

УДК 681.3

ОЦЕНКА ЭФФЕКТИВНОСТИ ИСПОЛЬЗОВАНИЯ ВЕЙВЛЕТОВ ДЛЯ КОДИРОВАНИЯ ИЗОБРАЖЕНИЙ

Вовк О.Л., Енин Д.С.
Кафедра ПМИ, ДонНТУ
vovkolga@ukrtop.com

Стаття присвячена проблемі стиснення зображень та використанню вейвлетів для розв'язання цієї проблеми. Проводиться аналіз існуючих технологій для кодування зображень та аналізуються ті переваги, які надає технологія вейвлет-аналізу у порівнянні з іншими існуючими методами.

Введение

Современный уровень развития технических средств позволяет пользователям работать с большими объемами визуальной информации. Для оперативной обработки и хранения таких цифровых данных используются различные алгоритмы и технологии сжатия. В частности, сжатие изображений необходимо как для хранения больших архивов изображений, так и для передачи больших объемов через линии связи с ограниченной пропускной способностью, или же обычным пользователям для экономии пространства жестких носителей информации [1].

При использовании различных алгоритмов сжатия основной проблемой является “потеря информации”. Рассматриваемый термин в данном случае означает, что восстановленное изображение не будет точно совпадать с исходным (кодируемым) изображением. В настоящее время, большинство изображений с непрерывным цветовым тоном кодируются с помощью алгоритма JPEG [2]. Рассматриваемая в данной статье технология сжатия при помощи вейвлетов, позволяет эффективно сжимать изображения с минимальной потерей качества, которая практически не заметна для человеческих глаз. И, в отличие от технологии JPEG, предлагаемая технология удовлетворяет как критерию минимизации занимаемого пространства, так и критерию максимизации качества восстанавливаемого изображения.

Цель данной статьи – обосновать целесообразность и оценить эффективность использования вейвлетов для кодирования визуальной информации.

В рамках данной статьи рассматриваются различные виды вейвлетов, анализируются различные технологии сжатия изображений и предлагается формат изображений на основе вейвлетов.

1 Определение вейвлетов и их виды

Понятие “вейвлет” появилось сравнительно недавно, его ввели Гроссман и Морле в середине 80-х годов в связи с анализом свойств сейсмических и акустических сигналов [3].

Вейвлет-преобразование сигнала состоит в его разложении по базису, сконструированному из обладающих определенными свойствами функций (вейвлетов), посредством масштабных изменений и переносов. Каждая из функций этого базиса характеризует как пространственную (временную) частоту, так и её локализацию в физическом пространстве (времени) [3].

Необходимо отметить, что вейвлет-преобразования являются усовершенствованным аналогом преобразований Фурье. Подробное описание вейвлетов приведено в [2-4].

Наиболее распространенными видами вейвлетов являются следующие (расположим их в порядке возрастания сложности их задания) [1,2,5]:

- вейвлеты Хаара. Данные вейвлеты являются “обучающим примером” ортонормальных базисов, но обладают многими недостатками, которые мешают их широкому применению: это плохие аналитические свойства с резким перепадом на границах интервалов (т.е. их плохая гладкость); они не ортогональны никаким полиномам, кроме тривиальной константы; также не обладают достаточно хорошей локализацией по частоте.
- семейство вейвлетов Добеши. Область задания этих вейвлетов шире, чем у вейвлетов Хаара. Для достаточно гладких функций вейвлет-коэффициенты Добеши намного меньше вейвлет-коэффициентов Хаара (т.е. исходный сигнал может быть сжат намного сильнее при помощи вейвлетов Добеши).
- Койфлеты. Этот тип вейвлетов был построен И. Добеши на основе идей Р. Койфмана (в честь которого они и были названы). Койфлеты обладают большим набором нулевых моментов. Они более симметричны. Разброс по гладкости у них больше, чем у вейвлетов Добеши. К недостаткам данных вейвлетов можно отнести

длину области задания, которая в 3 раза больше чем в вейвлетов Хаара.

- биортогональные вейвлеты используются, чтобы обеспечить полную симметрию и точную реконструкцию. При этом используются два дуальных вейвлет-базиса, связанные с двумя разными многомасштабными лестницами. В отличие от вейвлетов Добеши, у которых регулярность тесно связана с числом нулевых моментов, биортогональные вейвлеты обладают большей свободой выбора. Если один из них обладает гладкостью порядка r , то дуальный ему вейвлет автоматически имеет, по крайней мере, r нулевых моментов. Большое число нулевых моментов приводит к лучшим результатам при сжатии информации для достаточно гладкой функции f . Симметричные ортогональные вейвлеты, близкие к ортонормальному базису, похожи на койфлеты.
- сплайны – вейвлеты с некомпактными носителями, но с экспоненциальным спадом на бесконечности и с некоторым числом непрерывных производных.
- непрерывные вейвлеты. Одним из специальных и часто используемых примеров непрерывных вейвлетов является вторая производная гауссовой функции, известная под названием “мексиканская шляпа” (МНАТ) из-за ее формы.

2 Постановка задачи сжатия (кодирования) изображений

В течение последних 10 лет в рамках компьютерной графики бурно развивается совершенно новая область – алгоритмы архивации изображений. Появление этой области обусловлено тем, что изображения – это своеобразный тип данных, характеризующийся тремя особенностями [6]:

1. Изображения (как и видео) занимают намного больше места в памяти, чем текст. Так, скромная, не очень качественная иллюстрация на обложке книги размером 500x800 точек, занимает 1.2 Мб – столько же, сколько художественная книга из 400 страниц (60 знаков в строке, 42 строки на странице). Эта особенность изображений определяет актуальность алгоритмов архивации графики.
2. Второй характеристикой изображений является то, что человеческое зрение при анализе изображения оперирует контурами, общим переходом цветов, и, сравнительно нечувствительно к малым изменениям в изображении. Таким образом, можно создать эффективные алгоритмы архивации изображений, в которых декомпрессированное изображение не

будет совпадать с оригиналом, однако человек этого не заметит. Данная особенность человеческого зрения позволила создать специальные алгоритмы сжатия, ориентированные только на изображения.

3. Можно легко заметить, что изображение, в отличие, например, от текста, обладает избыточностью в 2-х измерениях. Т.е. как правило, соседние точки, как по горизонтали, так и по вертикали, в изображении близки по цвету. Кроме того, мы можем использовать подобие между цветовыми плоскостями R, G и B в наших алгоритмах, что дает возможность создать еще более эффективные алгоритмы. Т.е., при создании алгоритма компрессии графики мы используем особенности структуры изображения.

На данный момент известно несколько семейств алгоритмов, которые разработаны исключительно для сжатия изображений, и применяемые в них методы практически невозможно применить к архивации других видов данных. Основные три семейства алгоритмов сжатия следующие [13]:

- RLE-семейство. Данные алгоритмы необычайно просты в реализации. Групповое кодирование (от английского Run Length Encoding (RLE)) – один из самых старых и самых простых алгоритмов архивации графики. Сжимаемое изображение (как и в семействе, описанном ниже) вытягивается в цепочку байт по строкам раstra. Само сжатие в RLE происходит за счет того, что в исходном изображении встречаются цепочки одинаковых байт. Замена их на пары <счетчик повторений, значение> уменьшает избыточность данных.
- LZW-семейство. Название рассматриваемая группа алгоритмов получила по первым буквам фамилий его разработчиков – Lempel, Ziv и Welch. Сжатие в нем, в отличие от RLE, осуществляется уже за счет сокращения одинаковых цепочек байт. Более подробно данное семейство описано в следующем пункте.
- JPEG-семейство. Алгоритмы оперирует областями 8x8, на которых яркость и цвет меняются сравнительно плавно. Вследствие этого, при разложении матрицы значимыми оказываются только первые коэффициенты. Таким образом, сжатие в JPEG осуществляется за счет плавности изменения цветов в изображении. В целом алгоритм основан на дискретном косинусоидальном преобразовании (в дальнейшем ДКП), применяемом к матрице изображения для получения некоторой новой матрицы коэффициентов. Для восстановления исходного изображения применяется обратное преобразование.

3 Анализ основных форматов изображений

С каждым годом появляется всё более совершенная графическая аппаратура и более сложные графические приложения. Это предъявляет жесткие требования к форматам хранения графической информации. За прошедшие два десятилетия родились и умерли сотни форматов. Среди этого разнообразия стоит выделить форматы JPEG, GIF и BMP и производные от них [7-9].

Формат BMP – основной формат Windows. Это несжатый формат, хотя для него предусмотрено сжатие простым байт-ориентированным алгоритмом, которое более развито в производном от него формате PCX. Формат BMP легко конвертируется с другие форматы, без каких либо промежуточных форм, если для него не было применено сжатие. Файл BMP может содержать изображения, размером 232x232 пикселей и глубину цвета от 1 до 24 бит.

Формат файлов GIF – это стандарт фирмы CompuServe для определения растровых цветных изображений. Этот формат позволяет высвечивать на различном оборудовании графические высококачественные изображения с большим разрешением и подразумевает механизм обмена и высвечивания изображений. Данные файлов GIF хранятся в упакованном виде сжатым алгоритмом LZW (Lempel-Ziv Welch). LZW – это способ сжатия данных, который извлекает преимущества при повторяющихся цепочках данных. Поскольку растровые данные обычно содержат довольно много таких повторений, LZW является хорошим методом для их сжатия и раскрытия. Алгоритм LZW преобразует серию значений данных в серию кодов, которые могут быть самими значениями или кодами, описывающими серию значений. Если использовать аналогию с текстовыми символами, то выходные коды состоят из символов и кодов, которые описывают цепочки символов. В отличие от большинства форматов, формат GIF позволяет хранить в одном файле несколько изображений, что позволяет использовать данные файлы как анимационные.

Более прогрессивный алгоритм JPEG способен уменьшать объём графической информации в десятки раз. Высокая эффективность сжатия, которую дает этот алгоритм, основана на том факте, что в матрице частотных коэффициентов, образующейся из исходной матрицы после дискретного косинусного преобразования, низкочастотные компоненты расположены ближе к левому верхнему углу, а высокочастотные – внизу справа этой матрицы. Это важно потому, что большинство графических образов на экране компьютера

состоит из низкочастотной информации, так что высокочастотные компоненты матрицы можно безболезненно выбросить. “Выбрасывание” выполняется путем округления частотных коэффициентов. После округления отличные от нуля значения низкочастотных компонент остаются, главным образом, в левом верхнем углу матрицы. Округленная матрица значений кодируется с учетом повторов нулей. В результате графический образ сжимается более чем на 90%, теряя очень немного в качестве изображения только на этапе округления. При достаточно большом сжатии может ухудшиться качество изображения.

4 Кодирование изображений с помощью вейвлетов

Поскольку растровое изображение является двумерным дискретным сигналом, то к нему применимы двумерные дискретные вейвлет-преобразования [10].

Сокращенно, самый простой подход к сжатию изображений при помощи вейвлет-анализа состоит в следующем:

- выполнить вейвлет-преобразование матрицы цветовых компонент изображения;
- упорядочить коэффициенты;
- отбросить “хвост” упорядоченного массива, энергия которого равна допустимой (по условиям задачи) величине;
- запомнить сохраненные коэффициенты и их положение в массиве исходных коэффициентов;
- при восстановлении заменять отброшенные коэффициенты нулями.

Эта идея в той или иной форме присутствует во всех методах вейвлетного сжатия. Данную процедуру можно, например, применять отдельно к каждому из квадрантов, полученных при разложении по оптимально выбранным вейвлет-пакетам. Однако существуют и более совершенные методы, где кодирование сразу идет на уровне битов. Он является модификацией метода погруженного нуля-дерева (embedded zero-tree) и подробно описан в [11].

5 Сравнительный анализ технологий кодирования изображений

Прежде чем проводить какой-либо сравнительный анализ различных алгоритмов, технологий следует задать степень важности требований, предъявляемых к алгоритму [7, 12]:

1. Высокая степень компрессии. Некоторые алгоритмы дают лучшее соотношение качества к размеру файла при высоких степенях

компрессии, однако, проигрывают другим алгоритмам при низких степенях.

2. Высокое качество изображений. Выполнение этого требования напрямую противоречит выполнению предыдущего требования.
3. Высокая скорость компрессии. Это требование для некоторых алгоритмов с потерей информации является взаимоисключающим с первыми двумя. Интуитивно понятно, что чем больше времени нужно для анализа изображения, чтобы получить наивысшую степень компрессии, тем лучше будет результат.
4. Высокая скорость декомпрессии. Достаточно универсальное требование, актуальное для многих приложений. Однако существуют примеры приложений, где время декомпрессии далеко не критично.
5. Масштабирование изображений. Данное требование подразумевает легкость изменения размеров изображения до размеров окна активного приложения.
6. Возможность показать “огрубленное” изображение (низкого разрешения), используя только начало файла. Данная возможность актуальна для различного рода сетевых приложений, где перекачивание изображений может занять достаточно большое количество времени, и желательно, получив начало файла, предоставить возможность предварительного просмотра.
7. Устойчивость к ошибкам. Данное требование означает локальность нарушений в изображении при порче или потере фрагмента передаваемого файла.
8. Редактируемость. Под редактируемостью понимается минимальная степень ухудшения качества изображения при его повторном сохранении после редактирования. Многие алгоритмы с потерей информации могут существенно испортить изображение за несколько итераций редактирования.

Характеристики алгоритма относительно некоторых требований, сформулированные выше, зависят от конкретных условий, в которые будет поставлен алгоритм. Так, степень компрессии зависит от того, на каком классе изображений алгоритм тестируется. Аналогично, скорость компрессии нередко зависит от того, на какой платформе реализован алгоритм. Преимущество одному алгоритму перед другим может дать, например, возможность использования в вычислениях алгоритма технологий нижнего уровня, типа MMX, а это возможно далеко не для всех алгоритмов. Так, JPEG существенно выигрывает от применения технологии MMX, а LZW

нет. Кроме того, следует учитывать, что некоторые алгоритмы распараллеливаются легко, а некоторые нет.

Исходя из сделанных выше замечаний, выделим несколько наиболее важных критериев сравнения алгоритмов компрессии [7]:

- “Худший”, “средний” и “лучший” коэффициенты сжатия. “Худший” – доля, на которую возрастет изображение, если исходные данные будут наихудшими. “Средний” – некий среднестатистический коэффициент для того класса изображений, на который ориентирован алгоритм. “Лучший” коэффициент используется лишь теоретически, поскольку показывает степень сжатия наилучшего (как правило, абсолютно черного) изображения, иногда фиксированного размера.
- Класс изображений, на который ориентирован алгоритм. Иногда на других классах изображений получаются худшие результаты.
- Симметричность. Отношение характеристики алгоритма кодирования к аналогичной характеристике при декодировании. Характеризует ресурсоемкость процессов кодирования и декодирования. Для нас наиболее важной является симметричность по времени: отношение времени кодирования ко времени декодирования. Иногда выдвигается требование симметричности по памяти.
- Потеря качества. Коэффициент архивации. Дело в том, что у большинства алгоритмов сжатия с потерей информации существует возможность изменения коэффициента сжатия.
- Характерные особенности алгоритма и изображений, к которым его применяют. Здесь могут указываться наиболее важные для алгоритма свойства, которые могут стать определяющими при его выборе.

В соответствии с поставленными требованиями и выделенными критериями проведем сравнительный анализ кодирования изображений при помощи вейвлетов (или волнового (рекурсивного) алгоритма) в сравнении со следующими алгоритмами RLE, LZW, JPEG (таблица 1).

Таблица 1

Алгоритм	Особенности изображения, за счет которых происходит сжатие
RLE	Подряд идущие одинаковые цвета: 2 2 2 2 2 15 15 15
LZW	Одинаковые подцепочки: 2 3 15 40 2 3 15 40
JPEG	Отсутствие резких границ
Волновой	Плавные переходы цветов и отсутствие резких границ

Алгоритм	Коэфф. сжатия	Симметричность по времени	Ориентация	Потери	Размерность
RLE	32, 2, 0.5	1	3,4-х битные	Нет	1D
LZW	1000, 4, 5/7	1.2-3	1-8 битные	Нет	1D
Lossless JPEG	2 раза	~1	24-битные, серые	Нет	2D
JPEG	2-20 раз	~1	24-битные, серые	Да	2D
Волновой	2-200 раз	1.5	24-битные, серые	Да	2D

Из таблицы 1 видно, что алгоритм вейвлет-сжатия выигрывает по многим критериям у большинства других алгоритмов: как по времени декодирования, так и по степени сжатия.

Заключение

В ходе данного исследования было установлено, что уникальные математические свойства вейвлетов сделали их очень мощным инструментом анализа и последующего синтеза любого сигнала:

- свойство ортогональности позволяет получать независимую информацию на разных масштабах;
- нормируемость обеспечивает сохранение величины информации на различных этапах преобразования;
- свойство локальности помогает получить знание о тех конкретных областях, в которых проявляют себя изучаемые масштабы (частоты);
- наконец, полнота вейвлет-базиса, образованного сжатиями и сдвигом некоторой функции, обеспечивает возможность совершить обратное преобразование.

Вейвлет-преобразование легко обобщается на множества любых размерностей и потому может применяться также и для анализа многомерных объектов. Благодаря этому вейвлеты незаменимы при обработке изображений, что и было продемонстрировано в данной работе.

Использование вейвлетов для сжатия изображений – это лишь малая часть всех возможностей, которые предоставляют вейвлеты, поэтому среди дальнейших направлений исследований можно

выделить применение вейвлетов для полноценной обработки изображений: масштабирование, коррекция четкости, выделение перепадов, локальная обработка и т.д.; а также объединение различных типов вейвлетов в так называемые вейвлет-пакеты для создания более мощных средств расширяющих круг возможностей и способствующих более качественной обработке изображений.

Литература

1. Дремин И.М., Иванов О.В., Нечитайло В.А., Вейвлеты и их использование, Успехи физических наук, т. 171, №5, Май 2001, стр. 465–501.
2. Walker J.S., WAVELETS and their Scientific Applications – Chapman and Hall, 1999.
3. Астафьева Н.М., Вейвлет-анализ: основы теории и примеры применения, Успехи физических наук, т. 166, №11, Ноябрь 1996, стр. 1145–1170.
4. Переберин А.В., О систематизации вейвлет-преобразований, 2001г.
5. Дьяконов В.П., Вейвлеты. От теории к практике, - М: СОЛОН-Р, 2002, 448 стр.
6. Ватолин Д., Ратушняк А., Смирнов М., Юкин В. и др., Методы сжатия данных: сжатие изображений, 2001-2003.
7. Ватолин Д.С., Алгоритмы сжатия изображений, - М.: Диалог-МГУ, 1999.
8. Александров В.В., Горский Н.Д., Представление и обработка изображений: рекурсивный подход -Д.: Наука, 1985, 190 стр.
9. <http://graphics.cs.msu.su/library/>.
10. Valens C., Embedded Zerotree Wavelet Encoding, 1999.
11. Л. Левкович-Маслюк, А. Переберин, Введение в вейвлет-анализ, http://www.keldysh.ru/pages/gc98/cd/tutorial/leo_lev.
12. Kingsbury N., Image processing with complex wavelets, Phil. Trans. R. Soc. Lond. A, 1999.

Поступила в редакцию: 12.01.04