

Уменьшение граничных искажений при  
разбиении изображения на блоки

Александр Симаков  
Сыктывкарский государственный университет

15 марта 2008

### **Аннотация**

Искажения на границах блоков — это известная проблема, возникающая при вейвлетном сжатии изображений. Причем чем выше степень сжатия, тем заметнее будет этот неприятный визуальный эффект. В данной статье приводится простой и эффективный метод для уменьшения этого артефакта.

# 1 Введение

В настоящее время вейвлетные методы находят широчайшее применение в самых разных прикладных задачах. Одной из таких задач является сжатие изображений. К преимуществам вейвлетных методов можно отнести высокую степень сжатия, простой и изящный способ прогрессивной передачи, возможность заранее указывать размер архива (с точностью до байта!) и многое другое. Однако есть и некоторые сложности с которыми мы неизбежно столкнемся при разработке практической системы для сжатия изображений [1].

Поскольку изображение может оказаться очень большим, обрабатывать его необходимо поблочно. При этом, на высоких степенях сжатия становятся заметными стыки блоков. Причем чем сильнее степень сжатия, тем заметнее будет этот неприятный визуальный эффект.

В своей работе [2] авторы предлагают весьма оригинальный и эффективный способ для уменьшения искажений на границах блоков. Название статьи — «Boundary artifact reduction using odd tile lengths and the low pass first convention (OTLPF)» — фактически определяет суть метода. Авторы предлагают брать блоки размер которых не четен (Odd Tile Length) и первый коэффициент разложения — низкочастотный (Low Pass First). Поскольку низкочастотные и высокочастотные коэффициенты чередуются, а общее их количество нечетно, то и последний коэффициент разложения также будет низкочастотным. Отметим, что если размер блока взять равным  $2^N + 1$  то требуемые условия будут выполнены на любом уровне вейвлетного разложения.

Алгоритм OTLPF был успешно адаптирован для использования в программе EPSILON, которую разработал автор<sup>1</sup>. Проект EPSILON нацелен на создание открытой системы для параллельного и отказоустойчивого сжатия больших изображений. Программа имеет простую и понятную модульную структуру, поддерживает более 30 различных вейвлетных преобразований и, кроме того, может использоваться как удобный испытательный стенд для новых алгоритмов и технологий. Программа распространяется на условиях открытой лицензии GNU GPL. Полные исходные тексты EPSILON доступны для загрузки через Интернет: <http://epsilon-project.sourceforge.net>.

---

<sup>1</sup>Сайт автора: <http://entropyware.info>

## 2 Тестовое изображение

Для демонстрационных целей было использовано изображение LENA<sup>2</sup>. Уменьшенная копия изображена на Рис. 1. Исходный размер изображения —  $512 \times 512$  точек, количество бит на пиксел — 8.



Рис. 1: Тестовое изображение LENA

## 3 Метрики качества изображений

Для измерения качества восстановленных изображений использовались стандартные метрики: PNSR (1) и MSE (2). Метрика PSNR применялась ко всему изображению, а MSE — отдельно к строкам и столбцам. Нумерация элементов начинается с единицы.

$$PSNR(X, Y) = 10 \cdot \log_{10} \frac{255^2 \cdot 512 \cdot 512}{\sum_{i=1, j=1}^{512, 512} (X[i, j] - Y[i, j])^2} \quad (1)$$

$$MSE(x, y) = \frac{\sum_{i=1}^{512} (x[i] - y[i])^2}{512} \quad (2)$$

## 4 Исследование граничных искажений

Возьмем изображение LENA и сожмем его при помощи программы EPSILON в 20 раз. Размер блока выберем равным 64 (степень двойки), а в качестве вейвлетного преобразования возьмем фильтр Добеши 9/7. Сравним теперь исходное изображение с восстановленным: вычислим MSE отдельно для строк и столбцов. Результат вычислений для строк показан на Рис. 2, а для столбцов — на Рис. 3. Рассмотрим эти графики<sup>3</sup>

<sup>2</sup>Изображение можно загрузить по адресу: <http://www.ece.rice.edu/~wakin/images/>

<sup>3</sup>Все графики построены программой Gnuplot: <http://www.gnuplot.info/>

более подробно.

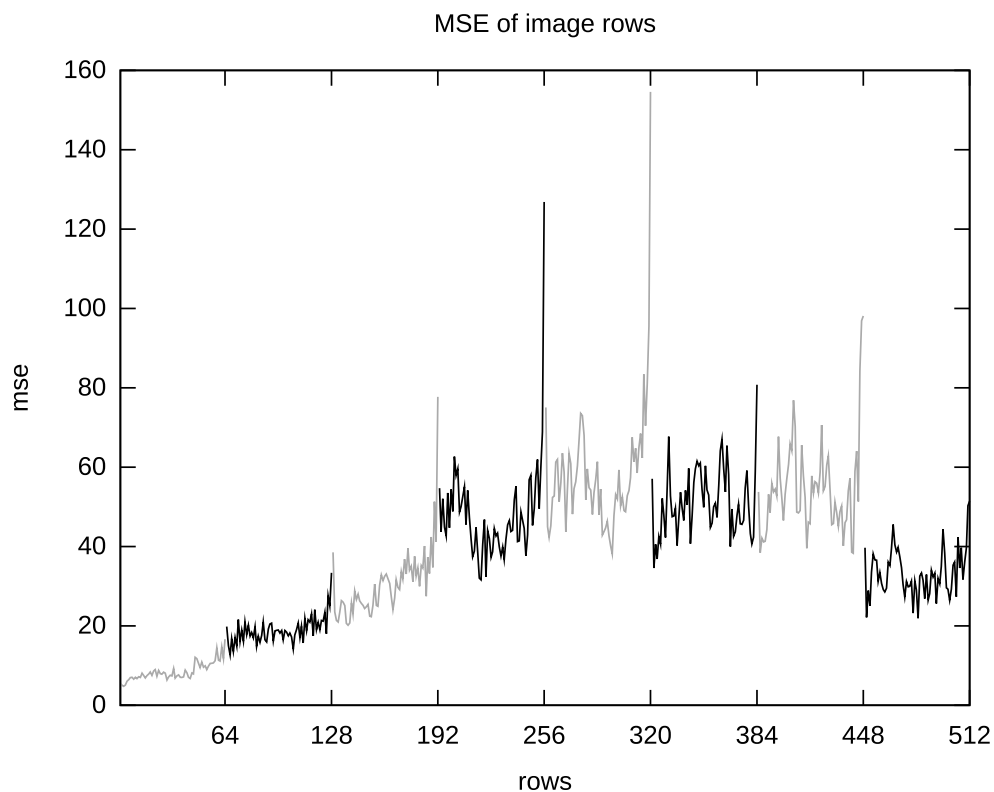


Рис. 2: MSE по строкам, размер блока 64

По оси  $x$  откладываются номера строк (столбцов), а по оси  $y$  — значение MSE между соответствующими строками (столбцами) исходного и восстановленного изображений. Для наглядности строки (столбцы) сгруппированы в полосы. Полоса — это набор идущих подряд строк (столбцов). Полосы раскрашены в чередующиеся цвета, кроме того, на оси  $x$  обозначены их границы. Отметим также, что строки в изображении следуют сверху вниз, а столбцы — слева направо.

Из графиков отчетливо видно, что величина ошибки в конце каждой полосы значительно выделяется. На Рис. 2 конец полосы соответствует самой нижней строке соответствующих блоков, а на Рис. 3 — самому правому столбцу. Таким образом, на нижней и правой сторонах блоков мы наблюдаем непропорционально большие ошибки. На Рис. 4 показан фрагмент восстановленного изображения LENA, на котором граничные искажения видны особенно отчетливо.

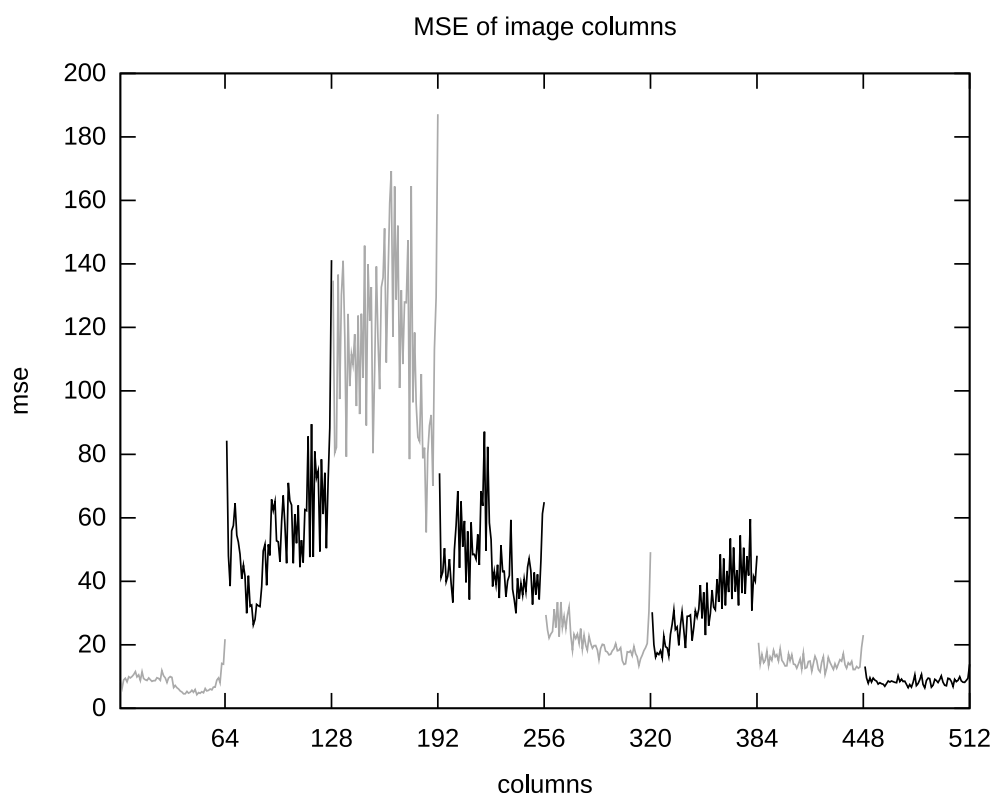


Рис. 3: MSE по столбцам, размер блока 64

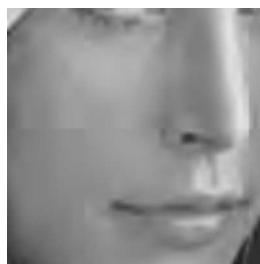


Рис. 4: Граничные искажения, размер блока 64

## 5 Уменьшение граничных искажений

Для того чтобы понять причину возникновения граничных искажений рассмотрим схему вейвлетного преобразования<sup>4</sup>. Итак:

- Изображение разбивается на квадратные блоки, ширина и высота которых равна степени двойки<sup>5</sup>.
- Каждый блок подвергается двумерному вейвлетному преобразованию.
- Двумерное вейвлетное преобразование сводится к применению одномерного сначала ко всем строкам, а затем ко всем столбцам изображения.
- Одномерное вейвлетное преобразование реализуется путем применения к сигналу низкочастотного и высокочастотного фильтров анализа. Первый фильтр выделяет низкочастотную и подавляют высокочастотную информацию, а второй — наоборот.
- Одномерное вейвлетное преобразование разбивает исходный сигнал на два подсигнала — низкочастотный (НЧ) и высокочастотный (ВЧ), причем коэффициенты размещаются в следующем порядке: НЧ ВЧ НЧ ВЧ ... НЧ ВЧ.
- Далее коэффициенты группируются по типу<sup>6</sup>: НЧ НЧ ... НЧ ВЧ ВЧ ... ВЧ.
- После того как на данном уровне вейвлетного разложения преобразованы все строки и столбцы, рекурсивно применяем двумерное вейвлетное преобразование к НЧ коэффициентам.

Обратное вейвлетное преобразование выполняется по аналогичной схеме за некоторым исключением:

- Обратное одномерное вейвлетное преобразование получает на вход сигнал, состоящий из двух частей: НЧ НЧ ... НЧ ВЧ ВЧ ... ВЧ.

---

<sup>4</sup>Данная схема реализована в программе EPSILON, в других программах процедура может несколько отличаться.

<sup>5</sup>Если размер изображения не кратен размеру блока, то изображение отражается относительно своих границ до минимально-необходимо размера. Добавленные таким образом данные игнорируются декодером при восстановлении картинка.

<sup>6</sup>Эта операция называется прореживанием (downsampling).

- НЧ и ВЧ коэффициенты помещаются на те же позиции, которые они занимали после применения прямого вейвлетного преобразования<sup>7</sup>. Пустые позиции при этом заполняются нулями. Таким образом, формируются два сигнала следующего вида: НЧ 0 НЧ 0 . . . НЧ 0 и 0 ВЧ 0 ВЧ 0 . . . ВЧ.
- К полученным на предыдущем шаге сигналам применяются низкочастотный и высокочастотный фильтры синтеза. После фильтрации, содержимое двух массивов складывается поэлементно и подается на выход процедуры.

Ключевой момент в нашем рассуждении заключается в том, что при фильтрации граничных точек, мы вынуждены мысленно продлевать сигнал в обе стороны. Другими словами, мы производим своего рода экстраполяцию данных. Ясно, что ошибки допущенные на данном этапе отразятся на качестве восстановленной картинке. Ситуация еще более усугубляется тем, что вейвлетное преобразование применяется рекурсивно и на каждой итерации ошибка будет только увеличиваться. Можно также предположить, что поскольку НЧ составляющая сигнала важнее для человеческого восприятия чем ВЧ, то и ошибки допущенные здесь будут намного заметнее для глаза.

Выдвинем гипотезу: больших ошибок на границах можно избежать если при восстановлении по краям НЧ сигнала не будут находиться нули.

Наложим на процедуру вейвлетного преобразования дополнительные условия:

- Длина сигнала должна быть нечетной
- Первый коэффициент в разложении должен быть НЧ

Поскольку коэффициенты чередуются, а длина сигнала нечетная, то и последний коэффициент также будет НЧ. Таким образом, НЧ подсигнал на этапе восстановления будет выглядеть так: НЧ 0 НЧ 0 . . . НЧ 0 НЧ. На границах нулей нет.

Отметим также, что поскольку вейвлетное преобразование вычисляется рекурсивно, условие нечетности длины сигнала должно выполняться на любом уровне разложения. Для этого достаточно брать квадратные блоки размера  $2^N + 1$ . Приведем пример: пусть размер блока равен 65. На первой итерации получим  $65 = 33$  НЧ +  $32$  ВЧ коэффициентов. Затем вейвлетное преобразование применяется к НЧ коэффициентам:  $33 = 17$  НЧ +  $16$  ВЧ и т.д.

---

<sup>7</sup>Эта операция называется нуль-восполнением (upsampling).

Проверим нашу гипотезу на практике. Возьмем изображение LENA и сожмем его в 20 раз. Размер блока выберем равным 65. В качестве вейвлетного преобразования возьмем фильтр Добеши 9/7. Как и прошлый раз, вычислим MSE между строками и столбцами исходного и восстановленного изображений. На Рис. 5 показан результат вычислений для строк, а на Рис. 6 — для столбцов.

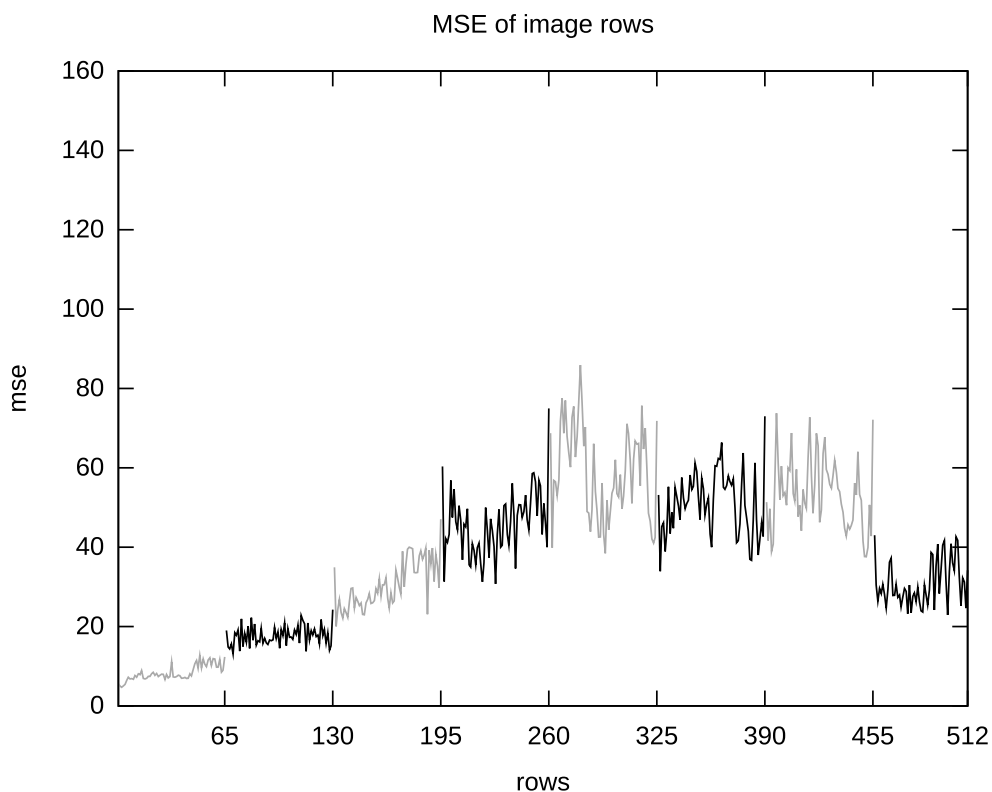


Рис. 5: MSE по строкам, размер блока 65

Из графиков отчетливо видно, что пики на концах полос практически исчезли, а сами полосы «стыкуются» теперь намного ровнее! Этот результат заметен и визуально: на Рис. 7 изображен тот же фрагмент изображения LENA, что и прошлый раз, но теперь искажений на стыке блоков существенно меньше.

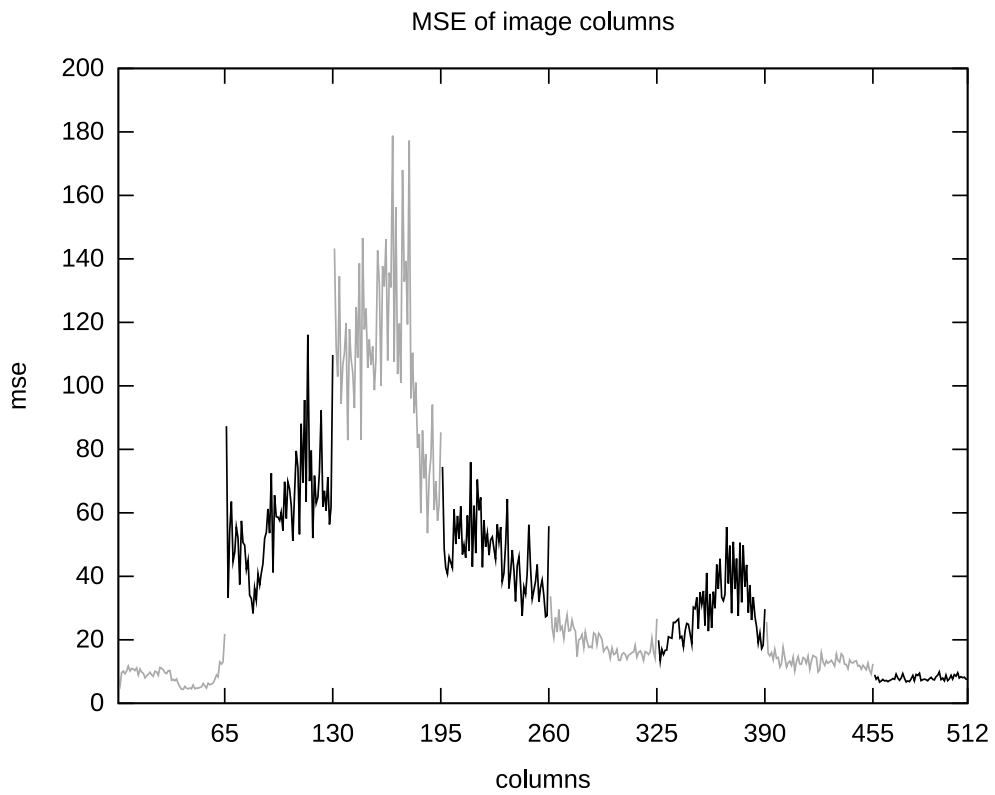


Рис. 6: MSE по столбцам, размер блока 65

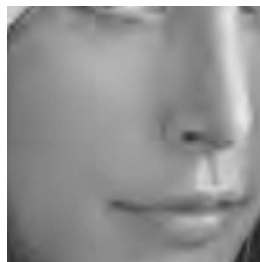


Рис. 7: Граничные искажения, размер блока 65

## 6 Выводы

Эксперименты показали, что применение алгоритма OTLPF существенно уменьшает граничные искажения на стыках блоков. Сам алгоритм очень прост и практически не требует существенных изменений в логике работы программы. Отметим, что применение алгоритма OTLPF поднимает и общий показатель PSNR для всего изображения:

Изображение	Без применения OTLPF	С применением OTLPF
LENA	32.33 dB	32.43 dB

## Список литературы

- [1] Симаков А.В. *Параллельное сжатие больших изображений* // Вестник Сыктывкарского университета. 2008. Вып. 6. Сер. 1. В печати.
- [2] Wei J, Pickering M.R, Frater M.R, Arnold J.F, Boman J, Zeng W. *Boundary artifact reduction using odd tile lengths and the low pass first convention (OTLPF)* // Proc. of SPIE, Appl. of Digital Image Proc., July, 2001.
- [3] Ватолин Д, Ратушняк А, Смирнов М, Юкин В. *Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео* // Диалог-МИФИ, 2003 г
- [4] Симаков А.В. *Прогрессивная передача изображений через Интернет* // Нелинейные проблемы механики и физики деформируемого твердого тела. СПбГУ, 2004. Вып. 8. С. 147–161.
- [5] Симаков А.В. *Сжатие изображений при помощи вейвлетных преобразований* // Вестник молодых ученых. Сер. Прикладная математика и механика. 2004. Вып. 4. С. 53–62.
- [6] Strang G, Nguyen T. *Wavelets and Filter Banks* // Wellesley-Cambridge Press, Boston, 1996.
- [7] Shapiro J. *Embedded image coding using zerotrees of wavelet coefficients* // IEEE Transactions on Signal Processing. 1993. V. 41. N 12. P. 3445–3462.
- [8] Said A, Pearlman W. *A New Fast and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees* // IEEE Trans. on Circuits and Systems for Video Technology. 1996. V. 6. P. 243–250.

- [9] Asad Islam, Pearlman W. *An embedded and efficient low-complexity hierarchical image coder //* in Proc. of SPIE V. 3653, Visual Comm. and Image Processing 99, San Jose, CA, Jan 1999, p. 294–305