

ITMO
More than a
UNIVERSITY

Современная теория информации

Лекция 8. Сжатие изображений. Часть 2.

Содержание лекции

- ① Сжатие изображений при помощи вейвлетного преобразования.
- ② Стандарт JPEG2000.
- ③ Кодеки EZW и SPIHT.
- ④ Алгоритм лагранжевых релаксаций для оптимального выбора параметров сжатия.
- ⑤ Другие подходы в сжатии изображений
 - ① Вейвлетные пакеты.
 - ② Направленное вейвлет преобразование.
 - ③ Интра предсказание.
 - ④ Стандарт JPEG-XR.
 - ⑤ Сжатие изображений на основе трехмерного ДКП.
 - ⑥ Фрактальное сжатие изображений.

1. Дискретное вейвлетное преобразование

Основные отличия от ДКП

Дискретное вейвлетное преобразование (discrete wavelet transform):

- Не требует деления изображения на блоки.
- Выделяет различные частоты из входного изображения, после чего появляется возможность точно контролировать часть сохраняемой (или отбрасываемой) информации из каждого поддиапазона частот.
- Эффективно декоррелирует изображение при помощи банка фильтров.
- Позволяет осуществить прогрессивное кодирование изображения в рамках одного битового потока (от сжатия без потерь до сжатия с потерями с разным соотношением степени сжатия/качества).

1. Дискретное вейвлетное преобразование

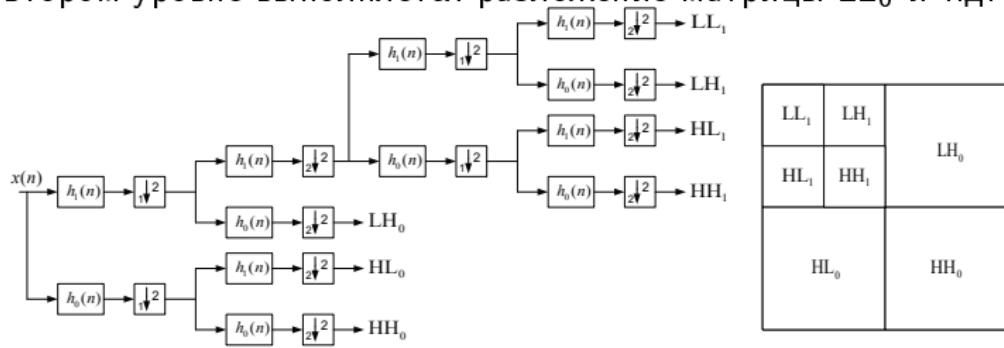
Преобразование Хаара (Haar transform). Пример.

- $h0 = \frac{1}{\sqrt{2}} \{1, 1\}$, $h1 = \frac{1}{\sqrt{2}} \{1, -1\}$, $g0 = \frac{1}{\sqrt{2}} \{1, 1\}$, $g1 = \frac{1}{\sqrt{2}} \{-1, 1\}$.
- Входной вектор $x = \{1, 2, 3, 4, 5, 6, 7, 8\}$
- Кодер:
 - ▶ Свёртка фильтрами $h0$ и $h1$:
 - ★ $y0 = x \otimes h0 = \frac{1}{\sqrt{2}} \{2, 3, 5, 7, 9, 11, 13, 15\}$
 - ★ $y1 = x \otimes h1 = \frac{1}{\sqrt{2}} \{0, -1, -1, -1, -1, -1, -1, -1\}$
 - ▶ Децимация $y0$ и $y1$ с шагом 2:
 - ★ $q0 = y0 \downarrow 2 = \frac{1}{\sqrt{2}} \{3, 7, 11, 15\}$
 - ★ $q1 = y1 \downarrow 2 = \frac{1}{\sqrt{2}} \{-1, -1, -1, -1\}$
- Декодер:
 - ▶ Восстановление $y0$ и $y1$ путём вставки нулей:
 - ★ $\hat{y}0 = q0 \uparrow 2 = \frac{1}{\sqrt{2}} \{0, 3, 0, 7, 0, 11, 0, 15\}$
 - ★ $\hat{y}1 = q1 \uparrow 2 = \frac{1}{\sqrt{2}} \{0, -1, 0, -1, 0, -1, 0, -1\}$
 - ▶ Свёртка фильтрами $g0$ и $g1$:
 - ★ $z0 = \hat{y}0 \otimes g0 = \frac{1}{2} \{3, 3, 7, 7, 11, 11, 15, 15\}$
 - ★ $z1 = \hat{y}1 \otimes g1 = \frac{1}{2} \{-1, 1, -1, 1, -1, 1, -1, 1\}$
 - ▶ Вычисление выходного вектора:
 - ★ $x = z0 + z1 = \{1, 2, 3, 4, 5, 6, 7, 8\}$

2. Стандарт JPEG2000¹

Преобразование цветового пространства и двухмерное ДВП

- Преобразование RGB в YCbCr (при сжатии с потерями) и RGB в RCT (при сжатии без потерь).
- Исп. низкочастотный и высокочастотный фильтры $h_0(n)$ и $h_1(n)$.
- На первом уровне преобразования, каждая цветовая компонента раскладывается при помощи двухмерного ДВП.
- В результате вычисляются матрицы HH_0 , HL_0 , LH_0 и LL_0 .
- На втором уровне выполняется разложение матрицы LL_0 и т.д.



¹JPEG 2000 Image Coding System: Core Coding System, ITU-T and ISO/IEC JTC 1, ITU-T Rec. T.800 and ISO/IEC 15444-1, JPEG 2000 Part 1, 2000.

2. Стандарт JPEG2000

Пример одного уровня преобразования



2. Стандарт JPEG2000

Пример двух уровней преобразования



2. Стандарт JPEG2000

Используемые в стандарте преобразования

- Daubechies 5/3 wavelets (для сжатия с потерями и без потерь):

$$h_0(n) = \begin{cases} 0.75, & \text{if } n = 0 \\ 0.25, & \text{if } n = \pm 1 \\ -0.125, & \text{if } n = \pm 2 \end{cases}$$

$$h_1(n) = \begin{cases} 0.5, & \text{if } n = 0 \\ -0.25, & \text{if } n = \pm 1 \end{cases}$$

- Daubechies 9/7 wavelets (только для сжатия с потерями):

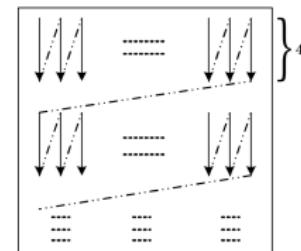
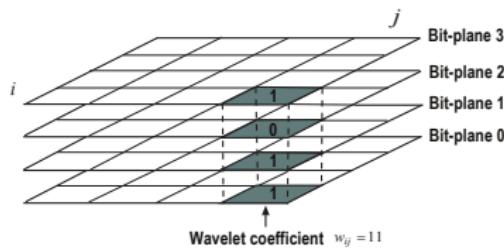
$$h_0(n) = \begin{cases} 0.602949018236, & \text{if } n = 0 \\ 0.266864118443, & \text{if } n = \pm 1 \\ -0.078223266529, & \text{if } n = \pm 2 \\ -0.016864118443, & \text{if } n = \pm 3 \\ -0.026748757411, & \text{if } n = \pm 4 \end{cases}$$

$$h_1(n) = \begin{cases} 0.557543526229, & \text{if } n = 0 \\ -0.295635881557, & \text{if } n = \pm 1 \\ -0.028771763114, & \text{if } n = \pm 2 \\ -0.045635881557, & \text{if } n = \pm 3 \end{cases}$$

2. Стандарт JPEG2000

Кодирование битовых плоскостей

- Каждая матрица ДВП представляется в виде битовых плоскостей и кодируется независимо от остальных.
- Сначала, передаётся число ненулевых битовых плоскостей.
- Затем начинается кодирование от старшей значимой битовой плоскости к младшим.
- Бит с номером p каждого вейвлет-коэффициента принадлежит битовой плоскости n .



2. Стандарт JPEG2000

Кодирование битовых плоскостей

- I. Significance propagation pass.

- ▶ Соответствующий бит незначимого коэффициента кодируется контекстным адаптивным арифметическим кодером. Всего используются 9 контекстов, которые определяются в зависимости от значимости соседних коэффициентов.
- ▶ Если текущий бит коэффициента равен единице, то такой коэффициент становится значимым.
- ▶ Знак кодируется при помощи пяти двоичных контекстов. Номер контекста определяется по знаку и значимости соседних коэффициентов.

2. Стандарт JPEG2000

Кодирование битовых плоскостей

D_0	V_0	D_1
H_0	X	H_1
D_2	V_1	D_3

Рис.: Соседние коэффициенты, используемые при определении контекста

LL и LH			HL			HH		Context number
Σ^H	Σ^V	Σ^D	Σ^H	Σ^V	Σ^D	$\Sigma^{(H+V)}$	Σ^D	
2	*	*	*	2	*	*	≥ 3	8
1	≥ 1	*	≥ 1	1	*	≥ 1	2	7
1	0	≥ 1	0	1	≥ 1	0	2	6
1	0	0	0	1	0	≥ 2	1	5
0	2	*	2	0	*	1	1	4
0	1	*	1	0	*	0	1	3
0	0	≥ 2	0	0	≥ 2	≥ 2	0	2
0	0	1	0	0	1	1	0	1
0	0	0	0	0	0	0	0	0

Рис.: Выбор контекста на significance propagation pass

2. Стандарт JPEG2000

Кодирование битовых плоскостей

- II. Magnitude refinement pass.
 - ▶ Биты значимых коэффициентов кодируются при помощи трёх двоичных контекстов.
- III. Clean-up pass.
 - ▶ Кодируется наиболее значимая битовая плоскость.
 - ▶ Кодируются биты незначимых коэффициентов на других битовых плоскостях.

2. Стандарт JPEG2000

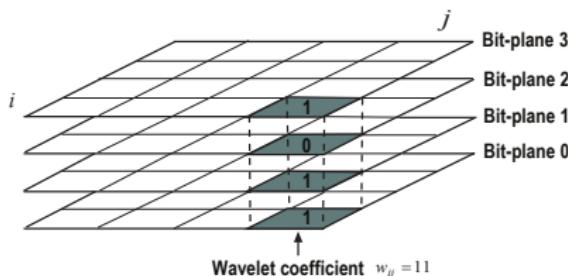
Отсечение битового потока вейвлетных матриц

- В результате генерируется прогрессивный битовый поток:

$$d_3 \geq d_2 \geq d_1 \geq d_0 = 0.$$

$$r_3 \leq r_2 \leq r_1 \leq r_0.$$

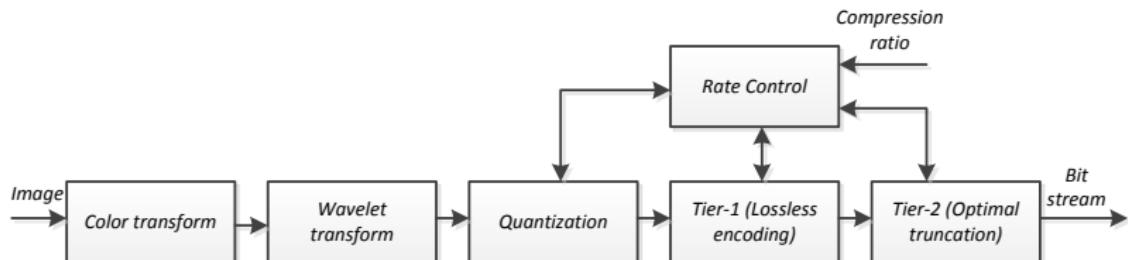
- Этот битовый поток может быть отсечен после каждого прохода.



Truncation point	Distortion	Rate
3	d_3	r_3
2	d_2	r_2
1	d_1	r_1
0	d_0	r_0

2. Стандарт JPEG2000

Общая схема кодера

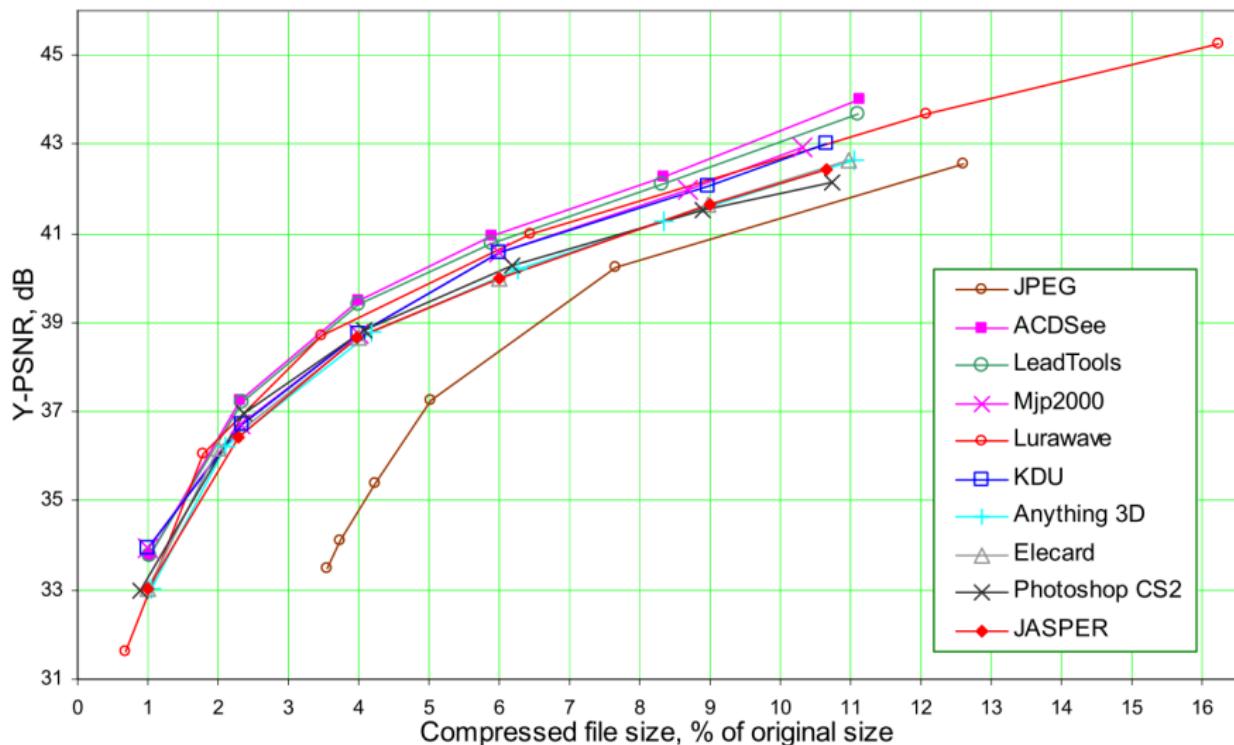


Основные этапы:

- ① Преобразование цветового пространства;
- ② Дискретное вейвлетного преобразование;
- ③ Сжатие без потерь каждой вейвлетной матрицы (Tier-1).
- ④ Достижение заданного уровня сжатия путём выбора точек отсечения (Tier-2).

2. Стандарт JPEG2000

Сравнение с JPEG²



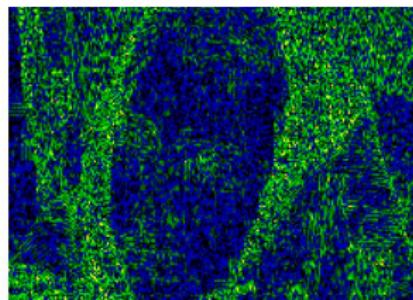
²<http://compression.ru/>

2. Стандарт JPEG2000

Сравнение с JPEG³



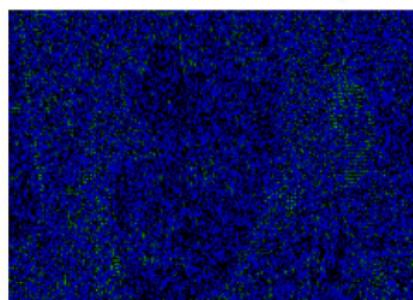
Barbara, JPEG, 31561 bytes



Difference between JPEG and original



Barbara, JPEG 2000, ACDSee, 32412 bytes

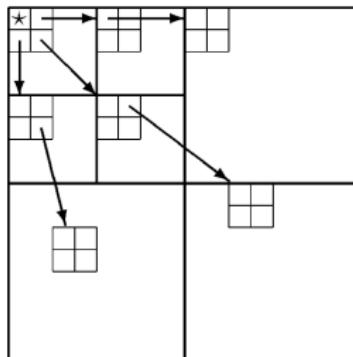


Difference between JPEG 2000 and original

³<http://compression.ru/>

3. Кодеки EZW и SPIHT

- EZW⁴ – Embedded Image Coding Using Zerotrees of Wavelet Coefficients.
- SPIHT⁵ – Image Codec Based on Set Partitioning in Hierarchical Trees.



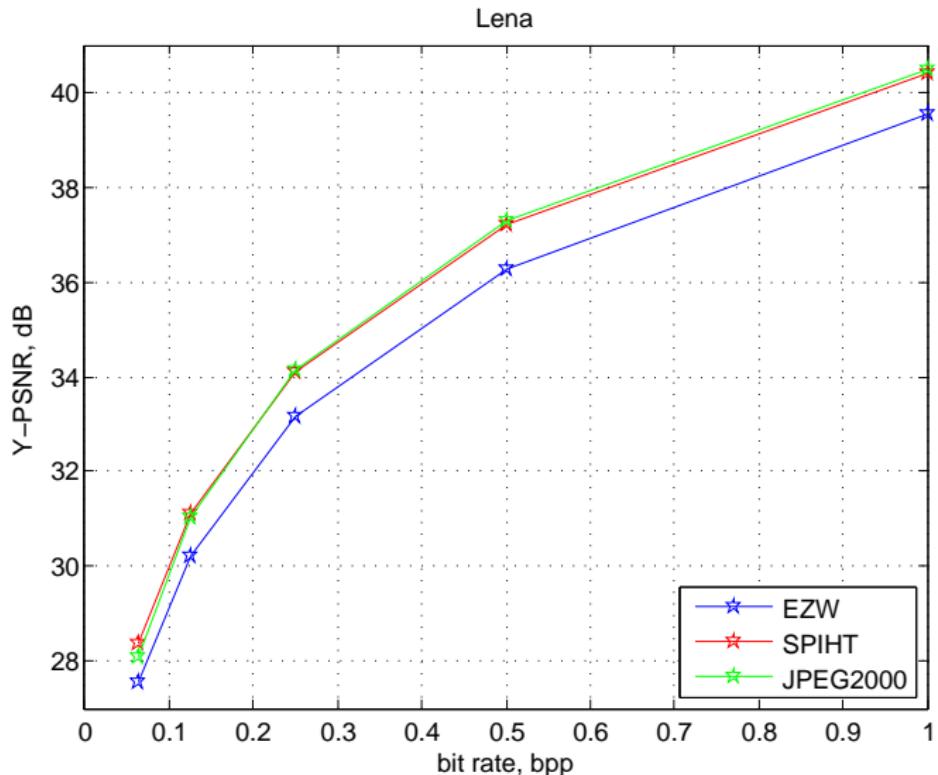
- Основная идея заключается в использовании зависимостей между коэффициентами в родительских и дочерних матрицах вейвлетного преобразования.

⁴J.M. Shapiro, Embedded image coding using zerotrees of wavelets coeicients, 1993.

⁵A.Said and W.Pearlman, A New Fast and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees, 1996.

3. Кодеки EZW и SPIHT

Сравнение с JPEG2000



4. Достижение функции скорость-искажение

Постановка задачи

- Пусть изображение разделено на неперекрывающиеся блоки $\{b_1, b_2, \dots, b_N\}$, которые кодируются независимо друг от друга. Определим $r_i^{\psi_i}$ и $d_i^{\psi_i}$ как количество бит и искажение как сумму квадратов разностей (sum of square errors, SSE) для блока b_i , когда используется вектор параметров кодирования ψ_i .
- Определим $\Psi = \{\psi_i\}$ как множество векторов параметров, где ψ_i означает, что при кодировании блока b_i используется вектор ψ_i .
- SSE для всего изображения:

$$d(\Psi) = \sum_{i=1}^N d_i^{\psi_i}.$$

- Количество бит на всё изображение:

$$r(\Psi) = \sum_{i=1}^N r_i^{\psi_i}.$$

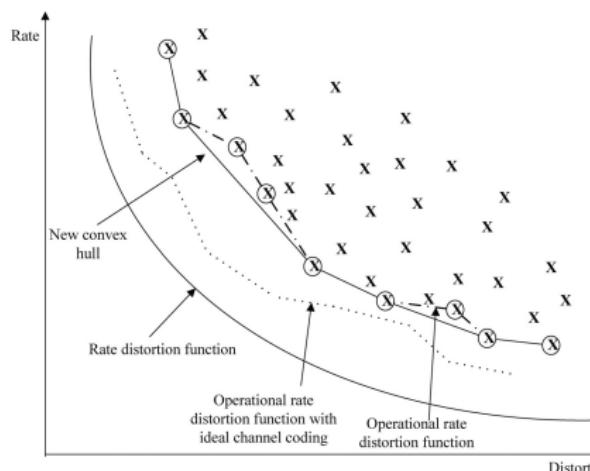
4. Достижение функции скорость-искажение

Постановка задачи

Для достижения точки на функции скорость-искажение необходимо найти такой вектор параметров кодирования $\Psi^* \in \{\Psi\}$, что

$$\begin{cases} d(\Psi^*) = \min_{\Psi \in \{\Psi\}} d(\Psi) \\ r(\Psi^*) \leq r_{max}, \end{cases}$$

где r_{max} – целевое значение количества бит на изображение.



4. Достижение функции скорость-искажение

Метод Лагранжевых релаксаций

Theorem

Для любого $\lambda \geq 0$, вектор параметров $\Psi_\lambda^* \in \{\Psi\}$, минимизирующий

$$d(\Psi) + \lambda \cdot r(\Psi), \quad (1)$$

является решением оптимизационной задачи, если $r_{max} = r(\Psi_\lambda^*)$.

Доказательство.

Предположим, что утверждение не верно, и существует вектор $\Psi \in \{\Psi\}$, такой что $d(\Psi) < d(\Psi_\lambda^*)$ и $r(\Psi) \leq r(\Psi_\lambda^*)$. Тогда $d(\Psi) + \lambda \cdot r(\Psi) < d(\Psi_\lambda^*) + \lambda \cdot r(\Psi_\lambda^*)$, т.е., вектор Ψ_λ^* не минимизирует (1), что противоречит утверждению. □

Из утверждения следует, что необходимо найти такое λ , что $r(\Psi_\lambda^*) = r_{max}$.

4. Достижение функции скорость-искажение

Метод Лагранжевых релаксаций

Theorem

Допустим, что для λ_1 и λ_2 , векторы $\Psi_{\lambda_1}^*$ и $\Psi_{\lambda_2}^*$, минимизирующие (1) найдены. Тогда, если $r(\Psi_{\lambda_1}^*) > r(\Psi_{\lambda_2}^*)$, то выполняется:

$$\lambda_2 \geq -\frac{d(\Psi_{\lambda_1}^*) - d(\Psi_{\lambda_2}^*)}{r(\Psi_{\lambda_1}^*) - r(\Psi_{\lambda_2}^*)} \geq \lambda_1. \quad (2)$$

Доказательство.

Из предыдущего утверждения следует, что

$$d(\Psi_{\lambda_1}^*) + \lambda_1 \cdot r(\Psi_{\lambda_1}^*) \leq d(\Psi_{\lambda_2}^*) + \lambda_1 \cdot r(\Psi_{\lambda_2}^*). \quad (3)$$

Из (3) и $r(\Psi_{\lambda_1}^*) > r(\Psi_{\lambda_2}^*)$ следует, что

$$-\frac{d(\Psi_{\lambda_1}^*) - d(\Psi_{\lambda_2}^*)}{r(\Psi_{\lambda_1}^*) - r(\Psi_{\lambda_2}^*)} \geq \lambda_1.$$

4. Достижение функции скорость-искажение

Метод Лагранжевых релаксаций

Доказательство.

Из первого утверждения следует, что

$$d(\Psi_{\lambda_2}^*) + \lambda_2 \cdot r(\Psi_{\lambda_2}^*) \leq d(\Psi_{\lambda_1}^*) + \lambda_2 \cdot r(\Psi_{\lambda_1}^*). \quad (4)$$

Из (4) и $r(\Psi_{\lambda_1}^*) > r(\Psi_{\lambda_2}^*)$ следует, что

$$\lambda_2 \geq -\frac{d(\Psi_{\lambda_1}^*) - d(\Psi_{\lambda_2}^*)}{r(\Psi_{\lambda_1}^*) - r(\Psi_{\lambda_2}^*)}. \quad (5)$$



Из второго утверждения следует, что функция $r(\Psi_\lambda^*)$ является неубывающей от аргумента λ .

4. Достижение функции скорость-искажение

Метод Лагранжевых релаксаций

С учетом того, что все блоки кодируются независимо, справедливо следующее равенство:

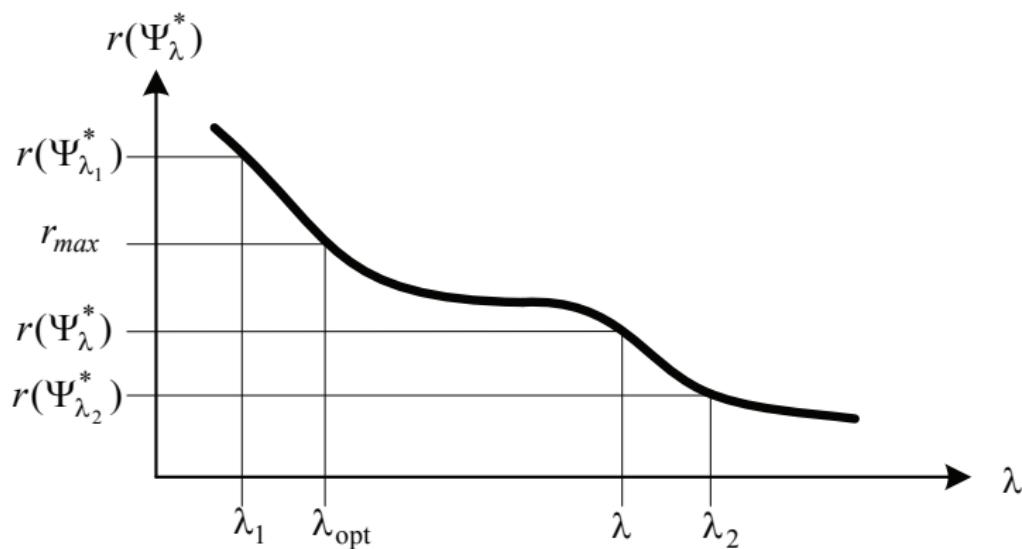
$$\min_{\Psi} \{d(\Psi) + \lambda \cdot r(\Psi)\} = \min_{\Psi} \sum_{i=1}^N (r_i^{\psi_i} + \lambda d_i^{\psi_i}) = \sum_{i=1}^N \min_{\psi_i} (r_i^{\psi_i} + \lambda d_i^{\psi_i}).$$

Поэтому, для поиска множества векторов параметров $\Psi = \{\psi_i\}$, которые минимизируют (1), достаточно минимизировать $r_i^{\psi_i} + \lambda d_i^{\psi_i}$ для каждого блока независимо. Поэтому

$$\psi_i = \arg \min_k \{r_i^k + \lambda d_i^k\}.$$

4. Достижение функции скорость-искажение

Метод Лагранжевых релаксаций



С учетом того, что $r(\Psi_\lambda^*)$ – невозрастающая функция, для поиска λ_{opt} может использоваться метод деления отрезка пополам.

4. Достижение функции скорость-искажение

Метод Лагранжевых релаксаций

Шаг 1. Найти λ_1 и λ_2 , так что неравенства $r(\Psi_{\lambda_1}^*) \leq r_{max} \leq r(\Psi_{\lambda_2}^*)$ заведомо выполняются. $\Psi^* \leftarrow \Psi_{\lambda_1}^*$, $n \leftarrow 0$.

Шаг 2.

$\lambda \leftarrow \frac{\lambda_1 + \lambda_2}{2}$. Вычислить $r(\Psi_{\lambda}^*)$.

Если $r(\Psi_{\lambda}^*) \leq r_{max}$ и $r(\Psi_{\lambda}^*) > r(\Psi^*)$, тогда $\Psi^* \leftarrow \Psi_{\lambda}^*$.
 $n \leftarrow n + 1$.

Шаг 3.

Если $|\lambda_1 - \lambda_2| > \varepsilon$ и $n \leq n_{max}$, тогда

если $r(\Psi_{\lambda}^*) \leq r_{max}$, тогда

$\lambda_1 \leftarrow \lambda$,

иначе

$\lambda_2 \leftarrow \lambda$,

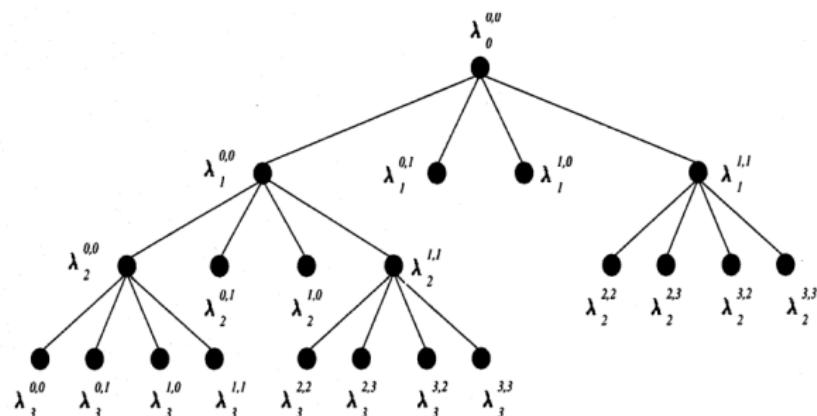
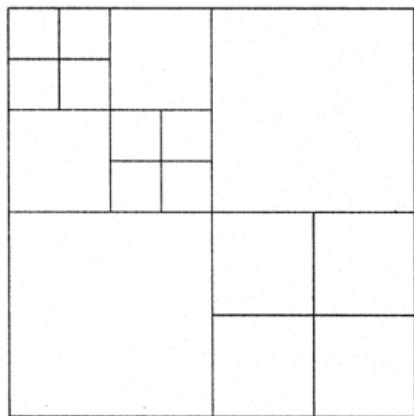
перейти на Шаг 2.

Иначе, множество векторов параметров Ψ^* найдено.

5. Другие подходы в сжатии изображений

Вейвлетные пакеты⁶

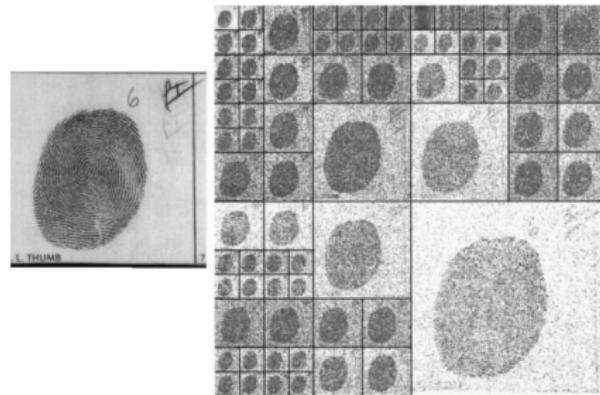
- Тип вейвлетного разложения выбирается в зависимости от свойства изображения.



⁶N. Rajpoot, R. Wilson, F. Meyer, R. Coifman, Adaptive Wavelet Packet Basis Selection for Zerotree Image Coding, 2003.

5. Другие подходы в сжатии изображений

Вейвлетные пакеты



- ВП позволяет улучшить качество на 1-1.5dB⁷.

Coder	Filters	0.20 b/p	0.25 b/p	0.30 b/p	0.40 b/p	0.50 b/p	0.60 b/p	0.70 b/p	0.80 b/p	0.90 b/p	1.00 b/p
SPIHT	7/9	26.65	27.58	28.56	30.10	31.39	32.53	33.74	34.66	35.59	36.41
WP&SPIHT	7/9	28.06	29.00	29.85	31.50	32.73	33.82	34.96	35.87	36.61	37.34

⁷Z.Xiong, K.Ramchandran, and M.Orchard, Wavelet Packet Image Coding Using Space-Frequency Quantization, 1998

5. Другие подходы в сжатии изображений

Направленное вейвлет преобразование⁸

- Вместо применения преобразования по горизонтали и вертикали, преобразование применяется в направлении наибольшей схожести пикселей.



⁸W.Ding, F.Wu, X.Wu,S.Li, H.Li, Adaptive Directional Lifting-Based Wavelet Transform for Image Coding, 2007.

5. Другие подходы в сжатии изображений

Направленное вейвлет преобразование



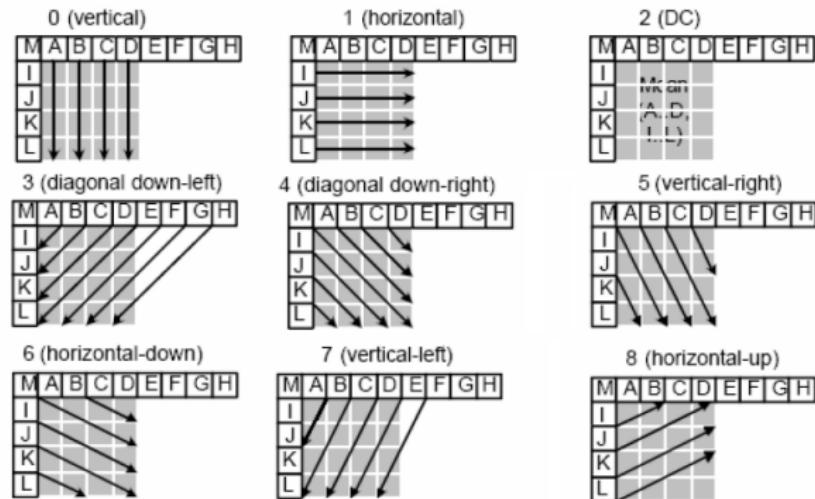
JPEG2000

Directional wavelets

Images	bpp	5/3 filter			
		J2K	ADL	ADL	ADL
			Integer Pixel	Half Pixel	Quarter Pixel
Barbara	0.125	24.59	25.69	26.07	25.95
	0.25	27.38	28.79	29.13	29.22
	0.5	30.95	32.35	32.87	32.95
	1	36.04	36.89	37.16	37.24

5. Другие подходы в сжатии изображений

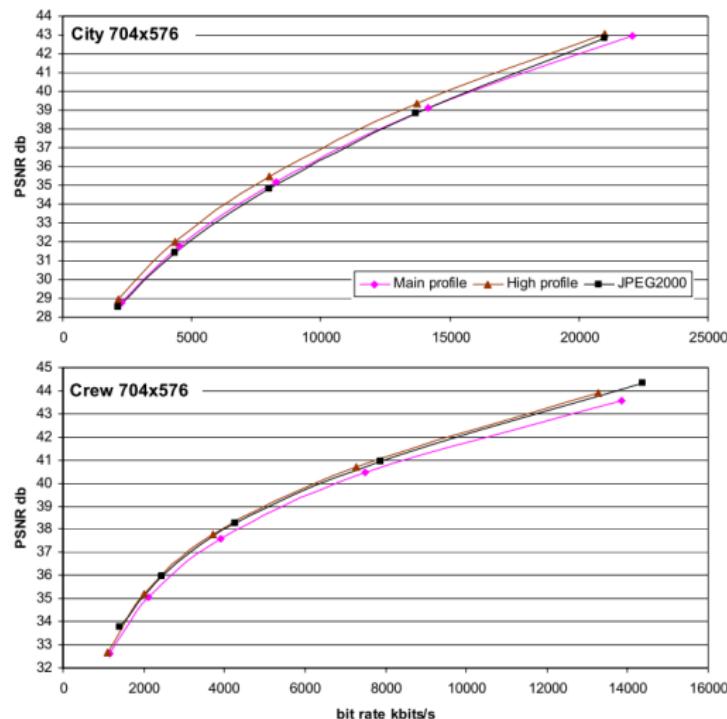
Интра-предсказание в стандартах H.264/AVC, H.265/HEVC и H.266/VVC



- Интра-предсказание принимает во внимание схожесть соседних блоков.
- Текущий блок предсказывается при помощи соседних блоков перед выполнением преобразования
- Стандарт H.264/AVC использует 9 режимов, стандарт H.265/HEVC – 33 режима, стандарт H.266/VVC – 67 режимов.

5. Другие подходы в сжатии изображений

Сравнение H.264/AVC и JPEG2000⁹ при кодировании одного изображения



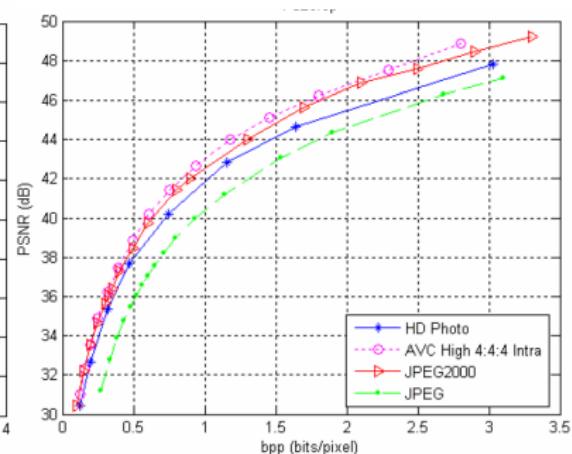
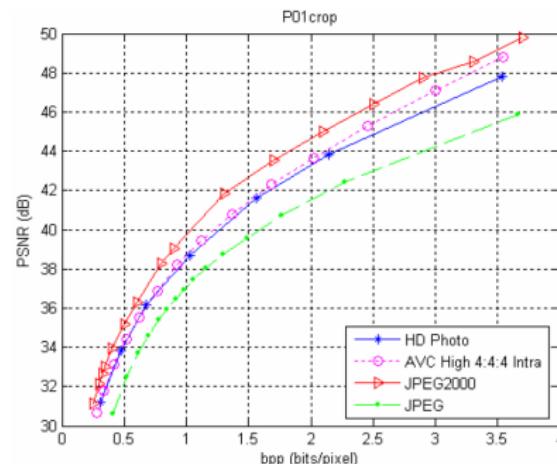
⁹M.Ouaret, F.Dufaux, On comparing JPEG2000 and Intraframe AVC, 2004.

5. Другие подходы в сжатии изображений

Стандарт JPEG-XR (Microsoft)¹⁰

Основные отличия от JPEG:

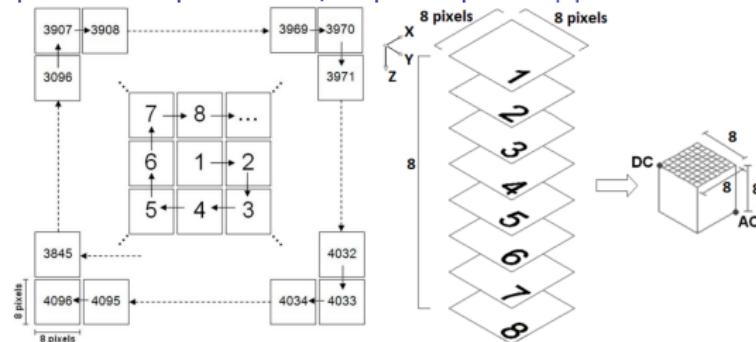
- Обратимое преобразование цветового пространства.
- Обратимое преобразование с перекрытиями;
- Адаптивное сканирование коэффициентов преобразования;



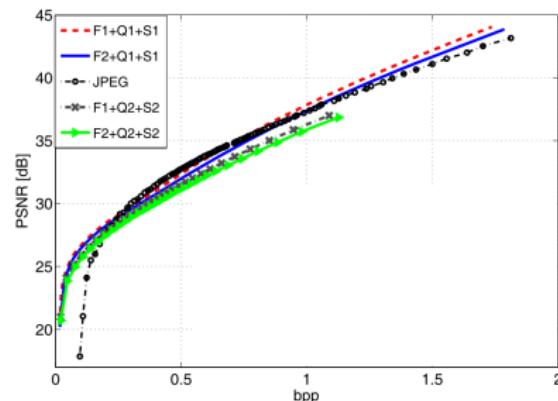
¹⁰F. De Simone, et. all, A comparative study of JPEG 2000, AVC/H.264, and HD Photo, 2008.

5. Другие подходы в сжатии изображений

Кодирование изображений при помощи трехмерного ДКП¹¹



- 1 Используется схожесть между блоками для лучшего сжатия;
- 2 Позволяет повысить качество до бдБ на высоких коэффициентах сжатия, по сравнению с JPEG.

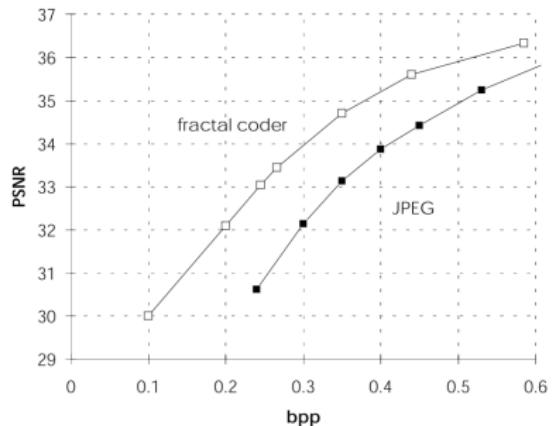


¹¹M.Engin,B.Cavusoglu, New Approach in Image Compression: 3D Spiral JPEG, 2011.

5. Другие подходы в сжатии изображений

Фрактальное кодирование изображений¹²

- Используется схожесть между далеко находящимися блоками;
- Используются афинные преобразования (поворот, масштабирование и т.д., входного изображения);



¹²K. Barthel, T.Voye, Adaptive Fractal Image Coding in the Frequency Domain, 1994.