

СПИСОК ЛИТЕРАТУРЫ

1. Кудинов А.А. Использование распознавания образов для обработки и восстановления музыкальных сигналов: Дис. ... канд. техн. наук. – М., 2003. – 179 с.
2. Симаненков И.О. Цифровые синтезаторы музыкальных звуков // Компьютерра. – 1998. – № 32. – С. 21–24.
3. Астафьева Н.М. Вейвлет-анализ: основы теории и примеры применения // Успехи физических наук. – 1996. – Т. 166. – № 11. – С. 1145–1170.
4. Фадеев А.С., Кочегурова Е.А. Выбор вейвлет-функций для анализа музыкальной информации // Современные техника и технологии: Труды XII Междунар. научно-практ. конф. студентов и молодых ученых. – Томск: ТПУ, 2006 – Т. 2. – С. 194–196.
5. Новиков Л.В. Основы вейвлет-анализа сигналов. – СПб.: ООО МОДУС+, 1999. – 152 с.
6. Курс лекций по сигналам и системам обработки сигналов. [Электронный ресурс] / авт. курса, А.В. Давыдов. – Электрон. текстовые дан. (32 файла) – [Екатеринбург]. – Режим доступа: <http://prodav.narod.ru/signals/index.html>, свободный. – Загл. с экрана. – Яз. рус., англ.
7. Кочегурова Е.А., Фадеев А.С. Вейвлет анализ в задаче идентификации музыкальной информации // Молодежь и современные информационные технологии: Сб. трудов IV Всеросс. научно-практ. конф. студентов, аспирантов и молодых ученых. – Томск: Изд-во ТПУ, 2006. – С. 149–151.
8. Смоленцев Н.К. Основы теории вейвлетов. Вейвлеты в MatLab. – М.: ДМК Пресс, 2005. – 304 с.
9. Фадеев А.С., Кочегурова Е.А. Подготовка результатов непрерывного вейвлет-преобразования к автоматизированной обработке // Известия Томского политехнического университета. – 2006. – Т. 309. – № 7. – С. 32–35.
10. Fujinaga I. Machine recognition of Timbre Using Steady-State tone of Acoustic musical Instruments // International Computer Music Conference. – Singapore, San Francisco, CA: ICMA, 2003. – P. 89–96.
11. Hebert S., Peretz I. Recognition of music in long-term memory: Are melodic and temporal patterns equal partners? // Memory & Cognition. – 1997. – V. 25. – № 4. – P. 518–533.

Поступила 19.10.2007 г.

УДК 004.93

ПРОСТОЙ АЛГОРИТМ ВЕЙВЛЕТ-СЖАТИЯ ПОЛУТОНОВЫХ И ЦВЕТНЫХ ИЗОБРАЖЕНИЙ

Д.В. Сидоров, А.Н. Осокин

Томский политехнический университет
E-mail: rauco@mail.ru

Разработан простой алгоритм вейвлет-сжатия, позволяющий реализовать недорогие аппаратно-программные платформы для систем видеонаблюдения и систем обработки телевидения высокой четкости в реальном времени.

Введение

Стандарт сжатия цифровых изображений JPEG2000 на основе вейвлет-преобразования считается одним из наиболее перспективных. Он состоит из двух обширных частей (JPEG2000 Part I, JPEG2000 Part II). В пределах стандарта существует множество программных реализаций (ACDSee, LeadTools, Mjp2000, Jasper, Lurawave и т. д.), значительно отличающихся степенью и скоростью сжатия изображений [1, 2]. Реализация JPEG2000 от ACDSee согласно [1] обладает наилучшими характеристиками: скоростью сжатия, степенью сжатия и качеством восстановленного изображения (далее при упоминании стандарта JPEG2000 будет рассматриваться реализация от ACDSee). JPEG2000 имеет явный выигрыш в 20...25 % в качестве восстановленного изображения по метрике PSNR [2, 3] (при равном коэффициенте сжатия) относительно обычного JPEG. Главной причиной ограниченного распространения JPEG2000 является сложность программной реализации и, как следствие, низкая скорость сжатия. Длительность процесса сжатия вызвана в большей степени сложностью этапов квантования, блочного кодирования и организации выходного

потока, занимающих порядка 80 % общего времени сжатия [2]. Поэтому, на основе первой и второй частей стандарта JPEG2000 (Part I, Part II) нами был разработан более простой и быстрый алгоритм вейвлет-сжатия QWC (Quick Wavelet Compress, название авторов), обладающий практически равным JPEG2000 качеством восстановленного изображения (при одинаковой степени сжатия).

Алгоритм сжатия QWC

Алгоритм QWC имеет аналогичную стандарту JPEG2000 схему сжатия (рис. 1).

Этапы предварительной обработки и вейвлет-преобразования стандарта JPEG2000 и алгоритма QWC совпадают. Согласно стандарту JPEG2000 предварительная обработка входного изображения состоит в выравнивании значений цветовых компонент RGB пикселей относительно нуля. Предварительная обработка изображения происходит только в том случае, если значения цветовых компонент RGB пикселей лежат в интервале $[0, 2^N - 1]$ (где N – разрядность бит цветовых компонент RGB или яркости пикселей изображения), тогда значение каждой цветовой компоненты RGB пикселей уменьшается на 2^{N-1} :



Рис. 1. Схема сжатия изображений стандарта JPEG2000 и алгоритма QWC

$$P_{i,j} = \begin{cases} P_{i,j}, P_{i,j} \in [-2^{N/2}, 2^{N/2} - 1] \\ P_{i,j} - 2^{N-1}, P_{i,j} \in [0, 2^N - 1] \end{cases}$$

где P_{ij} – значение одной из цветовых компонент RGB или яркости пикселя изображения, i, j – координаты пикселя в матрице изображения.

Следующим шагом алгоритма является вейвлет-преобразование [2]. Стандарт устанавливает, согласно части I, два возможных варианта преобразования:

- целочисленное преобразование 5/3, используемое для сжатия без потерь;
- иррациональное 9/7, используемое для сжатия с потерями.

В отличие от стандарта JPEG2000, где используется равномерный скалярный квантователь, в алгоритме QWC используется субполосный скалярный квантователь с различными коэффициентами квантования для субполос (уровней) высокочастотных вейвлет-коэффициентов. Применение неравномерного квантователя увеличивает качество восстановленного изображения при неизменной степени сжатия, независимо от используемого вейвлет-преобразования [4].

Исходя из специфики вейвлет-преобразования считается, что энергия высокочастотных (ВЧ) субполос HL_1 , HH_1 , LH_1 (рис. 2) описывает наиболее мелкие детали изображения, которые плохо воспринимаются человеком, следовательно, для этих субполос коэффициент скалярного квантования может быть достаточно большим, например заданным пользователем [4, 5].

Субполосы LH_2 , HH_2 , HL_2 – описывают более крупные детали и обладают большим значением энергии, следовательно, коэффициент квантования для данных полос должен быть меньше, в противном случае потери при восстановлении изображения будут значительно больше, чем на предыдущих уровнях HL_1 , HH_1 , LH_1 . Для субполос уровнем выше (HL_3 , HH_3 , LH_3 и т. д.) процесс аналогичен:

$$Q_1 > Q_2 > Q_3 \dots Q_n,$$

где Q – коэффициент квантования, n – шаг вейвлет-преобразования.

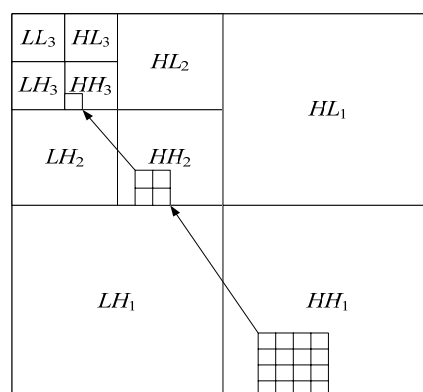


Рис. 2. Смещение энергии сигнала, представленного изображением, за три шага вейвлет-преобразования

Следует отметить, что для диапазонов HH_1 , $HH_2 \dots HH_n$ коэффициент скалярного квантования может быть значительно больше, нежели для субполос $LH_1 \dots LH_n$ и $HL_1 \dots HL_n$ вследствие слабой корреляции пикселей по диагонали, причем коэффициенты субполосы HH_1 могут быть обнулены без ощутимых потерь в качестве [2, 4, 6].

Указанная выше взаимосвязь распределения энергии между ВЧ субполосами позволяет построить субполосный скалярный квантователь с неравномерным шагом коэффициента квантования и обеспечить восстановление изображения с большей детализацией, нежели при использовании квантователя с одним коэффициентом квантования. К сожалению, вопрос о зависимости энергий субполос или способа получения коэффициентов квантования для субполос довольно сложен и на данный момент плохо освещен в современной литературе. Единственным указанием зависимости между коэффициентами субполос в литературе можно считать, что рост энергии от каждой $(n-1)$ -й полосы к n полосе зависит только от полосы $(n-1)$ -й для пирамидального разложения, рис. 2. Тогда уместно предположить, что зависимость коэффициентов квантования аналогична, т. е. коэффициент для n субполосы вычисляется из коэффициента $(n-1)$ -й субполосы [7]. Очевидно, что коэффициент квантования Q для каждой субполосы будет зависеть не только от коэффициента квантования предыдущей

субполосы, но и от конкретного сжимаемого изображения, поэтому для простоты реализации и высокой скорости квантования была оставлена только зависимость между коэффициентами квантования и введен поправочный коэффициент L :

$$Q_i = L Q_{i-1},$$

где i – уровень разложения коэффициентов или шаг вейвлет-преобразования.

В алгоритме сжатия QWC, так же как и в стандарте JPEG2000, главным параметром при сжатии, который указывает пользователь, является либо коэффициент сжатия, либо размер сжатого файла, т. е. для сжатия изображений пользователь может указать только один параметр – коэффициент сжатия K_c . Но коэффициент сжатия не единственный параметр, участвующий в квантовании и арифметическом сжатии, коэффициент квантования Q и поправочный коэффициент L также должны быть заданы. Так как пользователь задает только коэффициент сжатия (размер файла), то нужно определить зависимость коэффициента квантования Q и поправочного коэффициента L от коэффициента сжатия K_c . Теоретическое определение коэффициентов Q и L является достаточно сложным процессом, поэтому для облегчения получения коэффициентов K_c , Q и L был поставлен эксперимент.

Эксперимент

Целью проведения эксперимента является получение четырех экспериментальных зависимостей (массивов значений) коэффициентов Q и L , зависящих от K_c , и дающих наилучшее качество восстановленного изображения, т. е. необходимо получить по два массива значений коэффициентов Q и L для полутоновых и цветных изображений.

Таким образом, для каждого значения коэффициента сжатия K_c нужно экспериментально определить оптимальную пару значений коэффициентов Q и L , дающих наилучшее качество восстановленного изображения.

В качестве исходных данных для эксперимента были использованы:

- пять эталонных изображений из набора цветных изображений Calgary Corpus с глубиной цвета 24 бита на пиксель и пять полутоновых изображений глубиной 8 бит на пиксель, полученных конвертированием без потерь из оригинальных цветных (изображения выбраны с учетом их реалистичности);
- интервал коэффициента сжатия для цветных изображений $K_c \in [8, 200]$, для полутоновых изображений $K_c \in [4, 200]$, шаг 1;
- интервал коэффициента квантования Q для полутоновых и цветных изображений $Q \in [4, 60]$, шаг 1;

- интервал поправочного коэффициента L для полутоновых и цветных изображений $L \in [10, 99]$, шаг 1;
- коэффициент субдискретизации для цветовых компонент 1:8:8 (только для цветных изображений) [4].

Крайние значения интервалов коэффициентов K_c , Q и L были предварительно получены экспериментальным путем.

Качество восстановленного изображения определялась по метрике PSNR* [3]:

$$\text{PSNR}(x, y) = 10 \log_{10} \frac{255^2 nm}{\sum_{i=1, j=1}^{n, m} (x_{i,j} - y_{i,j})^2},$$

где n, m – количество отсчетов сигнала, x – отсчеты оригинального, исходного сигнала для сжатия, y – отсчеты распакованного, восстановленного сигнала.

Коэффициент сжатия K_c определялся по формуле:

$$K_c = \frac{S_o}{S_c},$$

где S_o и S_c – объем исходного и сжатого изображений.

Для проведения эксперимента и определения четырех массивов значений коэффициентов Q и L была разработана программа для СКК ТПУ «СКИФ-политех», определяющая на наборе эталонных изображений наиболее оптимальные соотношения коэффициентов K_c , Q и L . Использование кластера ТПУ обосновано большим объемом вычислений и простотой распараллеливания вычислений, например, для эксперимента над полутоновыми изображениями исходные данные будут следующими:

- количество эталонных изображений 5 (полутоновых);
- интервал коэффициента сжатия $K_c \in [4, 200]$, шаг 1;
- интервал коэффициента квантования $Q \in [4, 60]$, шаг 1;
- интервал поправочного коэффициента $L \in [10, 99]$, шаг 1.

Для эксперимента над цветными изображениями начальные данные будут следующими:

- количество эталонных изображений 5 (цветных);
- интервал коэффициента сжатия $K_c \in [8, 200]$, шаг 1;
- интервал коэффициента квантования $Q \in [4, 60]$, шаг 1;
- интервал поправочного коэффициента $L \in [10, 99]$, шаг 1.

Таким образом, для каждого значения K_c программой будут перебираться все комбинации значений коэффициентов Q и L , что без оптимизации отбора результатов может дать

* Использование метода оценки качества PSNR обосновано тем, что применение более точных методов оценки, например, VQM [3] или SSIM [3] затруднено, т. к. исходные файлы или бинарные библиотеки методов реализованы для ОС семейства Windows® и их использование на кластере ТПУ не представляется возможным, кластер управляется ОС Linux® SuSe® в консольном режиме.

$6(200-4+1)(60-4+1)(99-10+1)=6063660$ для полутоновых и $6(200-8+1)(60-4+1)(99-10+1)=5940540$ для цветных сжатых и восстановленных изображений. Максимальное количество параллельных вычислений может быть равно количеству вариантов значений K_c , Q и L . Поэтому для уменьшения количества вычислений и последующего отбора оптимальных результатов был применен метод уступок [8]. Метод позволяет задать процент уступок λ по приросту качества (критерия) каждого восстановленного изображения, тем самым гарантируя, что будет выбран оптимальный вектор прироста качества изображений и искомых значений Q и L с максимальным проигрышем между значениями внутри вектора не более λ , %. Поэтому максимальная ошибка при неправильном выборе вектора прироста качества (и значений Q и L для данного вектора), не превысит λ , %. Применение других методов, например аддитивного или мультипликативного методов свертки критериев, в общем случае может привести к взаимной компенсации частных критериев и неверному выбору оптимального вектора.

Результаты эксперимента

Результатом работы экспериментальной программы стали 4 массива значений Q и L для цветных и полутоновых изображений, задающие для выбранного пользователем коэффициента сжатия K_c значения коэффициента квантования Q и значение поправочного коэффициента L . В виду плохого качества оценки изображений по метрике PSNR часть значений массивов Q и L оказались не адекватны, зачастую в массиве значений присутствовали «пробелы» (нулевые или стартовые значения) в значениях коэффициентов. Для исправления ситуации массивы значений коэффициентов Q и L для цветных и полутоновых изображений были кусочно аппроксимированы и сглажены (см. пример на рис. 3, 4).

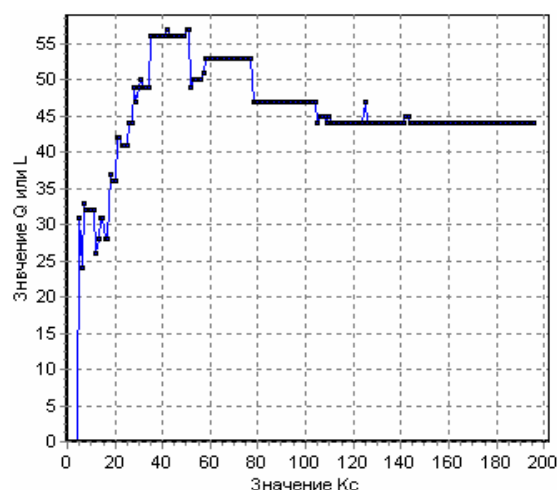


Рис. 3. Исходная кривая зависимости K_c от L для полутоновых изображений

Полученные после аппроксимации и сглаживания массивы значений коэффициентов Q и L были включены в квантователь алгоритма QWC.

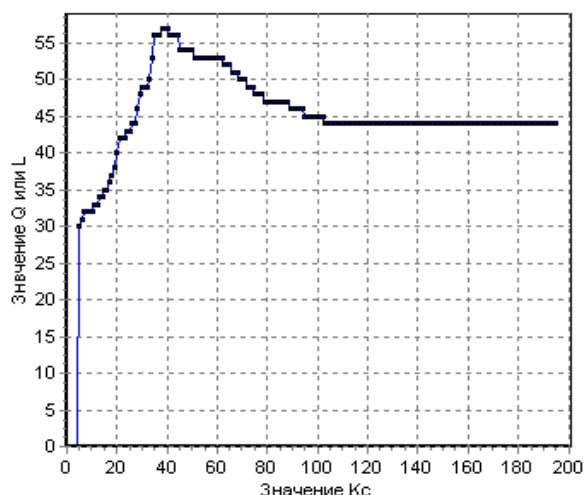


Рис. 4. Аппроксимированная кривая зависимости K_c от L для полутоновых изображений

Следующий этап сжатия согласно алгоритму QWC – этап блочного кодирования (см. рис. 1). Процесс блочного кодирования алгоритма QWC, в отличие от процесса блочного кодирования в стандарте JPEG2000, был значительно упрощен и ускорен (см. табл. 1).

Таблица 1. Этап блочного кодирования стандарта JPEG2000 и алгоритма QWC

Стандарт JPEG2000	Алгоритм QWC
ВЧ вейвлет-коэффициенты разбиваются на прямоугольные блоки произвольного размера, чаще 32×32 или 64×64 , но не более чем 4096 коэффициентов.	ВЧ вейвлет-коэффициенты разбиваются на блоки согласно уровням разложения коэффициентов (шагам преобразования).
Состояние арифметического кодера сбрасывается.	Исключен
Блок представляется в виде 11 битовых плоскостей.	Исключен
Для каждой битовой плоскости происходит предсказание значений соседних битов, для каждых 8 бит в плоскости предсказывается значение 9 бита.	Исключен
Плоскости упаковываются в виде битовых срезов, что, например, для блока в 4096 коэффициентов может дать от 1 до 11 проходов (соответственно 1 проход при максимальном сжатии и 11 проходов при минимальном) или $4096 \cdot 11 = 45056$ переборов коэффициентов при минимальном сжатии и не менее 4096 переборов коэффициентов при максимальном.	Упаковываются только сами значения коэффициентов, что, например, для блока в 4096 коэффициентов всегда даст только 1 проход, независимо от степени сжатия. В среднем количество проходов в 6..7 раз меньше, чем для стандарта JPEG2000.

Согласно стандарту JPEG2000 ВЧ вейвлет-коэффициенты сжимаются блоками. После сжатия всех ВЧ вейвлет-коэффициентов блоками, блоки перегруппировываются (согласно расположению, значимости битовых плоскостей) для записи в буфер выходного потока. В предложенном алгоритме блоки сжимаются сразу в требуемом порядке, поэтому организация выходного буфера не требуется.

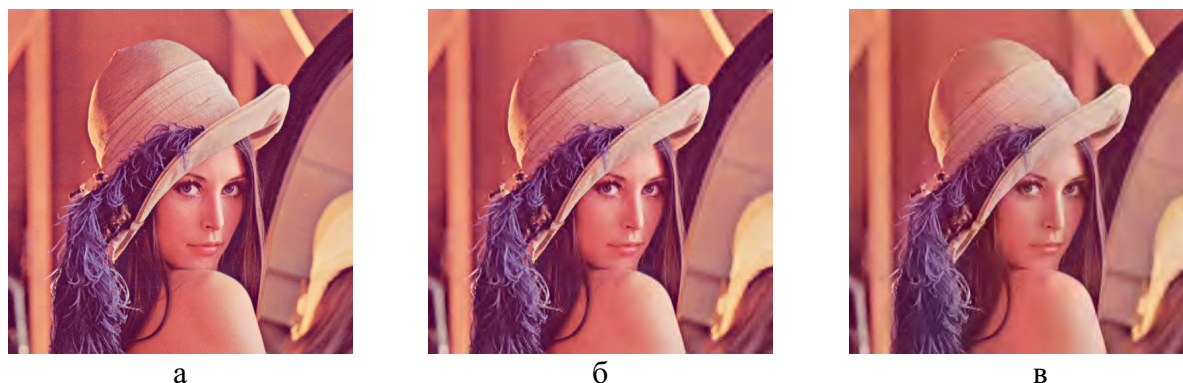


Рис. 5. Изображение: а) оригинальное; б) сжатое JPEG2000; в) сжатое QWC



Рис. 6. Участок изображения: а) оригинального; б) сжатого стандартом JPEG2000; в) сжатого алгоритмом QWC

На рис. 5, а, приведено оригинальное цветное изображение (размер изображения 512×512 пикселей, глубина цвета 24 бита), сжатое стандартом JPEG2000 (рис. 5, б, PSNR=33,77; SSIM=0,89179; $K_c=100$) и алгоритмом QWC (рис. 5, в, PSNR=30,71; SSIM=0,86779; $K_c=100$).

В табл. 2 приведены результаты сравнительного тестирования стандарта JPEG2000 и алгоритма QWC на эталонных изображениях из набора Calgary Corpus (изображения были исходными данными в эксперименте по получению массивов значений коэффициентов Q и L).

Таблица 2. Сравнение качества сжатия стандарта JPEG2000 и алгоритма QWC

Эталонное изображение из набора Calgary Corpus	Коэффициент сжатия	Показатель PSNR		Показатель SSIM		Разница в % в показателе для QWC относительно JPEG2000	
		JPEG2000	QWC	JPEG2000	QWC	PSNR	SSIM
lena.bmp	20	40,25906	33,88858	0,95792	0,95333	-18,79	-0,48
	50	36,84676	32,79163	0,92970	0,91856	-12,36	-1,21
	100	33,77797	30,71334	0,89179	0,86779	-9,97	-2,79
	150	32,07620	29,42957	0,86843	0,83540	-8,99	-3,95
	Среднее значение разницы по PSNR и SSIM						-12,53
bat.bmp	20	43,94340	34,93433	0,98256	0,97225	-25,78	-1,06
	50	36,73378	31,91089	0,96110	0,95133	-15,11	-1,02
	100	32,25620	29,47235	0,92720	0,91092	-9,44	-1,78
	150	30,15820	27,39804	0,89842	0,87320	-10,07	-2,88
	Среднее значение разницы по PSNR и SSIM						-15,10
papr.bmp	20	41,20751	34,94453	0,96824	0,95629	-17,92	-1,25
	50	37,36990	31,30622	0,93541	0,92038	-19,36	-1,63
	100	33,74086	28,04916	0,89790	0,87349	-20,29	-2,79
	150	31,64930	26,45925	0,85934	0,83862	-19,61	-2,47
	Среднее значение разницы по PSNR и SSIM						19,30
serf.bmp	20	34,50090	29,93922	0,93553	0,88968	-14,11	-5,15
	50	29,66250	27,36494	0,82053	0,77156	-7,07	-6,34
	100	27,15076	25,78640	0,70902	0,65324	-4,40	-8,53
	150	26,05561	24,87761	0,63702	0,58829	-3,70	-8,28
	Среднее значение разницы по PSNR и SSIM						-7,32
fl.bmp	20	39,26349	34,17872	0,96982	0,95998	-14,87	-1,02
	50	33,25353	31,44448	0,90210	0,90895	-5,75	+0,75
	100	30,00079	28,28929	0,83054	0,80631	-6,05	-3,01
	150	28,33185	26,63861	0,77671	0,74357	-6,36	-4,45
	Среднее значение разницы по PSNR и SSIM						-8,26
Среднее значение разницы по PSNR и SSIM						-12,5	-2,97

Согласно проведенному тестированию (табл. 2) алгоритм QWC проигрывает JPEG2000 в среднем 3 % по метрике SSIM, по метрике PSNR – 12,5 %. Экспертная оценка показывает, что метрика PSNR фактически не дает более или менее адекватную оценку качества, т. к. визуально изображения слабо различимы.

Разработанный и реализованный алгоритм QWC эффективно сжимает и изображения, не использованные в эксперименте по формированию массивов значений коэффициентов Q и L . В качестве примера на рис. 6, а, изображен участок произвольного изображения (не участвовавшего в определении коэффициентов, размер изображения 1600×1200 пикселей, глубина цвета 24 бита), сжатого стандартом JPEG2000 (рис. 6, б, SSIM=0,70929; $K_c=60$) и алгоритмом QWC (рис. 6, в, SSIM=0,64978; $K_c=60$). Экспертная оценка отдает предпочтение изображению сжатому QWC (рис. 6, в) перед JPEG2000 (рис. 6, б), т. к. алгоритм QWC при сжатии сохранил структуру мелких деталей изображения (стебли травы).

Заключение

Разработан более быстрый и простой алгоритм вейвлет-сжатия QWC с пирамидальной схемой квантования и сохранения коэффициентов. Он обладает практически равным стандарту JPEG2000 качеством восстановленных изображений (при равной степени сжатия), а в некоторых случаях превос-

ходит его. Алгоритм QWC может быть довольно легко перенесен на недорогую аппаратную платформу, способную в режиме реального времени обрабатывать видео-поток телевидения высокой четкости.

Разработка алгоритма сжатия поддерживается по программе «Участник молодежного научно-инновационного конкурса» («У.М.Н.И.К.») Фондом содействия развитию малых форм предприятий в научно-технической сфере.

СПИСОК ЛИТЕРАТУРЫ

1. Ватолин Д., Москвин А., Петров О. Сравнение кодеков изображений стандарта JPEG2000 и Windows Media Photo (новое название: Microsoft HD photo) [Электронный ресурс]. – режим доступа: http://compression.ru/video/codec_comparison/wmp_codecs_comparison.html (20.06.2007).
2. Ватолин Д., Ратушняк А., Смирнов М., Юкин В. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео. – М.: ДИАЛОГ-МИФИ, 2002. – 384 с.
3. Ватолин Д. Программный продукт оценки качества восстановленного изображения MSU Quality Measure [Электронный ресурс]. – режим доступа: http://compression.ru/video/quality_measure/video_measurement_tool.html (20.06.2007).
4. Сидоров Д.В., Осокин А.Н. Аппаратно-ориентированный субполосный квантователь для вейвлет-сжатия полутоновых изображений // Молодежь и современные информационные технологии: Сб. трудов V Всерос. научно-практ. конф. студентов, аспирантов и молодых ученых. Томск, 27 февраля – 1 марта 2007 г. – Томск: Изд-во ТПУ, 2007. – С. 446–448.
5. Уэлстид С. Фракталы и вейвлеты для сжатия изображений в действии. – М.: Триумф, 2003. – 320 с.
6. Rabbani M., Santa Cruz D. The JPEG2000 Still-Image Compression Standart. – Lausanne: Swiss Federal Institute of Technology (EPFL), 2002. – 246 p.
7. Sagetong P., Ortega A. Analytical model-based bit allocation for wavelet coding with applications to multiple description coding and region of interest coding // Proc. IEEE Int. Conf. on Multimedia and Expo (ICME). Tokyo, Japan August 2001. – Los Angeles: SIPI Department of EES University of Southern California, 2001. – P. 300–304.
8. Штойер Р. Многокритериальная оптимизация: теория, вычисления и приложения: пер. с англ. / Р. Штойер; под ред. А.В. Лотова. – М.: Радио и связь, 1992. – 504 с.

Поступила 12.10.2007 г.

УДК 681.3.06

МАТРИЧНЫЙ АЛГОРИТМ РЕШЕНИЯ ЗАДАЧИ РАЗРЕЗАНИЯ ГРАФОВ

В.К. Погребной

Институт «Кибернетический центр» ТПУ
E-mail: vk@ad.cctpu.edu.ru

Предложен матричный алгоритм решения задачи разрезания графов. Рассмотрены основные положения алгоритма, базирующиеся на матричном представлении графов. Приведена формализация основных процедур алгоритма – определение оценок для выбора перемещаемых элементов матрицы и преобразование матрицы путем взаимного переноса столбцов и строк. Работа алгоритма рассмотрена на примере графа передач данных между станциями локальной сети вычислительной системы.

Введение

Задачу разрезания графа на минимально связанные части можно отнести к числу классических задач теории графов, широко используемых в практических приложениях. Среди наиболее ранних и хорошо разработанных приложений выделяются задача разбиения схемы вычислительного устройства, представленного в виде графа [1] или в виде более адекватной модели – гиперграфа [2]. Для обыкновенного графа [3] задача разрезания формулируется следующим образом. Требуется разрезать граф $G=(S, V)$ на части $G_f=(S_f, V_f)$, $f=1, 2, \dots, F$, где F – число частей, на которые разрезается граф; S_f – множество вершин, принадлежащих f -ой части; V_f – множество ребер инцидентных вершинам S_f .

Совокупность частей $B(G)$ называется разбиением графа G , если

$$\forall G_f \in B(G) [G_f \neq \emptyset \& \bigcup G_f = G];$$

$$\forall G_f, G_q \in B(G) [G_f \neq G_q \& S_f \cap S_q = \emptyset \& V_f \cap V_q = V_{fq}], \quad f, q = 1, 2, \dots, F.$$

Здесь V_{fq} – множество ребер, связывающих пары вершин, одна из которых принадлежит части G_f , а другая – G_q .

Обозначим $|V_{fq}|=g_{fq}$ и назовем его числом реберного соединения частей G_f и G_q . Тогда число реберного соединения разрезания графа G определяется величиной g :

$$g = \sum_{f=1}^F \sum_{q=1}^F g_{fq}, \quad f \neq q.$$

Традиционным критерием решения задачи разрезания графа G является минимизация числа реберного соединения g при ограничении на число вершин в частях G_f .