

УДК 004.627

ПРИМЕНЕНИЕ ВЕЙВЛЕТНОГО ПРЕОБРАЗОВАНИЯ ДЛЯ СЖАТИЯ ИЗОБРАЖЕНИЙ

Стройкова К.А., студент

*Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана,
кафедра «Программное обеспечение ЭВМ и информационные технологии»*

*Научный руководитель: Rogozin O.V., к.т.н., доцент
кафедра «Программное обеспечение ЭВМ и информационные технологии»*

bauman@bmstu.ru

Развитие Интернета, рост производительности компьютеров и прогресс в технологии производства цифровых камер, сканеров и принтеров, привели к широкому использованию цифровых изображений. Качество изображений и их размер постоянно растут. Размер графических данных файла пропорционален количеству пикселей в изображении и количеству битов, требуемых для представления каждого пикселя. Необходимо улучшать алгоритмы сжатия данных, представляющих изображения. Сжатие данных важно как для скорости передачи, так и для эффективности хранения [1].

В настоящее время разработаны алгоритмы сжатия без потерь на основе универсальных методов сжатия и алгоритмы сжатия с потерями, использующие особенности графических данных. Продолжаются работы над алгоритмами сжатия с потерями, сохраняющими качество изображения на высоком уровне.

Особенности растровых изображений

Изображения – своеобразный тип данных, характеризуемый следующими особенностями:

- изображение требует для хранения гораздо большего объема памяти, чем текст, что определяет актуальность алгоритмов архивации графической информации,
- человеческое зрение при анализе изображения оперирует контурами, общим переходом цветов и сравнительно нечувствительно к малым изменениям в изображении,
- изображение обладает избыточностью в двух измерениях, то есть соседние точки по горизонтали и вертикали могут быть близки по цвету [2].

Растровое изображение представляет собой двумерный массив чисел. Элементы этого массива – пиксели. Основными характеристиками, необходимыми для отображения

изображения, являются размеры и глубина цвета. Размеры определяются шириной и высотой изображения в пикселях. Глубина цвета определяет количество бит информации на один пиксель. Чем больше глубина цвета, тем шире диапазон цветов. При использовании некоторой системы цветоопределения каждый пиксель представляет собой структуру, полями которой являются компоненты цвета.

Классификация алгоритмов сжатия изображений

Существующие алгоритмы сжатия изображений могут быть классифицированы следующим образом:

- сжатие изображений без потерь
- алгоритм RLE (Run Length Encoding) – кодирование длин повторов
- алгоритм LZW – Lempel, Ziv и Welch
- алгоритм HUFF (Huffman Coding) – алгоритм Хаффмана
- алгоритм JBIG (Joint Bi-level Experts Group) – сжатие 1-битовых черно-белых изображений
- алгоритм Lossless JPEG (Joint Photographic Expert Group) – сжатие полноцветных 8- или 24-битовых изображений в градациях серого
- сжатие изображений с потерями
- алгоритм JPEG (Joint Photographic Expert Group) – алгоритм для сжатия 24-битовых изображений
- фрактальный алгоритм – сжатие с помощью коэффициентов системы итерируемых функций
- алгоритм с использованием вейвлетных преобразований (или рекурсивный, или волновой алгоритм) – сжатие на основе когерентности областей изображения

Алгоритм сжатия изображений с использованием вейвлетных преобразований

Алгоритм с использованием вейвлетных преобразований состоит из двух основных этапов – прямого вейвлетного преобразования и обратного вейвлетного преобразования. При использовании такой последовательности действий происходит кодирование и декодирование графических данных изображения без потерь. При обнулении части коэффициентов после прямого вейвлетного преобразования происходит уменьшение количества ненулевых данных об изображении. Кодирование данных после удаления части вейвлетных коэффициентов обеспечивает сжатие изображения с потерями.

Основная идея вейвлетного преобразования

Рассмотрим вектор из двух точек $\{x_1, x_2\}$. Эти значения заменим на среднее значение a и полуразность d : $a = (x_1 + x_2)/2$ и $d = (x_1 - x_2)/2$. Тогда $\{a, d\}$ является вейвлет-преобразованием исходного вектора. В случае если d мало, исходный вектор может быть представлен значением $\{a\}$, что обеспечивает сжатие хранимой информации. Для вектора большего размера такое преобразование применяется ко всем парам в векторе. Затем преобразование применяется рекурсивно к полученным полусуммам.

Двумерное вейвлетное преобразование

Исходное изображение может быть рассмотрено как одномерный массив, к которому затем применяется вейвлетное преобразование. В этом случае происходит перераспределение данных изображения, и не используется особенность избыточности данных изображений в двух измерениях. При применении вейвлетного преобразования к изображениям будем поочередно преобразовывать строки и столбцы исходного изображения, затем уменьшать рассматриваемую область в два раза. В начале рассматриваемая область соответствует всему изображению.

При обратном вейвлет-преобразовании к данным применяется та же последовательность действий в обратном порядке.

Алгоритм вейвлетного преобразования

Для алгоритма прямого преобразования необходимо инициализировать размер рассматриваемой области. В начале работы алгоритма размер рассматриваемой области равен размеру изображения. В процессе работы алгоритма размер рассматриваемой области уменьшается. Если ширина и высота рассматриваемой области равны единице, то необходимо закончить вейвлет-преобразование.

На каждой итерации вейвлет-преобразование применяется ко всем строкам рассматриваемой области изображения, затем ко всем столбцам, ширина и высота рассматриваемой области уменьшаются в два раза. Для выполнения вейвлет-преобразования необходимо составить матрицу вейвлет-преобразования в соответствии с размером рассматриваемой области. Так как входное изображение может иметь не только квадратную, но и прямоугольную форму, то ширина и высота рассматриваемой области могут отличаться, матрицу вейвлет-преобразования необходимо составлять отдельно для преобразования строк и столбцов.

Прямое вейвлет-преобразование может быть реализовано как рекурсивно, так и итеративно. Подробная схема разработанного алгоритма прямого вейвлет-преобразования представлена на рисунке 1.

Обратное вейвлет-преобразование реализуется применением тех же действий в обратном порядке.

Результат работы разработанного алгоритма представлен на рисунке 2.

Удаление части информации об изображении после применения вейвлет-преобразования

После применения прямого вейвлетного преобразования необходимо обеспечить сжатие путем обнуления части коэффициентов. Степень сжатия задается пользователем в процентах. Из ненулевых элементов необходимо выбрать указанный пользователем процент максимальных элементов. Это значит, что необходимо найти указанную пользователем порядковую статистику в наборе графических данных. Изображения содержат несколько каналов, массив графических данных может содержать по несколько тысяч элементов. Поэтому необходимо производить эффективный поиск порядковой статистики, например с помощью алгоритма поиска порядковой статистики `Randomized_Select` [3].

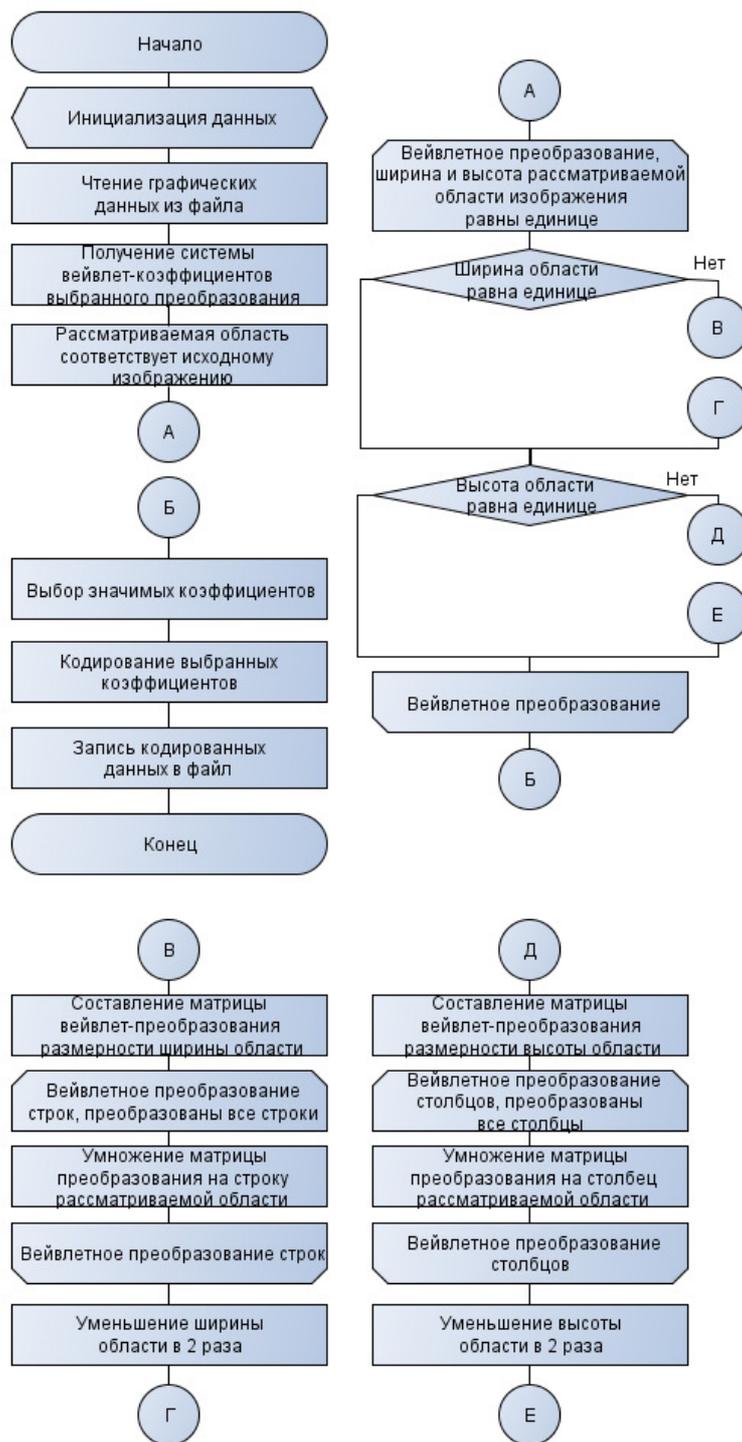


Рис. 1. Схема разработанного алгоритма прямого вейвлет-преобразования

Для работы прямого и обратного вейвлет-преобразований важна скорость работы. Так как вейвлет-преобразование состоит в умножении строк матрицы графических данных на матрицу преобразования, эту операцию можно распределить по потокам. Для распределения операций по потокам использовался открытый стандарт для распараллеливания программ на языке C++ OpenMP. Наличие инструментов OpenMP в среде разработки Visual Studio 2010 также упрощает ее использование. Стандарт OpenMP <http://sntbul.bmstu.ru/doc/649788.html>

является свободно распространяемым, имеет обширную документацию, постоянно обновляется, что является его преимуществами перед другими аналогичными инструментами.



Рис. 2. Вейвлет-преобразование изображения «Lena» с сохранением 10% (вверху справа), 5% (внизу слева), 2% (внизу справа) коэффициентов. Вверху слева представлено исходное изображение

Время работы разработанного алгоритма

Разработанный алгоритм состоит из трех основных частей: прямое вейвлет-преобразование, обратное вейвлет-преобразование, выбор значимых коэффициентов. Эксперименты по определению времени работы проводились для изображений с размерами 64x64, 128x128, 256x256, 512x512, 1024x1204. График зависимости времени работы от размера изображения прямого вейвлет-преобразования представлен на рисунке 3. На горизонтальной оси pixels – размер изображения в тысячах пикселей, на вертикальной оси seconds – время работы алгоритма прямого вейвлет-преобразования в секундах.

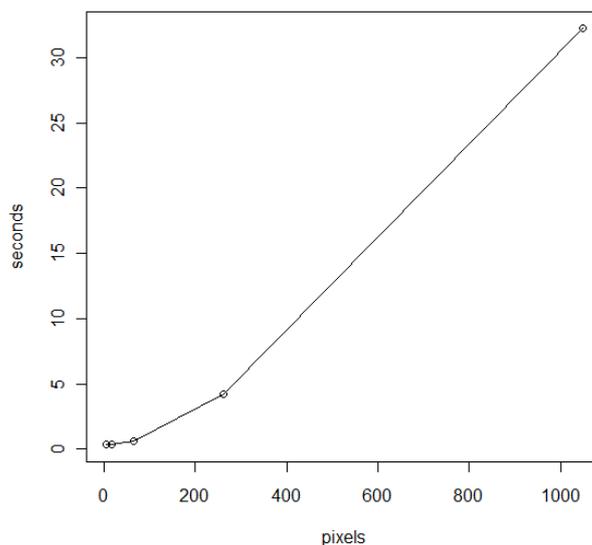


Рис. 3. График зависимости времени работы алгоритма прямого вейвлет-преобразования от размера изображения. На рисунке горизонтальная ось pixels – размер изображения в тысячах пикселей, вертикальная ось seconds – время работы алгоритма в секундах

График зависимости времени работы от размера изображения обратного вейвлет-преобразования представлен на рисунке 4. На горизонтальной оси pixels – размер изображения в тысячах пикселей, на вертикальной оси seconds – время работы алгоритма обратного вейвлет-преобразования в секундах.

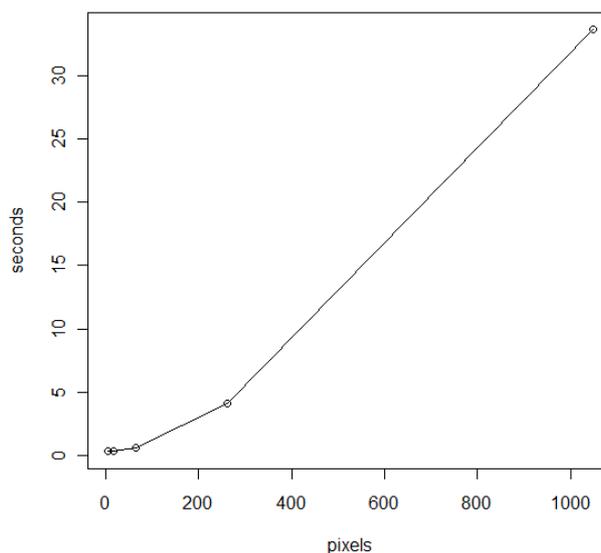


Рис. 4. График зависимости времени работы алгоритма обратного вейвлет-преобразования от размера изображения. На рисунке горизонтальная ось pixels – размер изображения в тысячах пикселей, вертикальная ось seconds – время работы алгоритма в секундах

Как видно из представленных на рисунках 3 и 4 графиках, время работы алгоритмов прямого и обратного вейвлет-преобразования изображений размерами от 512x512 производится за время от 15 секунд, что не применимо для сети Интернет, а так же для коммерческих баз данных. Разработанные алгоритмы требуют дальнейшей оптимизации.

График зависимости времени работы алгоритма выбора значимых коэффициентов представлен на рисунке 5. Как видно из графика, время работы алгоритма линейно зависит от размера данных.

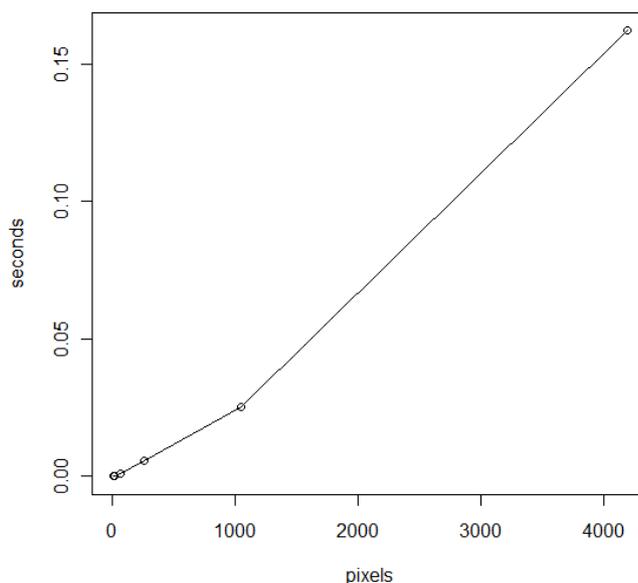


Рис. 5. График зависимости времени работы алгоритма выбора значимых коэффициентов от размера изображения. На рисунке горизонтальная ось pixels – размер изображения в тысячах пикселей, вертикальная ось seconds – время работы алгоритма в секундах

Расчеты проводились на персональном компьютере со следующими характеристиками:

- Процессор Intel Core 2 Duo CPU T5800 2.00GHz
- Оперативная память 2.00 GB

Список литературы

1. Фракталы и вейвлеты для сжатия изображений в действии [Текст] = Fractal and wavelet image compression techniques : учебное пособие / С. Уэлстид. – М. : Издательство Триумф, 2003. – 320с. ил. – ISBN 5-89392-079-1. – ISBN 0-8194-3503-1 (USA)

2. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео [Текст] : учебно-справочное издание / Д. Ватолин, А. Ратушняк, М. Смирнов, В. Юкин. – М. : ДИАЛОГ-МИФИ, 2003. – 384с. ил. – ISBN 5-86404-170-х
3. Алгоритмы: построение и анализ [Текст] = Introduction to algorithms / Т. Кормен, Ч. Лейзерсон, Р. Ривест, К. Штайн. – М. : Издательский дом «Вильямс», 2005. – 1296с. ил. – ISBN 5-8459-0857-4 (рус.)