

ОПТИМІЗАЦІЯ ШВИДКОСТІ ФРАКТАЛЬНОГО УЩІЛЬНЕННЯ ЗОБРАЖЕНЬ НА ОСНОВІ ПОПЕРЕДНЬОГО ВІДБОРУ

Вінницький національний технічний університет

Анотація

Проаналізовано фрактальний метод стиснення зображень. Проведено аналіз варіантів оптимізації та підвищення швидкодії фрактального ущільнення зображень. На основі досліджень розроблено швидкодуючий алгоритм фрактального ущільнення та його програмна реалізація.

Ключові слова - фрактальне ущільнення зображень, системи ітеруючих функцій, класифікація блоків.

Abstract

The fractal image compression method is analyzed. The analysis of variants of optimization and increase of speed of fractal coding of images is carried out. On the basis of researches, a high-performance fractal sealing algorithm and its software implementation are developed.

Keywords - fractal compaction of images, systems of deprivation functions, classification of blocks.

Вступ

Зображення, які представлені в цифровій формі, необхідно зберігати на носіях та передавати каналами зв'язку. Для економії пам'яті та більш ефективного використання ресурсів системи створюють спеціальні алгоритми кодування [1-4]. Зображення – це особливий вид даних, який має надлишковість в двох вимірах, що дає додаткові можливості для ущільнення [4]. Одним із перспективних методів ущільнення зображень є фрактальний метод [4]. Фрактальне кодування - це математичний процес для кодування растрів, які містять реальне зображення, в сукупність математичних даних, що описують фрактальні властивості зображення. Цей вид кодування заснований на тому, що усі природні і більшість штучних об'єктів містять надмірну інформацію у вигляді однакових блоків зображення, що повторюються. Вони отримали назву фракталів. Фрактал - це структура, яка складається з подібних форм і малюнків, що зустрічаються в різних розмірах.

Аналіз фрактального методу ущільнення зображень

З фізичної точки зору фрактальне кодування ґрунтується на твердженні, що зображення містить афінну надлишковість. Математична модель, яка використовується при фрактальному стисненні зображень, називається системами ітеруючих функцій (Iterated Function Systems – IFS). Системи ітеруючих функцій містять набір стискальних перетворень w_i , які можливо задати так [4]:

$$W(S) = \bigoplus_{i=1}^n w_i(S) \quad (1)$$

де S – зображення; w_i – набір стискальних перетворень.

Відповідно до теореми Банаха, існує певний клас відображень, які називаються стискальними і для них справедливо таке твердження: якщо до якогось зображення f_0 ми почнемо багаторазово застосовувати відображення W таким чином, що:

$$f_i = W(f_0), f_{i+1} = W(f_i) \quad (2)$$

то при i , що прямує до нескінченності, ми отримаємо таке саме зображення незалежно від того, яке зображення ми взяли за f_0 :

$$f = \lim_{i \rightarrow \infty} f_i \quad (3)$$

Зображення f називається нерухомою точкою перетворення W або атрактором.

В якості перетворень w_i використовуються афінні відображення:

$$w_i \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{bmatrix} a_i & b_i & 0 \\ c_i & d_i & 0 \\ 0 & 0 & S_i \end{bmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} dx \\ dy \\ O_i \end{pmatrix} \quad (4)$$

де a_i, b_i, c_i, d_i – афінні коефіцієнти деформації, стиснення, обертання; dx, dy – коефіцієнти переміщення; x, y – координати точки, що перетворюється; z – її інтенсивність. Параметр S_i керує контрастністю, а O_i – яскравістю зображення.

Алгоритм фрактального кодування зображень можна описати так. Процес ущільнення починається з того, що зображення спочатку розділяється на блоки, що не перекриваються (рангові області), а потім на доменні блоки, які можуть взаємно перекриватися, як це показано на рис. 1 [2].

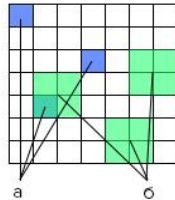


Рис. 1. Відбір блоків на зображенні: а – рангових, б – доменних

Одна з можливих схем кодування зображень фрактальним методом, запропонована Арно Жакеном (Arnaud Jacquin), містить такі етапи [4]:

– Зображення розділяється на області, що примикають одна до одної розміром $N \times N$ (рангові області).

– Задається набір доменних областей. Доменні області можуть перекриватися, вони не повинні обов'язково закривати всю поверхню зображення. Розміри доменних областей звичайно вибирають $2N \times 2N$.

– Для кожної рангової області підбирається доменна область, яка після афінних перетворень найбільш точно апроксимує рангову область. На практиці застосовується вісім варіантів відображення одного квадрата в інший з використанням афінних перетворень. Це повороти зображення на кути 0, 90, 180, 270 градусів відносно його центра і перетворення симетрії відносно ортогональних осей, які проходять через центр фрагменту перпендикулярно його сторонам.

– Точність апроксимації F визначається за допомогою середньоквадратичного критерію:

$$F = \sum_{i,j} (Sd_{ij} + O_{ij} - r_{ij})^2, \quad (5)$$

де d_{ij} – значення, отримані в результаті усереднення по фрагментах з розмірами 2×2 елементів доменної області, що приводить її розмір до розміру рангової області; r_{ij} – значення елементів рангової області. Зміщення O_{ij} може бути як константою, так і описуватися поліномами першого, другого, третього порядків.

Привівнявши до нуля часткові похідні від виразу по S і O :

$$\frac{\partial F}{\partial S} = 0, \quad \frac{\partial F}{\partial O} = 0, \quad (6)$$

знайдемо значення S і O , при яких досягається мінімум виразу:

$$O = \frac{1}{n^2} \left(\sum_{i,j} r_{ij} - S \sum_{i,j} d_{ij} \right) \quad (7)$$

$$S = \frac{n^2 \sum_{i,j} r_{ij} d_{ij} - \sum_{i,j} r_{ij} \sum_{i,j} d_{ij}}{n^2 \sum_{i,j} d_{ij}^2 - \left(\sum_{i,j} d_{ij} \right)^2} \quad (8)$$

Доменні блоки звичайно вибирають з кроком $n/2$ при $n=4$. У вихідний файл записуються такі параметри:

- координати доменної області з найменшим значенням F_{min} ;
- значення для O і S , отримані згідно з формулами (7, 8);

- номер афінного перетворення.

Алгоритм декодування полягає в тому, що беруться два екземпляри одного і того ж зображення А і Б, розподіл яскравості в яких неважливий. На цих зображеннях виділяються області, межі яких співпадають з межами рангових областей і доменів, а потім, використовуючи відомі значення афінних коефіцієнтів, по доменах, виділених на зображенні Б, знаходяться розподіли яскравості в рангових областях зображення А. Після цього зображення А і Б міняються місцями і операція повторюється.

Основним недоліком фрактального методу є низька швидкість кодування, яка пов'язана з тим, що для отримання високої якості зображення для кожного рангового блоку необхідно виконати перебір всіх доменних блоків, і для кожного доменного блоку необхідно виконати не менше восьми афінних перетворень [5].

В цифровій обробці сигналів швидкодія методу оцінюється кількістю арифметичних операцій, необхідних для виконання перетворення.

Отже, для підвищення швидкодії потрібно зменшити загальну кількість арифметичних операцій

Результати дослідження

Для підвищення швидкодії та ефективності фрактального кодування зображень використовують ряд методів оптимізації. Найпростіший і спосіб фрактального кодування є перевірка кожного доменного блоку і виконання відповідних обчислень. Такий спосіб називається повним пошуком або повним перебором. При кодуванні зображень природного походження можна підвищити швидкодію кодування, прийнявши $S=1$, оскільки враховуючи статистику зображень завжди знайдеться доменний блок, який апроксимує заданий ранговий блок з необхідною точністю. Тоді з виразів (5), (7) одержимо [4]:

$$F = \sum_{i,j} (d_{ij} + O_{ij} - r_{ij})^2 \quad (9)$$

$$O = \frac{1}{n^2} \left(\sum_{i,j} r_{ij} - \sum_{i,j} d_{ij} \right) \quad (10)$$

Контрастність декодованого зображення може бути відновлена іншими методами. Таке спрощення дозволяє знизити кількість арифметичних операцій на 60 % і відповідно підвищити швидкість ущільнення. Найбільш відомі методи підвищення швидкодії кодування зображень фрактальним методом такі [4-6]:

1. Пошук доменних блоків, для яких F не перевищує заданого значення.
2. Локальний та сублокальний пошук.
3. Ізометричне передбачення.
4. Класифікація доменних і рангових блоків, ранговий порівнюється з доменними блоками того ж самого класу.

Класифікація доменних блоків передбачає попередній відбір доменних блоків для кожного рангового блоку. Для попереднього відбору для кожного рангового та доменного блоку можна обчислити коефіцієнти деякого двовимірного перетворення і за їх значенням виконати класифікацію цих блоків. Найбільш економічним є перетворення Уолша-Адамара, оскільки для його обчислення достатньо лише операцій додавання [2]. Для класифікації доменних і рангових блоків використовується лише декілька значень коефіцієнтів перетворення (рис. 2), що зменшує кількість обчислень та підвищує швидкість кодування.

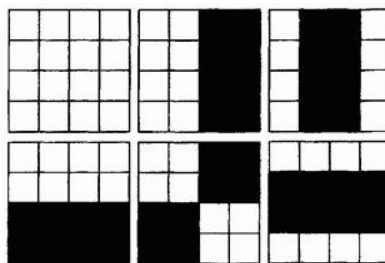


Рис.2. Базисні функції перетворення Уолша-Адамара

Перед відбором доменного блоку для рангового вираховуємо середнє відхилення заданого рангового блоку з кожним доменним, за формулою:

$$SQ = \sum_i (wr_i - wd_i)^2 \quad (11),$$

де wr_i – результат перетворення Уолша-Адамара за i -тим базисним зображенням рангового блоку, wd_i – результат перетворення Уолша-Адамара за i -тим базисним зображенням доменного блоку.

Відбирається задана кількість доменних блоків з найменшим відхиленням для поточного рангового блоку та виконується стандартний алгоритм фрактального ущільнення з відібраними доменними блоками.

Якщо порахувати необхідну кількість операцій для стиснення зображення в градаціях сірого розміром 512x512 пікселів при розмірі рангового блоку 8 пікселів.

Для алгоритму запропонованим Жакеном загальна кількість математичних операцій складе 2147483648.

Для алгоритму з попереднім відбором, запропонованим у даній статті загальна кількість операцій складе 239914189.

Отже при заданих умовах вдалося значно зменшити кількість арифметичних операцій.

Інший підхід передбачає класифікацію як доменних так і рангових блоків і фрактальні перетворення виконуються лише для одного рангового блоку із списку подібних. Кожен новий ранговий блок який створюється отримує ідентифікатор схожості (IC) який дорівнює «-1» що означає, що даний ранговий блок не схожий ні на один із вже отриманих.

Під час виконання розбиття зображення на рангові блоки необхідно кожен наступний ранговий блок порівняти з усіма попередніми які мають IC = «-1». Якщо різниця між кожним пікселем нового рангового блоку та попередніми ранговими блоками відсутня, тобто

$$r[i][j] - r_new[i][j] = 0, \quad (12)$$

то два рангові блоки є однаковими і в IC нового рангового блоку необхідно записати номер по порядку (ПП) рангового блоку з яким виконалось порівняння та було отримано $r[i][j] - r_new[i][j] = 0$. Наприклад, r має номер ПП = 0 і значення IC = -1, r_new має номер ПП=2, значення IC дорівнює номеру ПП блоку r , тобто IC = 0 (рис. 3).



Рис.3. Класифікація рангових блоків

Під час розбиття на доменні блоки необхідно порівняти кожен новий усереднений доменний блок з усіма унікальними доменами, які було отримано раніше. Порівняння виконується наступним чином. Необхідно знайти різницю між кожним пікселем з унікальних та новим за формулою

$$d[i][j] + O - d_new[i][j], \quad (13)$$

де O – коефіцієнт зміщення по яскравості між блоками. Якщо в результаті отримано значення яке по модулю не перевищує певного порогу то новий доменний блок не потрібно додавати у список унікальних, якщо ж хоча б один піксель перевищує встановлений поріг, то такий доменний блок необхідно вважати унікальним (значення різниці знаходиться в межах від -510 до +510, оскільки кожен піксель не перевищує значення 255 та зміщення по яскравості знаходиться в межах від -255 до +255).

Після розбиття зображення на доменні і рангові блоки виконується власне ущільнення фрактальним методом. Відмінністю є те що коли обирається ранговий блок, то перевіряється

значення IC і якщо $IC = -1$, то виконується перебір унікальних доменів, щоб знайти найбільш схожий до даного рангового блоку. Якщо IC не дорівнює -1 то замість виконання перебору унікальних доменів, необхідно лише дізнатись який доменний блок було знайдено для рангового блоку R з номером ПП що дорівнює IC . Саме цей підхід і досліджується у подальшому.

Висновки

Аналіз фрактального ущільнення зображень показав, що даний метод здатен забезпечити найкраще співвідношення ступеня ущільнення і якості відновленого зображення та має хороші перспективи для подальшого розвитку.

Запропоновано декілька оригінальних методів підвищення швидкості фрактального ущільнення. Зокрема, за рахунок попереднього відбору близьких доменних блоків за коефіцієнтами двовимірного ортогонального перетворення, що зменшує простір пошуку для кожного рангового блоку і відповідно підвищує швидкість фрактального ущільнення. Інший підхід передбачає класифікацію як доменних так і рангових блоків і фрактальні перетворення виконуються лише для одного рангового блоку із списку подібних.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Методы сжатия данных. устройство архиваторов, сжатие изображений и видео. / Ватолин Д., Ратушняк А., Смирнов М., Юкин В. – м.: диалог-мифи, 2002. – 384 с.
2. Сэломон д. сжатие данных, изображений и звука. / д. сэломон – м.: техносфера, 2004. – 368 с.
3. Кожем'яко В.П. Аналіз та перспективи розвитку кодування зображень / В.П. Кожем'яко, В.П. Майданюк, К.М. Жуков - Вісник ВПІ, 1999, № 3. – 42-48с.
4. Майданюк В. П. Методи і засоби комп'ютерних інформаційних технологій. кодування зображень. навчальний посібник // Вінниця: вду, 2001. – 63 с.
5. Кожем'яко В.П., Майданюк В.П., Жуков К.М., Хамді Р.Р., Піка С.О. Фрактальне стиснення зображень природного походження // міжнародний науково-технічний журнал "Вимірювальна та обчислювальна техніка в технологічних процесах", Хмельницький, 1999, № 2, с. 50-54
6. Kozhemiako V.P., Maidanuk V.P., Pika S., Zhukov K.M. Speeding up of fractal image compression // proceeding of spie, 2001, vol. 4425, p. 9-16.

Лішук Олександр Олександрович — аспірант кафедри програмного забезпечення Вінницького національного технічного університету, м. Вінниця, Україна.

Майданюк Володимир Павлович – к.т.н., доцент кафедри програмного забезпечення Вінницького національного технічного університету, м. Вінниця, Україна.

Науковий керівник: *Майданюк Володимир Павлович* – к.т.н., доцент кафедри програмного забезпечення Вінницького національного технічного університету, м. Вінниця, Україна.

Lishchuk Oleksandr O. — post-graduate student of the Department of Software at Vinnytsia National Technical University, Vinnytsia, Ukraine.

Maydanyuk Volodymyr P. - Candidate of Technical Sciences, Associate Professor, Department of Software, Vinnytsia National Technical University, Vinnytsya, Ukraine.

Supervisor: *Maydanyuk Volodymyr P.* - Candidate of Technical Sciences, Associate Professor, Department of Software, Vinnytsia National Technical University, Vinnytsya, Ukraine.