, кращий коефіцієнт стиснення та більша роздільна здатність в порівнянні з *.jpeg та *.bmp.*

*Науковий консультант – к. т. н., Залевська О.В.

Не зважаючи на всі переваги фрактальне стиснення графічної інформації використовується досить рідко. Це пов'язано зі складністю алгоритму, відсутності достатньої кількості спеціалістів з даного питання та кошторисом ліцензійного програмного забезпечення. Удосконалення націлене на спрощення алгоритму та його реалізацію дозволить уникнути наведених недоліків та розширить сферу застосування фрактального стиснення.

Ключові слова: фрактальне стиснення; графічні зображення; графічні дані; Python; фрактальна графіка.

Постановка проблеми. В час розвитку комп'ютерних технологій важливою є проблема зберігання інформації та економне використання оперативного простору комп'ютера. Зберігання графічних даних пов'язано з великими об'ємами інформації, особливо, якщо вимагається велика якість зображення. В зв'язку з цим виникає необхідність в:

1. Досліджені існуючих форматів зберігання графічних даних;
2. Досліджені алгоритмів створення та форматів зберігання графічних даних;
3. Удосконаленні існуючих методів фрактального стиснення графічних зображень.

Аналіз основних досліджень і публікацій. Зображення можуть бути відтворені за допомогою невеликої кількості вихідних даних. Метод системи ітеративних функцій використовує коефіцієнти афінних перетворень для побудови складних фракталів [1]. Коефіцієнти зберігаються у текстовому файлі. Використовуючи коефіцієнти відновлення вхідне зображення потребує задіяння більших об'ємів пам'яті комп'ютера та більшу пропускну здатність каналів зв'язку [2]. Наприклад, ілюстрація розміром 640x480 точок займає близько 1 Мбайт, такий самий розмір має весь текст книги у 300 сторінок [3]. Створення електронних бібліотек і мультимедійних енциклопедій неможливо без використання програм компресорів з великим коефіцієнтом стискання і декомпресорів, що швидко розпаковують файли [4]. Кожне зображення володіє властивістю самоподібності. Встановлення осередків такої самоподібності та зберігання лише її коефіцієнту дозволяє стискати зображення в відношенні 10 000:1 в залежності від кількості самоподібних ділянок та початкового розміру зображення [5]. Цей процес не залежить від роздільної здатності вихідного зображення. Практичні реалізації фрактального компресора пропонують різні рівні стиснення [6]. Нижні рівні мають більш спокійні критерії пошуку, щоб скоротити час обробки, але з втратою деталей. Вищі рівні дають дуже добрі деталі, але обробка кожного зображення займає багато часу. У роботі [7] детально показано, як алгоритм під назвою Ігри Хаосу використовується для побудови системи ітерованих функцій за допомогою листа папороті.

Застосування стискання зображення з використанням алгоритмів фрактальної геометрії зберігає зображення в надмалому об'ємі без втрати роздільної здатності.

Формування цілей статті. Надати апарат, що удосконалює та оптимізує алгоритм фрактального стиснення графічної інформації шляхом модифікації вхідних даних.

Основна частина. Фрактальне стиснення засноване на пошуку в зображенні самоподібних областей, близьких один до одного по розподілу інтенсивностей пікселів, і розрахунку коефіцієнтів, що пов'язують їх геометричні і оптичні параметри. В результаті кодування одержується компактний набір параметрів системи ітерованих функцій.

Проблема застосування системи ітераційних функцій для стиснення зображень полягає в перетворенні простору зображення в деяку його частину, оскільки, реальне зображення не має повної самоподібності.

Внаслідок цього виникає необхідність у введенні системи сегментованих ітераційних функцій (PIFS). Відмінність цієї системи від системи ітерованих функцій полягає в тому, що кожен оператор діє не на всі зображення, а тільки на певну його ділянку. Отже, зображення потребує попередньої кластеризації.

Зображення описуємо за допомогою серії стискаючих перетворень (в окремому випадку афінних), які будуть застосовані багаторазово на будь-якій ділянці кластеризації. Подання цих перетворень у вигляді даних за обсягом набагато менше вихідного зображення. Це пов'язано з можливістю зберігання лише перетворень за допомогою яких можливо відновити вхідне зображення. Декомпресія за методом фрактального стиснення полягає в побудові атрактора за допомогою коефіцієнта стискання системи ітеруєчих функцій (СІФ). Для відновлення закодованого зображення в якості початкового може бути вибрано будь-яке, в тому числі і чорне зображення.

Розглянемо одну з можливих структур програмного застосунку, що реалізує алгоритм фрактального стиснення. Зображення представляємо у вигляді двійкового зображення унікальність якого визначається рекурсивною властивістю алгоритму, що перетворює точку зображення на іншу. Оскільки, ітераційне застосування перетворень до довільних вихідних точок сходиться то зображення представляє собою атрактор. Ітераційний закон можливо задати двома методами:

1. Застосування кожного перетворення до поточного набору точок (пікселів та об'єднання точок, що потрапили в один і той же піксель,).
2. Вибір послідовності перетворень відповідно до ймовірностей та використовувати їх для поступового перетворення координат точки в запис відвіданих пікселів.

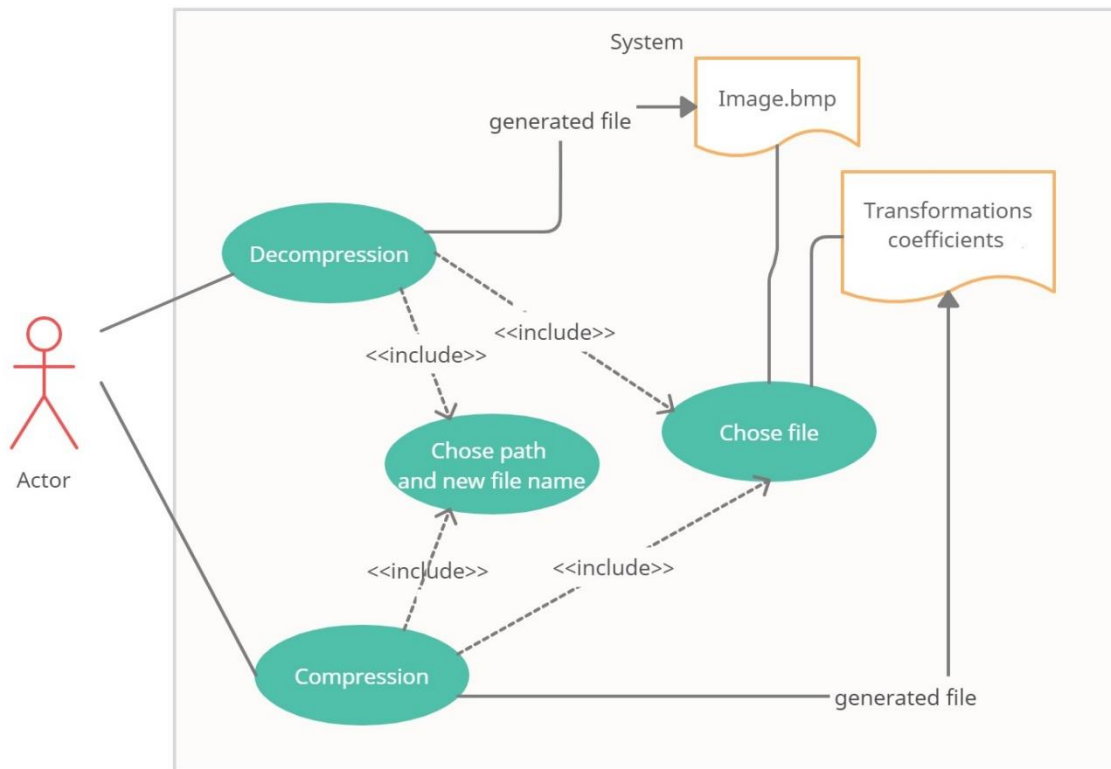


Рис. 1 Діаграма можливого вибору етапу алгоритму фрактального стиснення

Імовірності, використані у другому пункті, не впливають на зображення (аттрактор), зрештою створене, за умови, що вони ненульові: однак зображення покривається найбільш рівномірно і швидко, якщо ймовірності пропорційні площі, в яку перетворюється умовна одиниця. Обидва методи збігаються зі швидкістю залежною від найменш стислого перетворення. Перший метод застосовується при роботі в просторі зображень, зі стисненням перетворень, спотворюючи рухомі (двійкові) пікселі, що розташовані в растрових масивах у пам'яті комп'ютера. Ключовим аспектом алгоритму є те що роздільна здатність зображення може починатися з низької і постійно зростати. Для реалізації другого алгоритму необхідно задати забарвлення кожного пікселя на кожному кроці перетворення. Якщо зображення було розділене на сегменти, то до кожного з них необхідно застосовувати своє перетворення. Спростити даний метод можливо за допомогою підрахунку однакових за забарвленням пікселів.

Для реалізації алгоритму запропоновано використовувати мову програмування Python. Порівняно з іншими сучасними мовами програмування Python досягає значно кращих результатів за короткий час. Для даної мови створено досить велику кількість бібліотек, що дають можливість адаптувати готові рішення під потребу користувача. Головна складність алгоритму полягає у імпортуванні та експортуванні графічного зображення. Пропонуємо для попередньої обробки зображення застосовувати бібліотеки мови програмування Python. Програма

реалізація має забезпечувати як компресію зображення, так і декомпресію, оскільки без знання методів реалізації алгоритму неможливо відновити початкове зображення. В зв'язку з цим на рис. 1 наведено можливу діаграму вибору етапу фрактального стиснення.

Структуру програмного застосунку реалізованого на Python показано на рис. 2. Це програмне рішення дозволяє імпортувати зображення (в даному випадку формату .bmp) у вигляді матриці значень кожного пікселя, такий формат є дуже зручним для обробки, оскільки ця реалізація потребує безпосереднього порівняння фрагментів зображення за їх числовими значеннями.

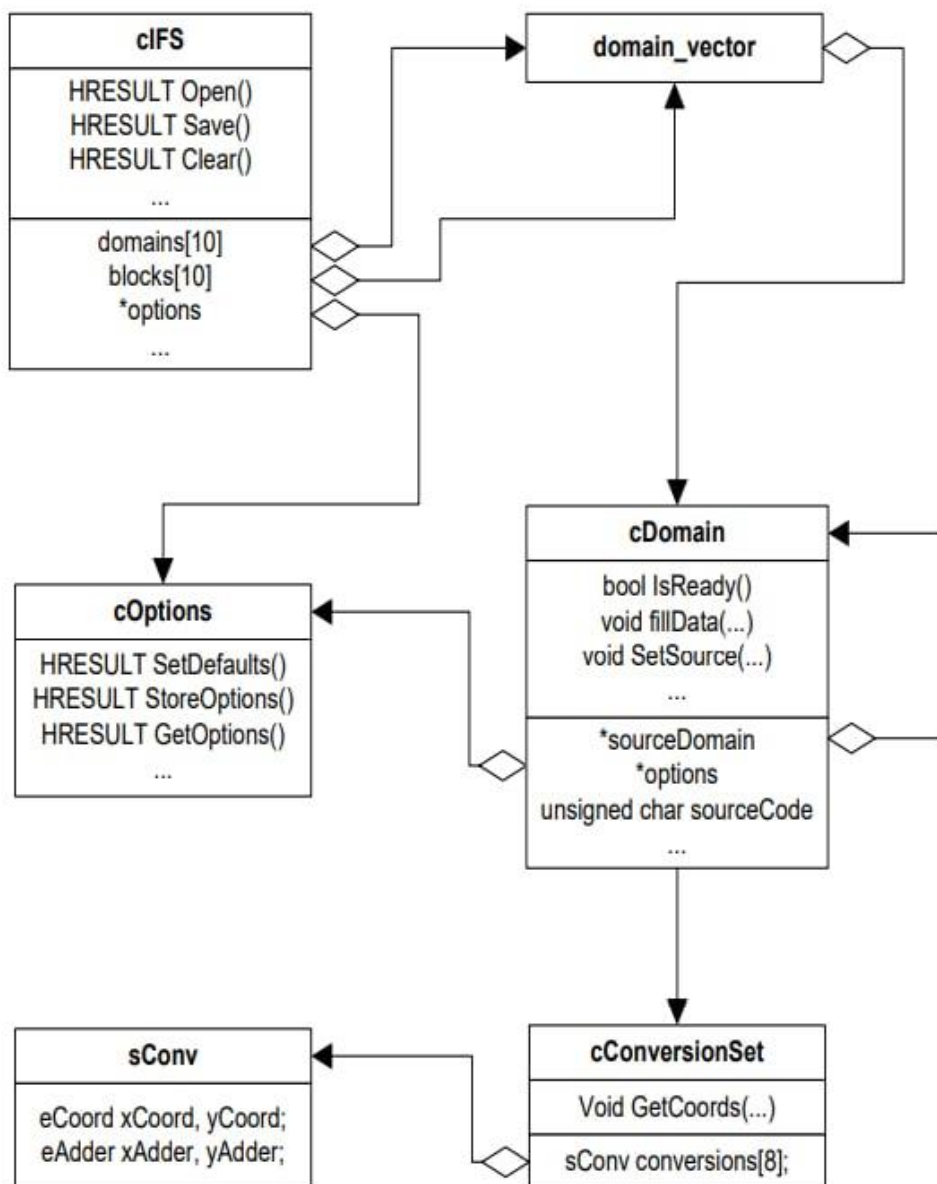


Рис. 1 Структура програмного застосунку для реалізації алгоритму стиснення

Висновки та перспективи. Порівнюючи інші методи стискання встановлено, що фрактальне стискання позбавлено таких недоліків як втрата тонких деталей, залежність від роздільної здатності початкового зображення, збільшення часу стиснення та декомпресії при підвищенні

роздільної здатності зображення. Оптимізація та удосконалення алгоритмів фрактального стискання надає значні переваги як у роботі з даними, так і у роботі з генерацією графічних складових або анімації. Спрощення даного алгоритму розширює сферу його застосування.

Література

1. *Anson, L.F.*, "Fractal Image Compression", BYTE, Oct. 1993, p. 195-202.
2. *Barnsley, M.F.*, *Fractal Everywhere*, 2nd ed., Academic Press, San Diego, 1993.
3. *Barnsley, M.F.*, "Fractal Image Compression", Notices of the AMS, June 1996, p. 657-662.
4. *Barnsley, M.F.*, and *Demko, S.*, "Iterated function systems and the global construction of fractals", Proc. R. Soc. London, A 399 (1985), p. 243-275.
5. *Barnsley, M.F.*, and *Hurd, L.P.*, *Fractal Image Compression*, AK Peters, Ltd., Wellesley, Massachusetts, 1992.
6. *Vanin V. V.*, *Zalevs'ka O. V.* Opisanie ustoychivyykh dinamicheskikh sistem metod fraktal'noy approksimatsii Modern problems of modeling. 2015. №. 4.
7. 'An Introduction to Fractal Image Compression' Literature Number: BPRA065 Texas Instruments Europe October 1997.

Referenses

1. *Anson, L.F.*, "Fractal Image Compression", BYTE, Oct. 1993, p. 195-202.
2. *Barnsley, M.F.*, *Fractal Everywhere*, 2nd ed., Academic Press, San Diego, 1993.
3. *Barnsley, M.F.*, "Fractal Image Compression", Notices of the AMS, June 1996, p. 657-662.
4. *Barnsley, M.F.*, and *Demko, S.*, "Iterated function systems and the global construction of fractals", Proc. R. Soc. London, A 399 (1985), p. 243-275.
5. *Barnsley, M.F.*, and *Hurd, L.P.*, *Fractal Image Compression*, AK Peters, Ltd., Wellesley, Massachusetts, 1992.
6. *Vanin V. V.*, *Zalevs'ka O. V.* Opisanie ustoychivyykh dinamicheskikh sistem metod fraktal'noy approksimatsii Modern problems of modeling. 2015. №. 4.
7. 'An Introduction to Fractal Image Compression' Literature Number: BPRA065 Texas Instruments Europe October 1997

к. т. н., **Залевська О. В.**,
o.zalevska@kpi.ua, ORCID[^] 0000-0002-3163-1695

к. т. н., **Яблонський П. Н.**,
ypn@ukr.net, ORCID: 0000-0002-1971-5140

к. т. н., **Сидоренко Ю. В.**,

sulico5@gmail.com ORCID:0000-0002-1953-0410

Мирошниченко И. В.

goodgod@ukr.net, ORCID: 0000-0001-7383-8013

Ситник А.Ю.

sytnik.akim@gmail.com, ORCID: 0000-0001-8085-2163

Национальный технический университет Украины «Киевский
политехнический институт имени Игоря Сикорского»

СОВЕРШЕНСТВОВАНИЕ МЕТОДОВ И АЛГОРИТМА ФРАКТАЛЬНОГО СЖАТИЯ ГРАФИЧЕСКОГО ИЗОБРАЖЕНИЯ

Во время интенсивного развития информационных систем растут объемы данных необходимых для хранения и обработки информации, в частности графических изображений. Одна из причин этого явления заключается в стремлении к постоянному улучшению качества контента, мы получаем. Постоянно растущий спрос на качество изображения требует разработки новых и совершенствования существующих подходов к сжатию информации. Алгоритмы сжатия используют наличие так называемого избытка в данных, который можно устранить при хранении данных и восстановить при их воспроизведении. Востребованы в настоящее время методы основаны на хранении только низкочастотных компонентов. Такие методы используются в алгоритмах сжатия JPEG, MPEG. Недостатком таких алгоритмов не большой Коэффициент сжатия. В связи с этим возникли методы, основанные на фрактальной компрессии данных. Главной идеей метода является хранение изображения в качестве аффинных преобразований, что приводит к его сжатию.

*В работе предложено предварительно обрабатывать графические данные для хранения их в виде файла с расширением * .json. Применение дальнейшего алгоритма фрактального сжатия уже полученному файлу позволяет сократить время обработки данных, вычислительные подсчеты. Полученный файл будет иметь преимущества фрактального сжатия такие, как скорость декомпрессии, лучший коэффициент сжатия и большая разрешающая способность по сравнению с * .jpeg и * .bmp.*

Несмотря на все преимущества фрактальное сжатие графической информации используется довольно редкая. Это связано со сложностью алгоритма, отсутствия достаточного количества специалистов по данному вопросу и сметой лицензионного программного обеспечения. Совершенствование нацелено на упрощение алгоритма и его реализацию позволит избежать указанных недостатков и расширит сферу применения фрактального сжатия.

Ключевые слова: фрактальное сжатие, графические изображения, графические данные, Python, фрактальная графика.

Ph.D., **Olha Zalevskaya**,
o.zalevska@kpi.ua, ORCID: 0000-0002-3163-1695

Ph.D., **Petro Yablonskyi**
ypn@ukr.net, ORCID :0000-0002-1971-5140

Ph.D., **Iuliia Sydorenko**,
sulico5@gmail.com, ORCID:0000-0002-1953-0410

Akim Sytnik
sytnik.akim@gmail.com, ORCID: 0000-0001-8085-2163

IMPROVEMENT OF METHODS AND ALGORITHM OF FRACTAL COMPRESSION OF A GRAPHIC IMAGE

During the intensive development of information systems, the volumes of data necessary for storing and processing information, in particular graphic images, grow. One of the reasons for this phenomenon is the desire to continually improve the quality of the content we receive. The ever-growing demand for image quality requires the development of new and improvement of existing approaches to information compression. Compression algorithms use the presence of so-called surplus in the data, which can be eliminated during data storage and restored during playback. Currently in demand methods are based on storing only low-frequency components. Such methods are used in JPEG, MPEG compression algorithms. The disadvantage of such algorithms is not a large compression ratio. In this regard, methods have emerged based on fractal data compression. The main idea of the method is to store the image as affine transformations, which leads to its compression.

*The paper proposes preliminary processing of graphic data for storing them in the form of a file with the *.json extension. The application of a further fractal compression algorithm to the already obtained file allows to reduce the processing time of data, computational calculations. The resulting file will have the advantages of fractal compression such as decompression speed, better compression ratio and higher resolution compared to *.jpeg and *.bmp.*

Despite all the advantages, fractal compression of graphic information is used quite rarely. This is due to the complexity of the algorithm, the lack of a sufficient number of specialists on this issue and the estimate of the licensed software. The improvement is aimed at simplifying the algorithm and its implementation will avoid these drawbacks and expand the scope of fractal compression.

Keywords: fractal compression; graphic images; graphic data; Python; fractal graphics.