

Торянік В.В., канд.техн.наук

Національний технічний університет "ХПИ" (м. Харків)

Глібоко О.А., канд.техн.наук

Національний технічний університет "ХПИ" (м. Харків)

Максимова М.А. канд.техн.наук

Національний технічний університет "ХПИ" (м. Харків)

РОЗРОБКА СХЕМ АЛГОРИТМІВ СТИСКАННЯ РАСТРОВИХ ЗОБРАЖЕНЬ

Розглянуто спосіб представлення растрових зображень у вигляді ієрархічної структури системи матриць – тетрадного дерева. Наведено алгоритми пакування растрових зображень, алгоритми роботи з масивами пікселів для отримання зображень визначених розмірів та ваги.

The method of representation of raster images as hierarchical structures of system of matrixes - tetrahedr tree is considered. Algorithms of packing of raster images, algorithms of work with massif of pixels for reception of images of the appointed sizes and weight are resulted.

Постановка проблеми. Растрові зображення, що знаходять застосування майже у всіх галузях людської діяльності, характеризуються зокрема кількістю точкових одиниць інформації, що безпосередньо впливають на визначення розміру та ваги зображення. (Однією з найбільш поширених задач при роботі з ними є можливість їх стискування).

Аналіз досягнень та публікацій. Висока вартість потужних графічних пакетів редагування растрових зображень висуває задачу розробки значно дешевших вітчизняних аналогів, що досить часто орієнтовані на рішення визначеного кола завдань /1,2,3/.

Постановка завдання. Запропонувати алгоритми стискування растрових зображень з метою зменшення їх ваги для подальшого використання в залежності від поставленої мети.

Основна частина. Дані растрових зображень представляються матрицями, елементами яких є пікселі зображення.

Іншим способом зберігання растрових зображень є їх представлення у вигляді дерев. Нехай зображення надано квадратною матрицею M_0 що має розмір $2n \times 2n$. Розіб'ємо її на чотири квадратні матриці M_{00} , M_{01} , M_{02} , M_{03} з розмірами $2n-1 \times 2n-1$.

Кожну результуючу матрицю теж розіб'ємо. Цей процес подовжимо доти, поки розмір результуючих матриць не стане рівним одиниці, тобто доки вони не будуть містити один піксель вихідного зображення.

Таким чином, отримаємо n рівнів матриць-частиць зображення. Якщо помічати кожен матрицю послідовністю символів, що утворена з позначки батьківської матриці та порядкового номеру поточного розбиття, то можна отримати деревовидну структуру. Кожен рівень цього дерева складається з групи по чотири матриці (рис.1). Така структура зветься тетрадним деревом. Тетрадні дерева можна побудувати на зображеннях, які є квадратним масивом з парним числом рядків (стовпців).

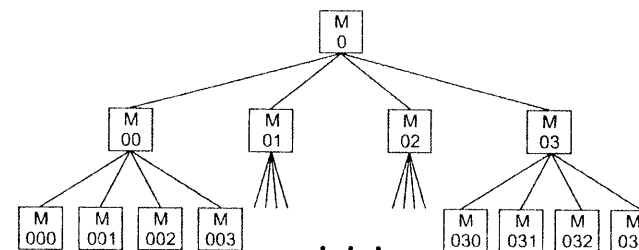


Рис. 1. Растрове зображення, що представлено у вигляді тетрадного дерева.

Якщо зображення представлено матрицею $n \times n$, то кількість рівнів L тетрадного дерева буде складати

$$L = (2l_n n) / l_n 4.$$

Кількість елементів k -того рівня складає 4^k матриць. Загальна кількість елементів дерева (вузлів) дорівнює

$$N = \sum_{k=0}^L$$

Вхідні дані процедури Get_level: L – номер рівня дерева; P – матриця зображення L -того рівня.

Вихідні дані: R – матриця зображення $(L-1)$ рівня дерева; D – матриця різниць $(L-1)$ рівня дерева.

Нехай зображення задано матрицею P . Побудова тетрарного дерева здійснюється починаючи з формування останнього його рівня. Для цього виконується послідовний вибір груп 2×2 пікселів з матриці P . Процес реалізується за допомогою двох вкладених циклів за рядками та стовпцями матриці P з кроком 2. В циклах формуються початкові та кінцеві індекси i_1, i_2, j_1, j_2 поточної групи по 4 пікселі. Ці індекси визначають зв'язок поточної групи з вихідною матрицею P . Наступним кроком визначається середня інтенсивність світіння PAV поточної групи пікселів

$$PAV = SP/4,$$

де SP – сума інтенсивностей світіння пікселів поточної групи.

Далі визначається новий елемент результуючої матриці R рівня, що формується, та матриця різниць D .

Елементи матриці D є різницею інтенсивностей світіння пікселя групи та середньої інтенсивності. Кількість елементів матриці D , що зберігаються, для поточної групи з чотирьох пікселів дорівнює трьом. Невистачаючий четвертий елемент визначається виразом

$$I = R(x,y) - (D(x,y,1,1) + D(x,y,1,2) + D(x,y,2,1)).$$

Алгоритм побудови рівня тетрарного дерева, що реалізується у вигляді процедури `Get_level`, представлений на рис.2.

Вищий рівень можна отримати при використанні алгоритму для результуючої матриці зображення, що отримана на попередньому рівні. Цей процес подовжується доти, доки результуюча матриця не буде містити один елемент(рис.3).

При побудові тетрарного дерева інформація про зображення зберігається в матриці D . Розмір пам'яті, що необхідна для зберігання тетрарного дерева, більший за розмір пам'яті, що необхідна для зберігання вихідного зображення. Скоротити необхідну кількість пам'яті можна, якщо зображення має ділянки одного кольору.

Наприклад, якщо елементи матриці D для якийсь групи пікселів дорівнює нулю (а це свідчить про однаковий колір пікселів групи), то замість зберігання трьох нулів можна в матрицю D ввести умовне значення, яке може зустрітися як відхилення від середнього значення яскравості. Цим значенням може бути число 255, як максимальне значення яскравості компонентів кольору.

Розглянемо елементи матриці R для якогось рівня тетрарного дерева. Для зображення, що має зони однакового кольору, в матриці R буде ряд елементів, що мають однакове значення. Для зберігання інформації про даний рівень дерева елементи матриці R послідовно записуються в довготривалу пам'ять комп'ютера.

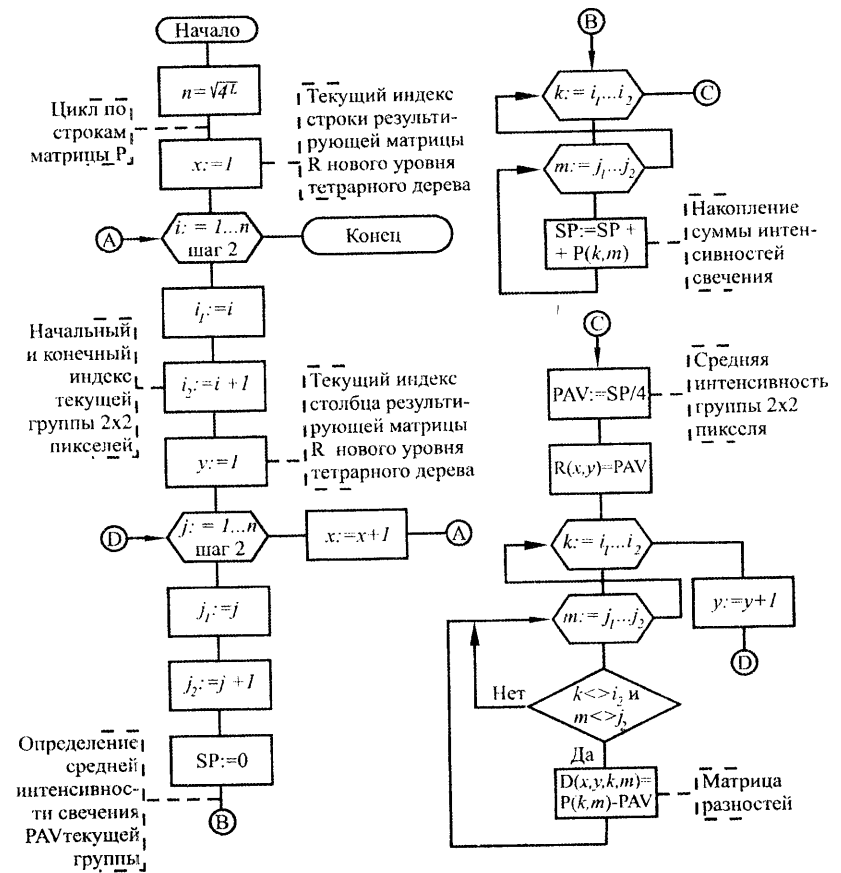


Рис.2. Алгоритм побудови рівня тетрарного дерева

При цьому, якщо йдуть підряд декілька елементів з однаковими значеннями, то доцільно зберігати не всі ці елементи, а їх кількість K та значення цих елементів (RGB).

Такі прийоми мають назву стискання зображення. Алгоритм стискання або пакування рівня зображення представлений на рис.4.

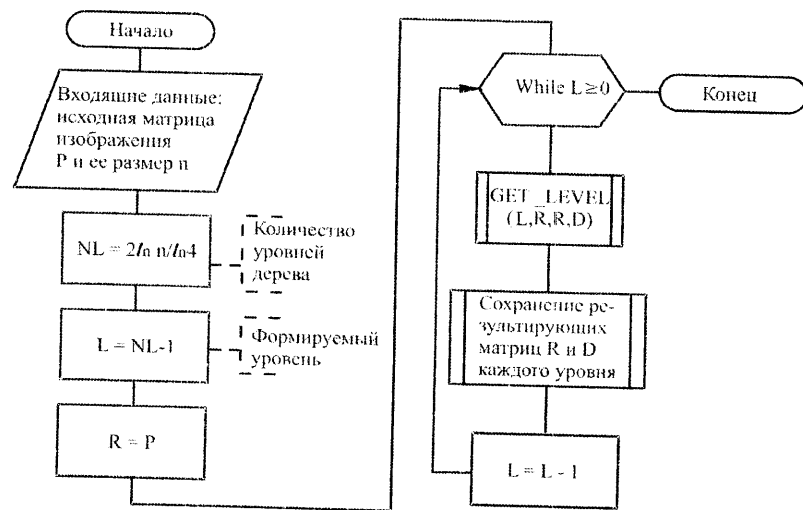
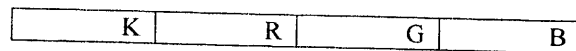


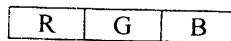
Рис.3. Алгоритм побудови повного тетрадного дерева

Для кольорових зображень запис матриці R можна здійснювати поблочно. Структура блока наступна:

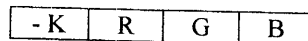


K – 32 бітне ціле; R, G, B – 8 бітне.

При записи інформації про одиничний піксель значення K буде дорівнювати одиниці. Таким чином, до корисної інформації RGB додаються зайві 32 біт для зберігання K. Цього можна уникнути, якщо при K = 1 структура блока будет виглядати наступним чином



при для K > 1



Від'ємний знак при K використовується для розпізнавання поточної структури при читанні та розпакуванні стиснутого зображення. В алгоритмі під C розуміється трикомпонентна величина (RGB).

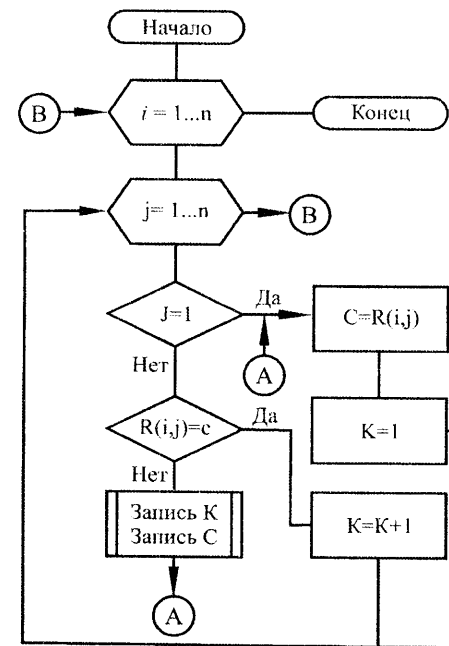


Рис.4. Алгоритм пакування зображення.

Надана методика може бути реалізована в різноманітних галузях, коли необхідно формування «легких» растрових зображень для їх зберігання, передачі та подальшого зручного застосування з економними витратами ресурсів пам'яті.

Список літератури

1. Торянік В.В., Глібо О.А., Савченко Л.М. Алгоритм визначення колірних зон растрового зображення. "Сучасні проблеми геометричного моделювання". Збірка праць Міжнародної науково-практичної конференції. - Харків, 2001.- с.223.
2. Торянік В.В., Глібо О.А., Савченко Л.М. Алгоритм порівняння контурів плоских геометричних форм "Актуальні проблеми геометричного моделювання". Збірка праць Міжнародної науково-практичної конференції. - Львів, 2003.- с.277-279.
3. Торянік В.В., Глібо О.А., Максимова М.А. Розробка схем алгоритмів гістограм растрових зображень. Геометричне та комп'ютерне моделювання. 36. наук. праць. Харківський державний університет харчування та торгівлі. -Харків, 2010.- Вип.27 – с.130

Отримано 18.03.2011, ХДУХТ, м. Харків.

© Торянік В.В., Глібо О.А., Максимова М.А., 2011.