

Оптимизация алгоритма сжатия изображений JPEG-2000 с помощью подбора длины R-D кривых.

Сокол А.В. (sokol_glaz@mail.ru)

НПО «Лептон»

Алгоритм сжатия изображений JPEG-2000 [1,2] на сегодняшний день считается одним из лучших и по эффективности сжатия заметно превосходит общеизвестный метод JPEG. Этот алгоритм основан на вейвлет-преобразовании исходного изображения и работе с полученными вейвлет-коэффициентами, которые сжимаются значительно лучше, чем пиксели исходного изображения. Вместе с тем как вейвлет-преобразование, так и обработка коэффициентов достаточно сложны и требуют большой вычислительной мощности, и поэтому JPEG-2000 пока применяют лишь в тех случаях, когда требуется получить как можно лучшее качество сжатого изображения, а время выполнения не критично. В связи с этим большой интерес представляют способы, позволяющие повысить временные характеристики преобразований.

В данной статье приводится описание одного такого способа.

Коротко суть алгоритма JPEG-2000 состоит в следующем. К исходному изображению применяется вейвлет-преобразование с заданным числом уровней. Дискретное вейвлет-преобразование – это фильтрация строки (или столбца) исходного изображения двумя вейвлет-фильтрами. Это КИХ-фильтры, один из которых – низкочастотный фильтр L , а другой – высокочастотный фильтр H . Поскольку после фильтрации частотный диапазон сокращается в два раза, профильтрованные данные, согласно теореме Котельникова, можно проредить вдвое, т.е. убрать, допустим, все четные коэффициенты. Затем из полученных компонент составляется новая строка, как показано на Рис1.

$$\begin{array}{c}
 \mathbf{x} \\
 \boxed{x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad x_6} \\
 l_i = \sum_j L_j x_{j-i} \quad h_i = \sum_j H_j x_{j-i} \\
 \boxed{l_1 \quad l_2 \quad l_3} \quad \boxed{h_1 \quad h_2 \quad h_3} \\
 \begin{array}{cc}
 L & H \\
 \boxed{l_1 \quad l_2 \quad l_3} & \boxed{h_1 \quad h_2 \quad h_3}
 \end{array}
 \end{array}$$

Рис.1. Вейвлет-преобразование одномерного сигнала.

Для двумерного сигнала существует отдельное двумерное вейвлет-преобразование. Но его достаточно сложно реализовать, и на практике его обычно заменяют одномерным. Такая замена не сильно проигрывает в качестве сжатия, но при этом существенно упрощает фильтрацию. В одномерном случае сначала фильтруют строки, потом столбцы. Изображение при этом распадается на 4 части (рис. 2).

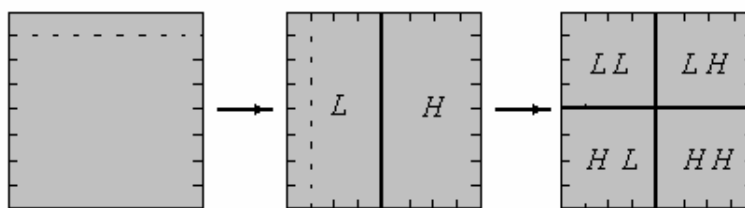


Рис. 2. Фильтрация двумерного сигнала – изображения.

Затем к части, которая профильтрована только низкочастотными фильтрами применяют ту же самую процедуру. Такой алгоритм носит название алгоритма Малла (рис.3).

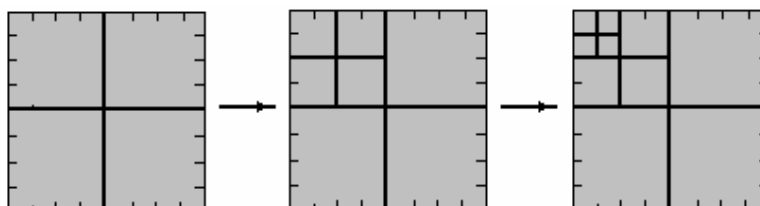


Рис.3. Алгоритм Малла.

Результат вейвлет-преобразования будет выглядеть таким образом (рис. 4):

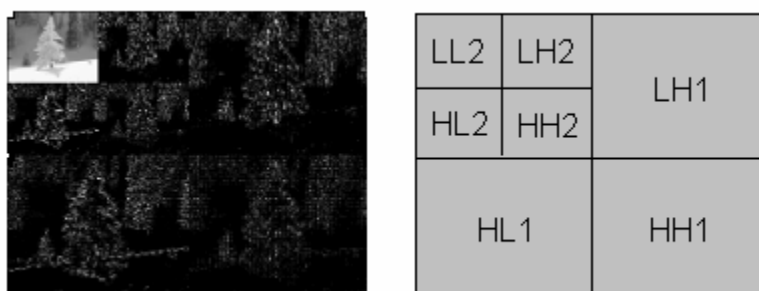


Рис. 4. Результат вейвлет-преобразования.

Здесь показано двухуровневое вейвлет-преобразование. Часть в верхнем левом углу профильтрована только низкочастотными фильтрами и представляет собой сглаженную уменьшенную копию оригинала. Вся остальная область профильтрована как низкочастотными, так и высокочастотными фильтрами. На ней заметны обширные участки близких к нулю коэффициентов и отдельные яркие точки, которые означают резкие перепады яркости на исходном изображении. Эта часть очень хорошо подходит для сжатия даже традиционными алгоритмами. Собственно, получение большого числа нулевых коэффициентов и является основной целью вейвлет-преобразования, само же преобразование сжатия не дает.

Опытным путем установлено, что оптимальным с точки зрения критерия “качество – сложность реализации” является двухуровневое вейвлет-преобразование. Значительное увеличение числа уровней не приводит к заметному повышению эффективности сжатия, а сложность алгоритма существенно возрастает.

Далее применяется последовательное кодирование всех частотных блоков LL2, LH2 и т. д. Кодирование осуществляется арифметическим методом с помощью вероятностной модели, и проводится в следующей последовательности:

- 1) определяется разрядность коэффициентов блока по формуле

$$N_{\max} = \text{ceil}(\log_2(C_{\max}))$$
 где C_{\max} -- это максимальный по модулю коэффициент.
- 2) у всех коэффициентов кодируются биты на месте N_{\max} , затем $N_{\max} - 1$ и т.д.

Такой способ называется кодированием этажей битов (bit plane), в алгоритме JPEG-2000 каждый этаж битов кодируется в три стадии, которые называются significance pass, refinement pass, cleanup pass. Вероятностная модель для арифметического кодирования строится на том факте, что большие и малые коэффициенты блока не перемешаны равномерно, а концентрируются в отдельных областях, и это позволяет судить о значении кодируемого бита, рассматривая величину ближайших вейвлет-коэффициентов.

Отличительной чертой JPEG-2000 является наличие так называемого метода оптимизации длины кода блоков (optimized truncation). Как уже было сказано, биты последовательно кодируются от значения N_{\max} вплоть до младшего бита (LSB, least significant bit), причем закодированные данные сразу не посылаются в выходной поток, а запоминаются в памяти. После того, как закодированы все блоки, в памяти остается набор закодированных данных (для двухуровневого вейвлет-преобразования этот набор состоит из 7 элементов). Задача метода – оптимизировать длину элементов таким образом, чтобы обеспечить требуемую степень сжатия и одновременно так подобрать соотношение длин элементов, чтобы среднеквадратическая ошибка сжатия была наименьшей.

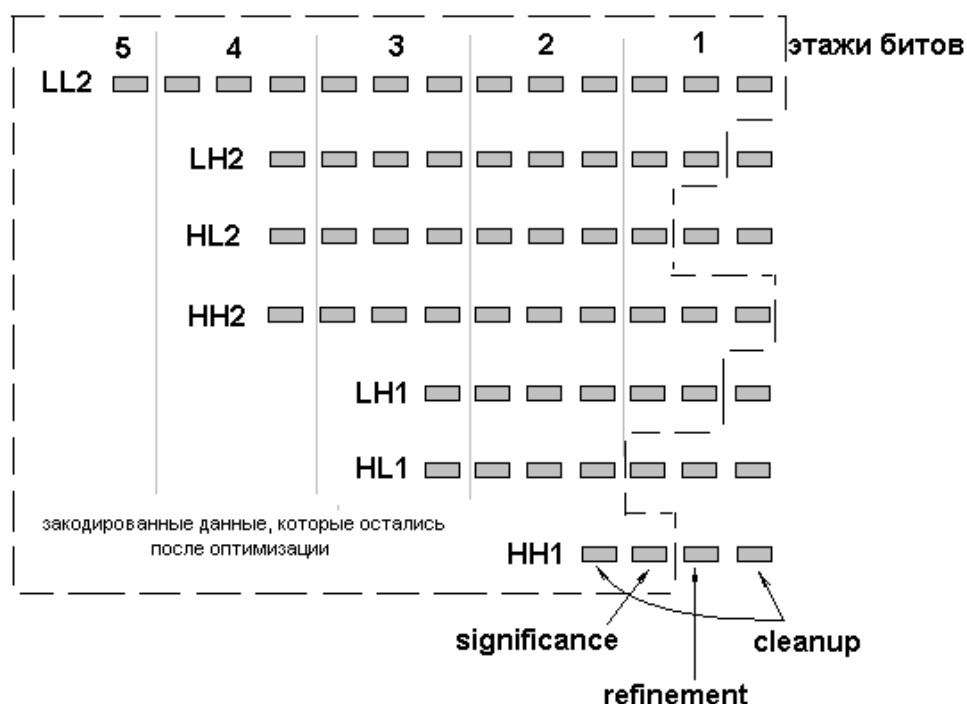


Рис. 5. Оптимизация длин закодированных данных.

В памяти остаются только те данные, которые обеспечивают лучшее качество сжатия при заданной степени сжатия.

Такой подбор делается с помощью R-D кривых (Rate-Distortion curves), а получают их следующим образом. Как уже было сказано, каждый этаж битов кодируется в 3 стадии, и на каждой стадии определяются две величины – вклад, который данная стадия дает в уменьшение среднеквадратической ошибки -- $MSE_{decrease}$, и увеличение длины кода Δ . Их отношение, т.е. величина $MSE_{decrease}/\Delta$ и есть значение R-D кривой для данной стадии. Как видно на рисунке, эта кривая убывает с увеличением количества стадий. Теперь, когда для каждого блока построены R-D кривые, нужно только выбрать подходящий уровень для обеспечения требуемой степени сжатия, и закодированные данные “усекутся” таким образом, что среднеквадратическая ошибка будет наименьшей (рис5).

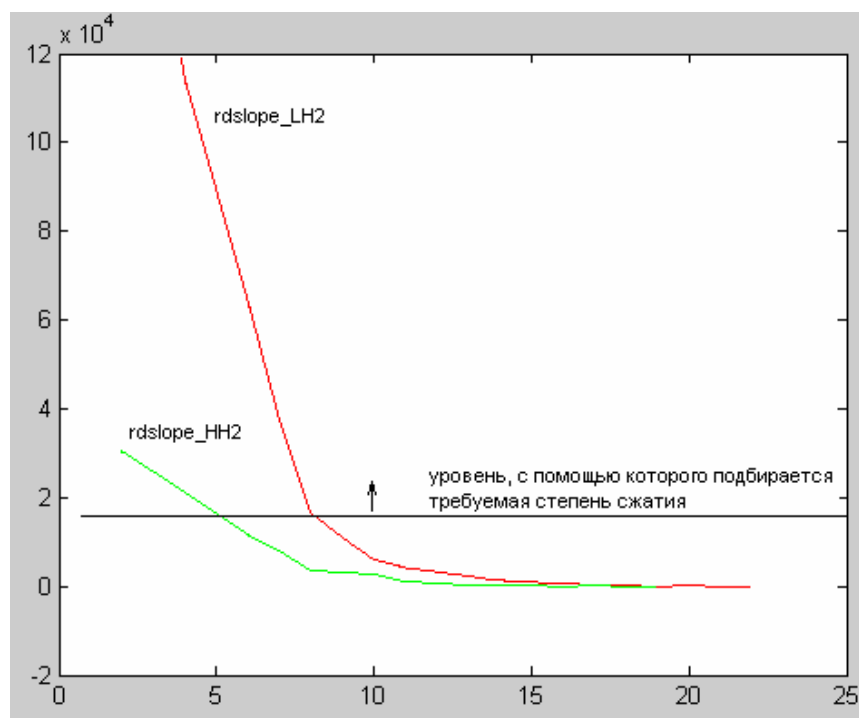


Рис. 6. Подбор соотношения длин закодированных данных.

По оси ординат – величина $MSE_{decrease}/\Delta$,
по оси абсцисс – количество стадий кодирования.

Этот метод оптимизации длины кода создает определенные неудобства, главное из которых – необходимость кодировать все блоки до наименьшего бита LSB. Поскольку именно стадия кодирования этажей битов отнимает наибольшее время, представляет интерес найти некоторый критерий, по которому можно определить, до какого бита производить кодирование – LSB или можно остановиться раньше. Например, кодирование до бита $LSB+2$ дает выигрыш во времени вычисления порядка 20%. Поиск подобного критерия упрощается, если ввести некоторые ограничения, например, рассматривать только природные изображения и ограничиться одним выбранным коэффициентом сжатия. Далее

речь пойдет о подборе такого критерия для спутникового устройства сжатия изображений со степенью сжатия 5, для остальных степеней сжатия исследование пока не проводилось. Основная проблема при нахождении такого критерия заключается в том, что трудно найти постоянную величину, не зависящую от статистики сжимаемого изображения. Например, нельзя жестко задать значение бита, до которого нужно производить кодирование – для разных изображений это значение будет разным. Для примера можно взять два космических снимка с отличающейся статистикой, например, снимок города и снимок моря, и сжать их в пять раз (рис 6).

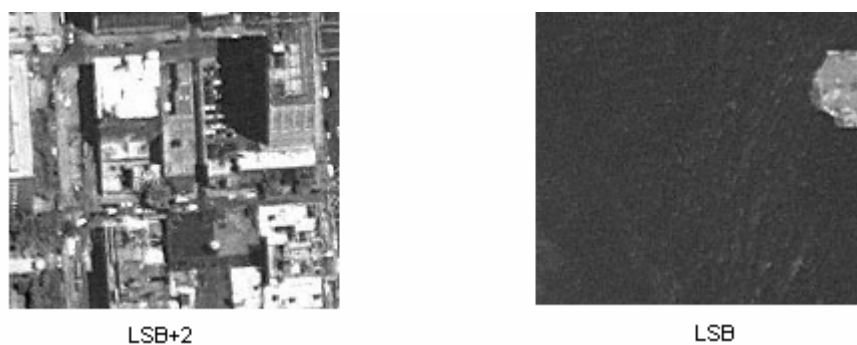


Рис. 7. Два космических снимка с разной статистикой.
Снимок моря кодируется до значения LSB, снимок города до значения LSB+2.

Так как снимок моря менее информативен, то позволяет закодировать больше этажей битов для данной степени сжатия, фактически для данного снимка у всех блоков удастся закодировать почти все этажи битов. Таким образом, производится кодирование до значения LSB. Другой снимок заметно информативнее, поэтому все этажи битов не помещаются в сжатый файл. Опытным путем установлено, что для природных изображений метод оптимизации подбирает длину блоков так, что у всех блоков в конце блока значения этажей битов одинаковы. В данном случае это LSB+2, поэтому есть смысл производить кодирование только до LSB+2. Однако, если снимок моря кодировать также до значения LSB+2, то степень сжатия окажется выше требуемой. Таким образом, конечное значение этажа битов нельзя задавать жестко.

Точно также в качестве критерия нельзя считать, например, уровень R-D кривых. То есть, если жестко задать значение R-D кривой, по достижении которого прекращать кодирование, то это может привести к сбоям в работе программы, так как этот уровень для степени сжатия 5 сильно зависит от статистики. Наиболее подходящей величиной для критерия, как показал опыт, является отношение длины закодированных данных блока LL2 к полной длине закодированных данных $\frac{b_1}{\sum_i b_i}$.

Это отношение слабо зависит от статистики и находится в интервале 0.17...0.23 для всех типов исследованных космических снимков. Данный интервал соответствует сжатию 5, для других величин этот интервал будет другой.

Исследования показывают, что зависимость этого отношения от степени сжатия линейная, причем наклон линии одинаков для всех изображений, однако этот факт пока не нашел практического применения.

Итак, получаем:

$$\frac{b_1}{\sum_i b_i} = 0.17...0.23 \approx 0.2 \quad (1)$$

Теперь эту величину надо встроить в алгоритм, а точнее – в функцию сжатия блока LL2.

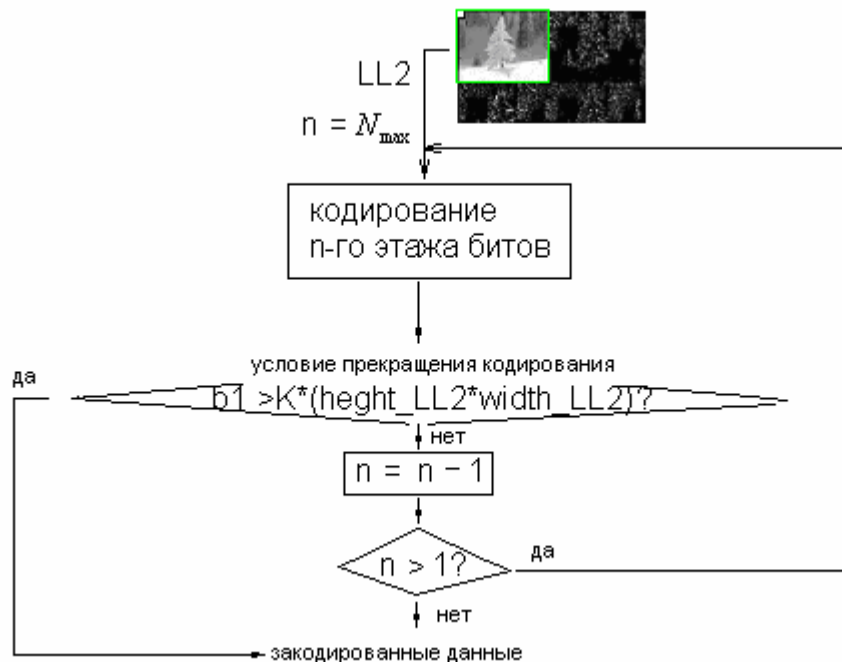


Рис. 8. Кодирование блока LL2. На выход идут закодированные данные b1.

Поскольку в функцию передается блок LL2, можно связать значение $\frac{b_1}{\sum_i b_i}$ с размерами этого блока и найти коэффициент K в неравенстве для прекращения кодирования.

$$b_1 \approx 0.2 \sum_i b_i \approx \frac{0.2HW}{5} \quad (2)$$

где H и W – соответственно высота и ширина изображения. Тогда размеры LL2 это $\frac{1}{4}H$ и $\frac{1}{4}W$. Находим K:

$$K = \frac{b_1}{\frac{1}{16}HW} \approx \frac{\frac{0.2HW}{5}}{\frac{1}{16}HW} \approx 0.64 \quad (3)$$

Таким образом, условие прекращения кодирования:

$$b_1 > 0.64 \cdot height_LL2 \cdot width_LL2 \quad (4)$$

Этот критерий испытан на ряде различных по статистике снимков и доказал свою эффективность. Например, для представленных снимков моря он обеспечивает кодирование до LSB и LSB+2 соответственно, что для снимка города обеспечивает экономию времени 18%. Если бы требовалась более сильное сжатие, тогда экономия была бы еще выше. Однако для фотографирования со спутника по техническим причинам используют степень сжатия именно в пять раз.

Как показал опыт, для сжатия космических снимков именно алгоритм JPEG 2000 оказался наиболее подходящим. Основное его достоинство в том, что он не вносит в снимок ложные высокочастотные объекты, как, например, алгоритм JPEG, а работает как низкочастотный фильтр. Это преимущество очень легко оценить визуально, однако, если сравнивать качество сжатия этих алгоритмов по распространенному критерию среднеквадратической ошибки, то по этому критерию преимущества у JPEG 2000 при сжатии некоторых снимков нет [3]. По этой причине специалисты в области сжатия изображений постепенно отказываются от этого критерия, предпочитая заменить его другими. С этой точки зрения для космических снимков большой интерес представляет тестирование качества изображения на мирах, так как установлено, что существует связь между предельно разрешаемым элементом мира и вероятностью распознавания объекта [4].

Алгоритм тестирования по критерию предельно разрешаемого элемента мира проводится на фрагменте изображения стандартной трехшпальной мира. На несжатом изображении фрагмента мира определяется предельно разрешаемый элемент по критерию предельно-разрешаемого элемента (ОСТ ВЗ-3599-77).

Суть этого критерия.

Предельно-разрешаемым элементом считается тот элемент, в котором:

1. Можно сосчитать число штрихов.
2. Число штрихов группы мира совпадает с числом штрихов группы в ее изображении.
3. Между соседними штрихами одинаковой яркости допускаются перемычки, их плотности визуально не различимы от изображения штрихов в зоне слияния, размеры перемычки не должны превышать периода группы этих штрихов.
4. Не допускаются перемычки между несмежными штрихами.

Размер шпалы этого элемента принимается за начальную точку отсчета и соответствует величине линейного разрешения на местности ($LRM_{k=1}$). Исходное изображение сжимают в 2, 3, 5 и т.д. раз и оценивают предельно разрешаемый элемент мира на сжатых изображениях (см. рис. 10). Строится график, где по оси абсцисс откладывают степень сжатия K , а по оси ординат величину (в процентах) относительного изменения LRM , т.е.

$$\left(\frac{LRM_k}{LRM_{k=1}} - 1\right) \cdot 100\%, \quad (5)$$

которое определяет снижение качества изображений.

Тестирование алгоритмов JPEG и JPEG 2000 с помощью данного критерия (Рис. 9) показало существенное превосходство JPEG 2000, особенно при высоких степенях сжатия, что хорошо согласуется с результатами визуального сравнения качества алгоритмов при сжатии аэрофотоснимков.

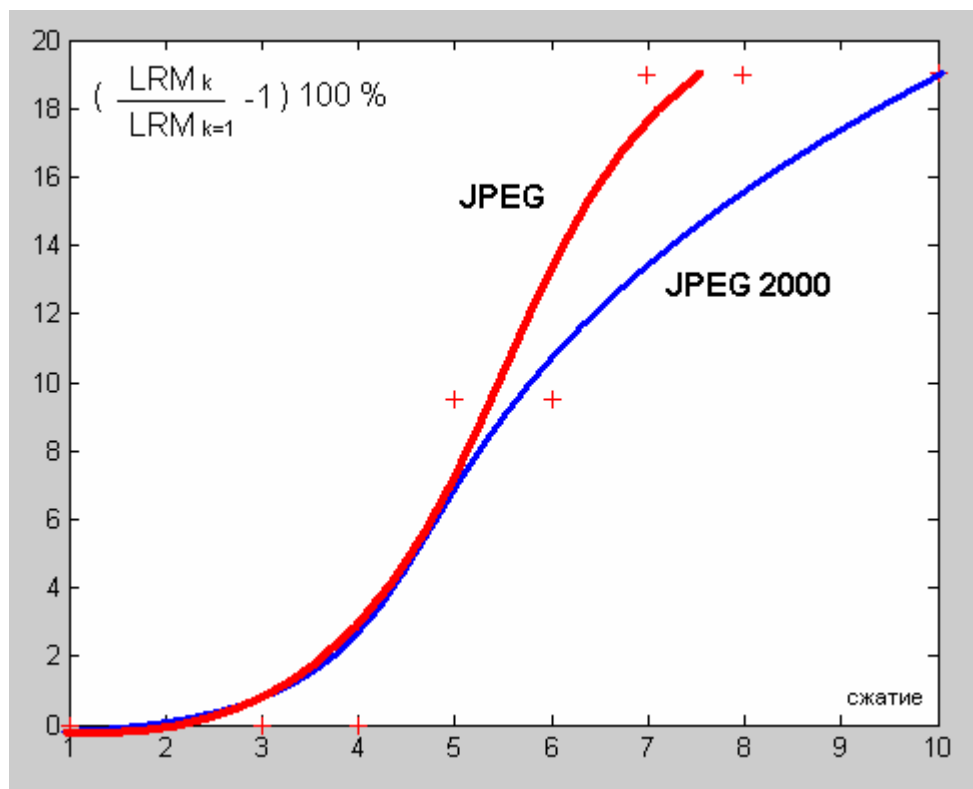


Рис. 9. Результаты сравнения алгоритмов JPEG и JPEG 2000 с помощью критерия предельно разрешимой миры.

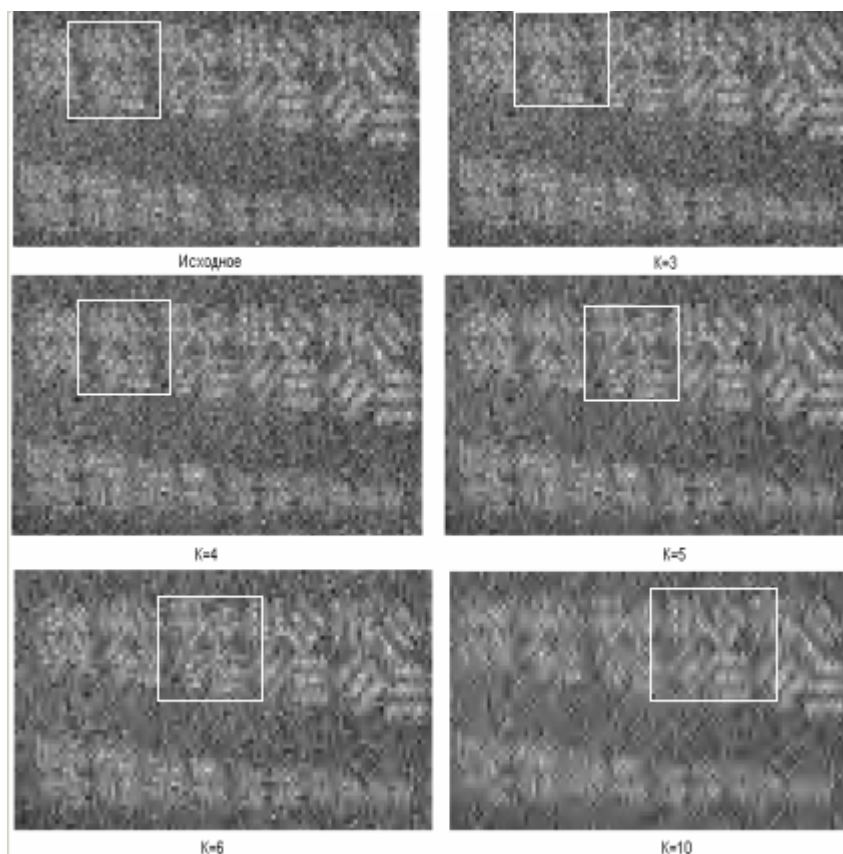


Рис. 10. Тестирование алгоритма JPEG 2000 на мирах.
Прямоугольниками отмечены предельно разрешаемые миры.

Выводы:

1. Алгоритм JPEG 2000 неэффективен по времени выполнения, так как входящие в него вейвлет-преобразование и побитовая обработка коэффициентов требуют больших временных затрат. Время выполнения можно сократить для одного частного случая, когда сжимаются снимки природного происхождения и зафиксирована степень сжатия. В этом случае существует устойчивая связь между кодированием низких и высоких частот, что позволило оптимизировать длину R-D кривых. В среднем это дает экономию времени 20%.
2. Преимущество JPEG 2000 над JPEG в качестве алгоритма сжатия в некоторых случаях нельзя однозначно установить, используя показатель среднеквадратической ошибки (MSE). Для космических снимков предложен новый метод – метод тестирования на мирах.

Литература

1. Michael D. Adams, “The JPEG-2000 Still Image Compression Standard”, June 30, 2001.

2. Jin Li, “Image Compression - the Mechanics of the JPEG 2000”, Microsoft Research.
3. Wenjun Zeng¹, Scott Daly and Shawmin Lei, “An Overview of the Visual Optimization Tools in JPEG 2000”, Sharp Laboratories of America.
4. J. M. Lloyd, “Thermal Imaging Systems”, Plenum Press, New York, 1975.