

ОЦЕНКА КАЧЕСТВА ЦИФРОВЫХ ИЗОБРАЖЕНИЙ, ПОЛУЧЕННЫХ В АЛГОРИТМЕ ИЕРАРХИЧЕСКОГО КОДИРОВАНИЯ

Волохов В.А., Новоселов С.А., Моисеев А.А., Приоров А.Л

Ярославский государственный университет им. П.Г. Демидова
150000, Россия, Ярославль, ул. Советская, 14. Тел. (0852) 79-77-75. deslab@uniyar.ac.ru

Цифровые изображения занимают все большую часть информационного пространства. Развитие Интернета, наряду с доступностью все более мощных компьютеров и прогрессом в технологии производства цифровых камер, сканеров и принтеров, привели к широкому использованию цифровых изображений. Отсюда постоянный интерес к улучшению алгоритмов сжатия данных, представляющих изображения. Сжатие данных важно как для скорости передачи, так и эффективности хранения.

В последнее время активно развиваются такие алгоритмы сжатия как JPEG (Joint Photographic Experts Group), методы фрактального кодирования, вейвлет-преобразования [1]. Данные методы различны по своей структуре, но идея всех трех состоит в одном: преобразовать данные изображения к другому виду, где легче отбросить часть данных, которую можно было бы безболезненно удалить. В случае преобразования Фурье, которое наряду с дискретным косинус-преобразованием (ДКП) является основой большинства методов сжатия JPEG, это обычно высокочастотные сигналы. Для вейвлет-преобразования это данные, соответствующие мелким деталям. Фрактальные методы пытаются непосредственно представить информацию об изображении в компактном виде.

Результат алгоритмического кодирования может быть затем сжат путем компактного представления закодированных чисел. Квантование может дать как уменьшение количества чисел в списке, так и уменьшения количества бит, необходимых для записи каждого числа.

При декодировании исходное изображение восстанавливается из закодированных данных. В случае методов преобразования (Фурье, ДКП, вейвлет-преобразования, методов фрактального кодирования) этап декодирования применяет обратное преобразование. Декодирование может сопровождаться дополнительной пост-обработкой, направленной на улучшение качества декодированного изображения, например, удаление артефактов блочности, которые могут появиться в результате работы алгоритма сжатия.

Работа посвящена рассмотрению нестандартизованного, но достаточно эффективного алгоритма SPIHT (Set Partition in Hierarchical Trees – “Разбиение множества в иерархические деревья”) [2], основанного на вейвлет-преобразовании и предназначенного для сжатия и прогрессивной передачи изображений. На основе упрощенной реализации указанного алгоритма, произведена оценка качества декодированного изображения в зависимости от количества переданной информации.

Двумерное вейвлет-преобразование, применяемое к изображениям, можно представить как последовательность вертикальных и горизонтальных высокочастотных и низкочастотных операторов, применяемых к изображению. Схемы на рис. 1 (слева) и показывают размещение результатов действия этих операторов в массиве вейвлет-преобразования, где LL – выделенная низкая частота по горизонтали и низкая по вертикали, LH – низкая по горизонтали и высокая по вертикали и так далее соответственно. Эти блоки могут быть структурированы в дерево, как показано на рис. 1 (справа). Каждый коэффициент в блоке имеет четыре “дочерних” коэффициента в соответствующих блоках следующего уровня. Смысл введения структуры дерева в том, что, как правило, коэффициенты в блоках на различных уровнях обладают значительной степенью подобия. То есть дочерние коэффициенты на некотором уровне, вероятно, будут похожими на своих “родителей”, находящихся на предыдущем уровне.

В частности, довольно распространен случай, что если коэффициент квантуется до нуля, то и его дочерние коэффициенты тоже квантуются до нуля. Это наблюдение легло в основу так называемого кодирования с нуль-деревом (wavelet zero tree encoding), впервые примененного Льюисом и Ноулесом. Однако, условие о квантовании дочерних коэффициентов до нуля, при квантовании их родительских коэффициентов до нуля, может быть верно не всегда. Эта возможность привносит ошибки в метод нуль-дерева Льюиса и Ноулеса. Шапиро предложил модифицированный метод кодирования с нуль-деревом, который учитывает условие изолированных нулей. В последствии, Сейд и Пилман предложили усовершенствованный вариант кодирования с нуль-деревом, названный “разбиение множества в иерархических деревьях” (SPIHT).

Обозначим пиксели исходного изображения P через $P[i,j]$. Любое множество фильтров T может быть использовано для преобразования пикселей в вейвлет-коэффициенты $C[i,j]$. Эти коэффициенты образуют некоторый образ C . Само преобразование обозначается $C = T(P)$. При прогрессивном методе передачи декодер вначале присваивает значение ноль реконструированному образу \tilde{C} . Затем он принимает преобразованные коэффициенты, декодирует их и использует для получения улучшенного образа \tilde{C} , который, в свою очередь, производит улучшенное изображение $\tilde{P} = T^{-1}(\tilde{C})$.

Основная цель прогрессивного метода состоит в скорейшей передаче самой важной части информации об изображении, которая дает самое большое сокращение расхождения исходного изображения и реконструированного образа. Для количественного измерения этого расхождения алгоритм SPIHT использует среднеквадратическую ошибку (СКО):

$$СКО = \frac{1}{N} \sum_i \sum_j (P[i, j] - \tilde{P}[i, j])^2, \text{ где } N - \text{общее число пикселей.}$$

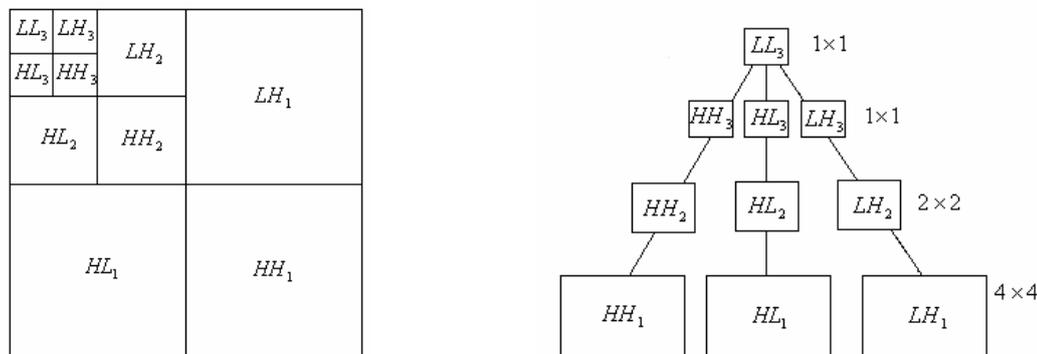


Рис. 1. Схема двумерного вейвлет-преобразования как последовательности прохождения по вертикали и горизонтали через высокочастотный фильтр (H) и низкочастотный фильтр (L) (слева). Блоки вейвлет-поддеревя, упорядоченные в структуру дерева, для изображения 8×8 (справа)

Два основных принципа алгоритма SPIHT, используемые при прогрессивной передаче изображения, состоят в том, что, во-первых, кодер должен посылать в первую очередь самые большие (по абсолютной величине) коэффициенты, а во-вторых, самые старшие биты (двоичного представления коэффициентов), так как они несут в себе информацию, которая больше всего сокращает расхождение СКО.

Основные шаги кодера алгоритма SPIHT следующие (предполагается, что коэффициенты отсортированы до начала цикла):

Шаг 1. Для заданного сжимаемого изображения вычислить его вейвлет-преобразование, используя подходящие вейвлет фильтры, разложить его на коэффициенты преобразования $C[i, j]$ и представить их в виде целых чисел фиксированной разрядности;

Шаг 2. Передача n_{\max} - количество разрядов (считая от нуля), занимаемых максимальным отсчетом разложения, то есть $n_{\max} = \log_2(\max_{i,j}(a_{i,j}))$ (округление вниз);

Шаг 3. Инициализация списка значимых коэффициентов – пустое множество;

Шаг 4. Сортировка. Передать число l коэффициентов $C[i, j]$, которые удовлетворяют неравенству $2^n \leq |C[i, j]| \leq 2^{n+1}$. Затем передать l пар координат и l знаков этих коэффициентов;

Шаг 5. Поправка. Передать $(n-1)$ -ые старшие биты всех коэффициентов, удовлетворяющих неравенству $|C[i, j]| \geq 2^n$. Эти коэффициенты были выбраны на шаге сортировки предыдущей итерации цикла;

Шаг 6. Итерация. Уменьшить n на 1. Если необходимо сделать еще одну итерацию, пойти на Шаг 4.

Обычно последняя итерация совершается при $n=0$, но кодер может остановиться раньше. В этом случае наименее важная часть информации (некоторые менее значимые биты всех вейвлет-коэффициентов) не будет передаваться. В этом заключается естественное отбрасывание информации в методе SPIHT. В альтернативе кодер передает весь образ (то есть, все биты всех вейвлет-коэффициентов), а декодер может остановить процесс декодирования в любой момент, когда восстанавливаемое изображение достигло требуемого качества. Это качество и предопределяется пользователем, или устанавливается декодером автоматически.

Описанный выше алгоритм очень прост, так как в нем предполагается, что коэффициенты были отсортированы (упорядочены) до начала цикла. В принципе изображение может состоять из очень большого числа пикселей, в нем может быть более миллиона коэффициентов, их сортировка может оказаться весьма медленной процедурой.

Алгоритм, используемый методом SPIHT, основан на том, что нет необходимости сортировать все коэффициенты. Главной задачей этапа сортировки на каждой итерации является выявление коэффициентов удовлетворяющих неравенству $2^n \leq |C[i, j]| \leq 2^{n+1}$.

Поскольку результат каждого теста записывается в сжатый файл, то хорошо бы минимизировать число необходимых тестов. Для достижения этой цели было предложено использовать специальную структуру

данных – нуль-дерево (введенное выше), усовершенствованная модификация которого используется в алгоритме SPIHT, и получила название пространственно ориентированного дерева.

Кодирование коэффициентов заключенных в структуру пространственно ориентированного дерева осуществляется за счет трех списковых субструктур *LIS*, *LIP*, *LSP*:

1. *LIS* – список не значащих множеств (list of insignificant sets);
2. *LIP* – список не значащих точек (list of insignificant pixels);
3. *LSP* – список значащих точек (list of significant pixels).

Последовательный анализ указанных структур выявляет значимые и не значимые коэффициенты на каждой итерации.

В качестве практической реализации SPIHT в работе рассматривается упрощенная версия данного алгоритма (без энтропийного кодирования и матрицы значимости).

Для того чтобы оценить, как изменяется качество восстановленного изображения (для данного алгоритма) с изменением числа битовых плоскостей, можно воспользоваться величиной, получившей название, пиковое отношение сигнал шум (*PSNR*) (Peak Signal-to-Noise Ratio), измеряемой в децибелах.

$$PSNR = 20 \log_{10} \frac{\max_{i,j} |P[i, j]|}{\sqrt{\widehat{NEI}}}, \text{ где СКО - среднеквадратическая ошибка, } \max_{i,j} |P[i, j]| -$$

максимальный пиксель исходного изображения.

Для рассмотренного в работе изображение Lena полученные графические зависимости для трех типов вейвлетов (Хаара(Haar), Добеши.4 (Db4), Добеши.6 (Db6)), используемых в алгоритме, изображены на рис. 2.

Как видно из этих графиков с ростом количества битовых плоскостей, с помощью которых представляется изображение, качество восстановленного изображения при декодировании растет, что и наблюдается визуально. Помимо этого из графика видно, что вейвлет Хаара дает лучше качество восстановленного изображения, чем Db4 с первой по девятую битовые плоскости, а потом наоборот. Что касается Db6, то график зависимости *PSNR* (bit plane) с первой по девятую ведет себя практически так же, как и Db4, а при дальнейшем росте числа битовых плоскостей *PSNR* чуть лучше, чем для вейвлета Хаара. Объяснить тот факт, что Db4 и Db6 дают лучшее восстановление исходного изображения, чем вейвлет Хаара, при использовании полного количества битовых плоскостей, можно тем, что Db4 и Db6 обладают большей гладкостью (большим числом нулевых моментов, существующих производных), поэтому, обрабатывают низкочастотную (детализирующую) составляющую изображения лучше, чем вейвлет Хаара, или, как говорят, Db4 и Db6 лучше концентрируют энергию образа.

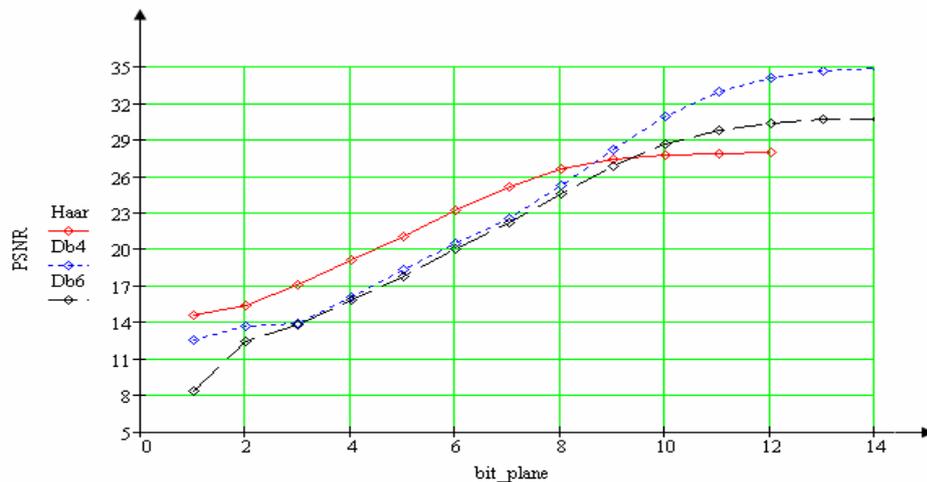


Рис. 2. Зависимость *PSNR* от битовой плоскости для вейвлетов Хаара (Haar), Добеши.4(Db4), Добеши.6 (Db6)

В качестве примера на рис. 3 представлены декодированные изображения при сохранении десяти битовых плоскостей.



Рис. 3. Изображения, полученные в результате декодирования при использовании десяти битовых плоскостей (Db2, Db4, Db6 соответственно)

Литература

1. Daubechies I. Ten Lectures on Wavelets. SIAM, Philadelphia, PA, 1992.
2. Said A. and Pearlman W. A new, fast, and efficient image codec based on set partitioning in hierarchical Trees // IEEE Trans. on Circ. and Syst. for Video Tech. 1996. V. 6, P. 243-250.

ESTIMATION OF QUALITY OF IMAGE RECEIVED ON THE BASE OF HIERARCHICAL CODING ALGORITHM

Volohov V., Novoselov S., Moiseev A., Priorov A.

Yaroslavl State University

14 Sovetskaya st., Yaroslavl, Russia 150000. Phone: 7-4852-797775. dcslab@uniyar.ac.ru

Digital images occupy more and more part of informational space. The development of Internet with the accessibility of the more powerful computers and with progress in technology of producing digital cameras, scanners and printers lead to the wide using of digital images. Hence the permanent interest to improvement of algorithms of compressing data, which are presented images. Compressing data is important as for the speed of transfer as for effectiveness storage.

At last time image compressing algorithms such as JPEG (Joint Photographic Experts Group), methods of fractal coding, wavelet-transform are developing very active. These methods are different for its structure, but the idea consist in following: to transfer image to another form, where is easy to reject a part of data, which can be removed smooth. In the case of Fourier transform, which with the discrete cosine transform (DCT) are the base of the majority methods of compressing JPEG, it's a high frequency signals. Fractals methods tries directly presented information about image in a compact form.

Then results of coding algorithm can be compressed by the compact representation of coding numbers. Quantization can give as the decrease quantity of bit, which are necessary for writing down every numbers.

While decoding the initial image are restored from the coded data. In the case of the method of transform (Fourier, DCT, wavelet-transform, methods of fractal coding) the part of decoding apply reverse transform. Decoding can be accompany by supplementary post-treatment, which are directed on improvement quality of decoded image, for example, removing artifact of pulley, that can appear as the result of the work of algorithm of compression.

This work is devoted to algorithm SPIHT (Set Partition in Hierarchical Trees) [1], which are not standard, but it's very effective. This algorithm is based on the wavelet-transform and intend for the compression and progressive transfer of image. On the base of simplify model of this algorithm, the estimation of image quality of transfer information.

To estimate how image quality depends from numbers of bit plane value of PSNR (Peak Signal-to-Noise Ratio) can be used.

$PSNR = 20 \log_{10} \frac{\max_{i,j} |P[i, j]|}{\sqrt{MSE}}$, where MSE – mean square error, $\max_{i,j} |P[i, j]|$ – maximal pixel of initial image.

References

1. Said A. and Pearlman W. A new, fast, and efficient image codec based on set partitioning in hierarchical Trees // IEEE Trans. on Circ. and Syst. for Video Tech. 1996. V. 6, P. 243-250.