

СЖАТИЕ ИЗОБРАЖЕНИЙ НА ОСНОВЕ ВЕРОЯТНОСТНОГО КОДИРОВАНИЯ ДЛИН СЕРИЙ И ДВУХПОРОГОВОГО КВАНТОВАНИЯ

Х.К. Аль-Бахдили, В.Ю. Цветков, В.К. Конопелько

*Белорусский государственный университет информатики и радиоэлектроники
П. Бровки, 6, Минск, 220013, Беларусь*

Предложены модифицированные алгоритмы кодирования длин серий для сжатия полутоновых изображений без потерь и с потерями, отличающиеся от базового алгоритма учетом вероятности повторения значений пикселей изображений и обеспечивающие за счет этого уменьшение размера кодированных данных.

Ключевые слова: сжатие изображений, кодирование длин серий

Введение

Для сжатия изображений с потерями в настоящее время широко используются алгоритмы JPEG [1] и JPEG 2000 [2], основанные на энтропийном кодировании (Хаффмана и арифметическом) коэффициентов дискретно-косинусного и вейвлет-преобразований. Данные алгоритмы позволяют сжимать изображения в десятки - сотни раз с приемлемым качеством, однако имеют достаточно высокую вычислительную сложность. Для сжатия изображений без потерь достаточно эффективны алгоритмы LZW (Lempel-Ziv-Welch) (используемый в стандартах GIF (Graphics Interchange Format), TIFF (Tagged Image File Format) [5, 6]), Deflate и LZMA (используемые в архиваторах Zip, 7-zip [3]), PPM и LZSS (используемые в архиваторе Rar [3]), кодирующие значения пикселей. Данные алгоритмы позволяют сжимать полутоновые изображения без потерь примерно до 2 раз в зависимости от распределения яркости пикселей, однако эти методы также имеют достаточно высокую вычислительную сложность. В тех случаях, когда временные и вычислительные ресурсы ограничены необходимо использовать более простые алгоритмы эффективного кодирования. К ним относится алгоритм кодирования длин серий RLE (Run-Length Encoding) [4], основанный на учете повторов символов. Он может использоваться для сжатия без потерь изображений с малым числом резких перепадов яркости (мультипликационных, медицинских, квантованных и сегментированных). В сочетании с предварительным квантованием изображений алгоритм RLE может использоваться для сжатия с потерями. Целью работы является разработка модифицированных алгоритмов кодирования длин серий, основанных на учете вероятности повторения значений пикселей для сжатия изображений без потерь и с потерями.

Алгоритм RLE

Алгоритм RLE основан на подсчете числа повторов значений следующих друг за другом символов, может иметь блочную и поточную реализацию.

Блочная реализация алгоритма RLE предполагает предварительное накопление кодируемых данных для их анализа и выбора параметров кодирования. Анализ производится на основе таблицы I/N-длин серий, в которой одному или нескольким одинаковым по значению и следующим друг за другом символам I ставится в соответствие их количество N (табл. 1).

Табл.1. Таблица длин серий

<i>I</i>	<i>i</i> (0)	<i>i</i> (1)	...	<i>i</i> (<i>s</i>)	...
<i>N</i>	<i>n</i> (0)	<i>n</i> (1)	...	<i>n</i> (<i>s</i>)	...

На основе таблицы длин серий определяются битовая глубина BD_I изображения и битовая глубина значений длин серий BD_N с помощью выражений:

$$BD_I = \left\lceil \log_2 \left(\max(i(s))_{(s=0, S-1)} \right) \right\rceil$$
$$BD_N = \left\lceil \log_2 \left(\max(n(s))_{(s=0, S-1)} \right) \right\rceil,$$

где $i(s)$ – значение кодируемого s -го символа из таблицы длин серий; $n(s)$ – число повторов s -го символа (длина серии); S – число кодируемых символов (число строк в табл. 1). Затем осуществляется кодирование длин серий в результате формирования на выходе кодера пар $\{i(s), n(s)\}$. Блочная реализация алгоритма RLE позволяет получить минимальный объем кода. Ее недостатком является задержка при кодировании, необходимая для накопления данных и построения таблицы длин серий.

При поточной реализации алгоритма RLE таблица длин серий может не строиться, а подсчет числа символов может осуществляется по мере их поступления. Это позволяет существенно повысить скорость кодирования. Структуры алгоритма и закодированных данных при поточной реализации алгоритма RLE такие же, как при блочной. Значения битовой глубины BD_I изображения и битовой глубины значений длин серий BD_N выбираются независимо от поступающих данных. Они могут быть выбраны с избытком или недостатком (при переполнении $n(s)$ формируется новая серия $\{i(s), n(s)\}$). В некоторых случаях это может приводить к увеличению объема кода, что является недостатком поточной реализации алгоритма RLE.

При кодировании изображений размер $R_{I/N}$ (бит) кода, коэффициент сжатия CR и вычислительная сложность $C_{I/N}$ для алгоритма RLE определяются с помощью выражений:

$$R_{I/N} = S(BD_I + BD_N),$$
$$CR = 8YX / R_{I/N},$$
$$C_{I/N} = YX + 4S$$

где YX – размер изображения, определяющий число операций на буферизацию и формирование таблицы I/N размером S записей; $4S$ – дополнительные операции на поиск максимального значения по I (S операций), поиск максимального значения по N (S операций), кодирование и передачу ($2S$ операций).

Алгоритм кодирования длин серий I2/B/N

Предлагается модификация I2/B/N алгоритма кодирования длин серий RLE для сжатия полутоновых изображений, основанная на учете вероятности повтора значений пикселей в строках. В процессе выполнения алгоритма строится таблица N/P вероятностей повторов (табл. 2), в которой каждому значению длины серии n ставится число таких серий $p_n(n)$.

Табл.2. Таблица вероятности повторов

N	1	2	...	n	...
P	$p_n(1)$	$p_n(2)$...	$p_n(n)$...

Для изображений характерно постепенное убывание функции $p_n(n)$ при увеличении значения n .

При кодировании длины серии по алгоритму I2/B/N сначала формируется символ I. Затем, если символ I повторяется, то формируется бит $b1(s)=1$, иначе $b1(s)=0$ (первый символ повтора). Если символ I повторяется снова, то формируется бит $b2(s)=1$, иначе $b2(s)=0$ (второй символ повтора). Если символ I повторяется снова, то формируется символ $n(s)$, учитывающий число повторов (изначально $n(s)=0$, если символ I повторяется снова, то $n(s) > 0$). В результате могут формироваться серии:

$$\{i(s), b1(s)=0\}, \{i(s), b1(s)=1, b2(s)=0\}, \{i(s), b1(s)=1, b2(s)=1, n(s)\}.$$

Структуры алгоритма I2/B/N кодирования длин серий и закодированных им данных приведены на рис. 1.

Алгоритм I2/B/N кодирования длин серий также, как алгоритм RLE, может иметь блочную и поточную реализации.

Размер $R_{I/2B/N}$ (бит) кода для алгоритма I2/B/N определяется с помощью выражения

$$R_{I/2B/N} = S(BD_I + 1) + \sum_{s=0}^{S-1} b1(s) + BD_N \sum_{s=0}^{S-1} b2(s) \quad (1)$$

Для определения вычислительной сложности используется выражение

$$C_{I/2B/N} = YX + 4S + \sum_{s=0}^{S-1} b1(s) + \sum_{s=0}^{S-1} b2(s) \quad (2)$$

Выражение (2) учитывает YX операций на буферизацию и формирование таблицы I/N-длин серий размером S записей, S операций на поиск максимального значения по I, S операций на поиск максимального значения по N, $2S + \sum_{s=0}^{S-1} b1(s) + \sum_{s=0}^{S-1} b2(s)$ операций на кодирование и передачу.

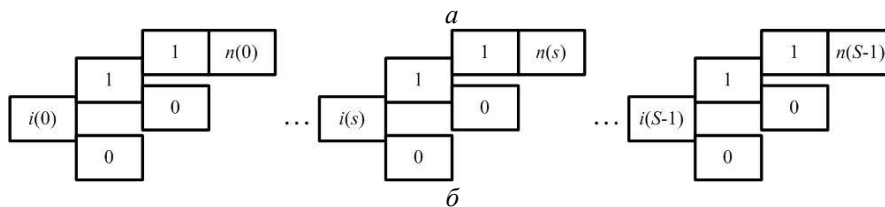
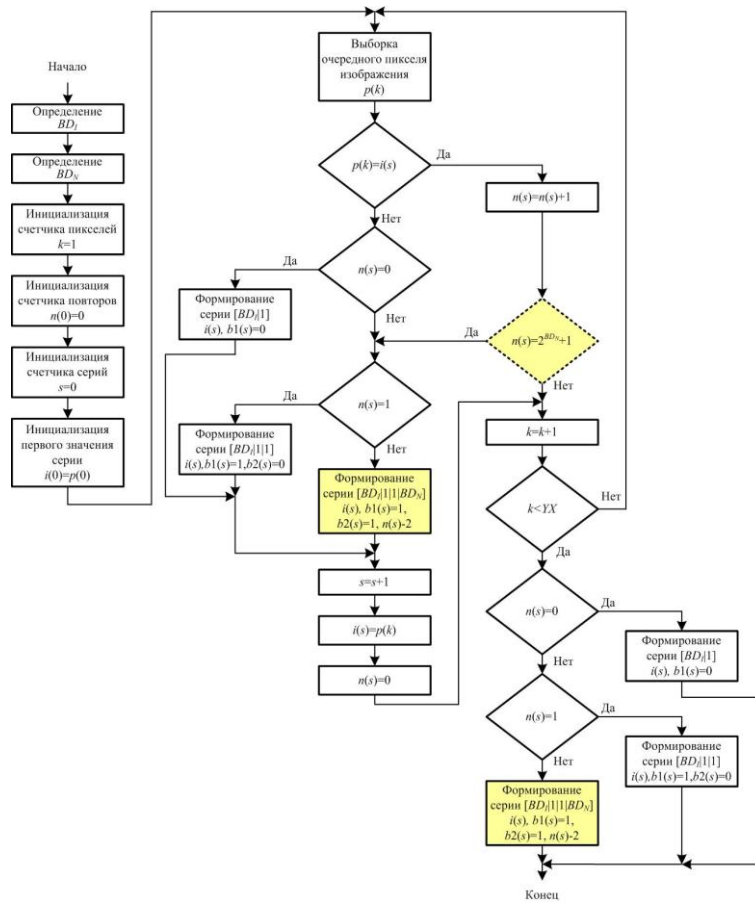


Рис. 1. Модифицированное I2/V/N кодирование длин серий
 а – алгоритм I2/V/N кодирования длин серий; б – структура закодированных данных по алгоритму I2/V/N

Алгоритм кодирования длин серий I3/V/N

Предлагается модификация I3/V/N алгоритма кодирования длин серий RLE для сжатия полутоновых изображений, основанная на учете вероятности повтора значений пикселей в строках. Алгоритм I3/V/N отличается от алгоритма I2/V/N использованием дополнительного символа $b3(s)$, который принимает значение 1, если символ I повторяется в третий раз подряд, и принимает значение 0, если третий повтор символа I отсутствует. Структура закодированных данных в соответствие с алгоритмом I3/V/N приведена на рис. 2.

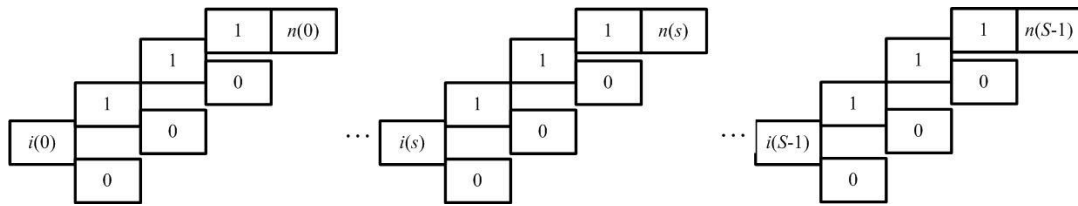


Рис. 2. Структура закодированных по алгоритму I3/V/N данных

Размер $R_{I/3B/N}$ (бит) кода для алгоритма I3/V/N определяется с помощью выражения

$$R_{I/3B/N} = S(BD_I + 1) + \sum_{s=0}^{S-1} b1(s) + \sum_{s=0}^{S-1} b2(s) + BD_N \sum_{s=0}^{S-1} b3(s) \quad (3)$$

Вычислительная сложность алгоритма I3/V/N оценивается с помощью выражения

$$C_{I/3B/N} = YX + 4S + \sum_{s=0}^{S-1} b1(s) + \sum_{s=0}^{S-1} b2(s) + \sum_{s=0}^{S-1} b3(s) \quad (4)$$

Выражение (4) учитывает YX операций на буферизацию и формирование таблицы I/N длин серий размером S записей, S операций на поиск максимального значения по I , S операций на поиск максимального значения по N , $2S + \sum_{s=0}^{S-1} b1(s) + \sum_{s=0}^{S-1} b2(s) + \sum_{s=0}^{S-1} b3(s)$ операций на кодирование и передачу.

Другие модификации алгоритма кодирования длин

Для сжатия полутоновых изображений кроме алгоритмов I2/V/N и I3/V/N разработаны другие модифицированные алгоритмы кодирования длин серий, основанные на учете вероятности повтора значений пикселей в строках, отличающиеся использованием различного числа дополнительных символов для кодирования повторов символов (табл. 3).

Табл. 3. Модифицированные алгоритмы кодирования длин серий

№	Алгоритм	Особенность структуры закодированных данных
1	I/S/N	переменный размер поля длины серии
2	I/V/N	дополнительный символ повтора
3	I/V/S/N	дополнительный символ повтора и переменный размер поля длины серии
4	I/2B/N	два дополнительных символа повтора
5	I/2B/S/N	два дополнительных символа повтора и переменный размер поля длины серии
6	I/3B/N	три дополнительных символа повтора
7	I/3B/S/N	три дополнительных символа повтора и переменный размер поля длины серии
8	2I/N	повтор кодируемого символа
9	2I/S/N	повтор кодируемого символа и переменный размер поля длины серии
10	2I/V/N	повтор кодируемого символа и дополнительный символ повтора
11	2I/V/S/N	повтор кодируемого символа, дополнительный символ повтора и переменный размер поля длины серии
12	2I/2B/N	повтор кодируемого символа и два дополнительных символа повтора
13	2I/2B/S/N	повтор кодируемого символа, два дополнительных символа повтора и переменный размер поля длины серии
14	2I/V/2N(L/R)	повтор кодируемого символа, дополнительный символа повтора и двухсегментное поле длины серии

Квантование изображений для сжатия с потерями в пространственной области

Методы сжатия изображений с потерями основаны на эффективном кодировании коэффициентов преобразования с их предварительным квантованием. Во многих методах именно квантователь вносит потери и определяет, в основном, коэффициент сжатия. Кодирование коэффициентов преобразования позволяет достичь наибольших коэффициентов сжатия за счет концентрации основной энергии в относительно небольшом числе значимых коэффициентов преобразования. Однако, само преобразование требует существенных вычислительных ресурсов и временных затрат. Поэтому актуальной задачей является разработка алгоритма сжатия изображений с потерями, использующего квантование значений пикселей изображения и их последующее эффективное кодирование.

Предлагается алгоритм построения двухпорогового квантования значений пикселей полутонового изображения, ориентированный на последующее эффективное кодирование длин серий. В алгоритме используется оценка разности значений центрального пикселя и соседних с ним пикселей в окрестности Мура по отношению к двум порогам Δ_B (определяет условия разделения значений яркости соседних пикселей по соседним уровням квантования) и Δ_S (определяет условия приведения значения пикселя к несоответствующему ему уровню квантования из-за локального характера отличия яркости этого пикселя от яркости окрестных пикселей), связанным соотношением

$$\Delta_B = K\Delta_S,$$

где $K \geq 1$ – балансный коэффициент, определяющий степень влияния значения второго порога на результаты квантования (если значение K стремиться к ∞ , то результаты квантования стремятся к результатам однопорогового алгоритма).

Если модуль минимальной разности значения центрального пикселя окрестности Мура со значениями квантованных левого и трех верхних пикселей меньше порога Δ_B , то значение центрального пикселя окрестности Мура меняется на значение ближайшего к нему по яркости квантованного пикселя. Если данное условие не выполняется, то для оценки используется порог Δ_S .

Определяется квантованный пиксель окрестности Мура (левый или один из трех верхних), ближайший по значению к центральному пикселю. Вычисляются модули разностей значений этого пикселя со всеми значениями окрестных пикселей в окрестности Мура. Если все эти разности меньше порога Δ_S , то центральному пикселю окрестности Мура присваивается значение ближайшего ему по значению квантованного

пикселя в окрестности Мура. Если данное условие не выполняется, то центральный пиксель окрестности Мура сохраняет свое значение.

Оценка эффективности вероятностного кодирования длин серий для сжатия полутоновых изображений без потерь

Для тестовых изображений, представленных на рис. 3, в табл. 4 приведены размеры кода R_x , полученные для алгоритма RLE, алгоритмов из табл. 3, а также алгоритма Хаффмана, архиваторов Zip и Rar, где X – обозначение алгоритма в табл. 3.

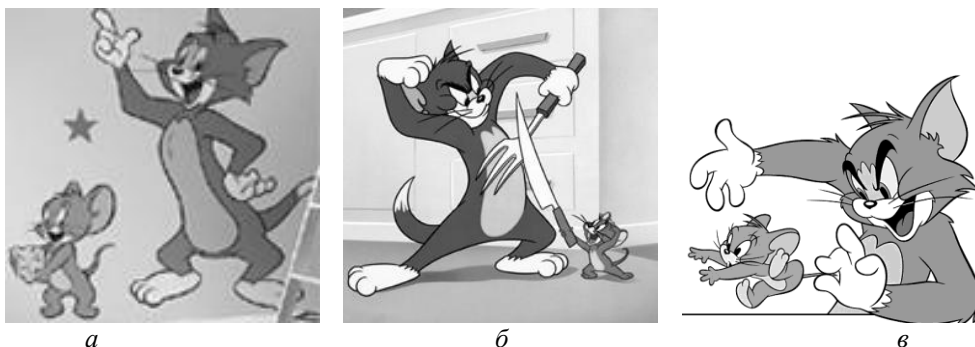


Рис. 3. Тестовые полутоновые изображения:

а – M1 (128x128 пикселей); б – M2 (256x256 пикселей); в – M3 (512x12 пикселей)

Из табл. 4 следует, что минимальный размер кода обеспечивают алгоритмы Rar и Zip. По сравнению с ними алгоритм I/3B/N показывает примерно на 26% худший результат, но превосходит на 57% и 8% алгоритмы RLE и Хаффмана соответственно. При этом алгоритмы RLE, I/2B/N, I/3B/N обеспечивают по сравнению с алгоритмами Rar и Zip уменьшение времени кодирования в 31,2 и 53,0 раз соответственно при размере изображения 256x256 пикселей, в 18,5 и 16,6 раз соответственно при размере изображения 1024x1024 пикселей, в 10,0 и 13,5 раз соответственно при размере изображения 8192x8192 пикселей (рис. 4).

Табл.4. Размеры кода при сжатии тестовых изображений без потерь

Алгоритм	Размер кода (байт) для изображений			Алгоритм	Размер кода (байт) для изображений		
	M1	M2	M3		M1	M2	M3
RLE	18620	77896	242604	2I/S/N	34484	135364	261921
I/S/N	18739	72412	143162	2I/B/N	32599	127870	222123
I/B/N	21504	91397	303235	2I/B/S/N	32826	128820	225640
I/B/S/N	20957	81407	165901	2I/2B/N	32713	128345	223002
I/2B/N	14694	57583	106871	2I/2B/S/N	32825	128753	225460
I/2B/S/N	17689	68649	118177	2I/B/2N(L/R)	30046	118028	208692
I/3B/N	14577	56701	103512	Zip	13070	44603	76470
I/3B/S/N	17687	68581	117997	Rar	12505	43062	75887
2I/N	30714	120377	208857	Хаффмана	14361	58281	113091

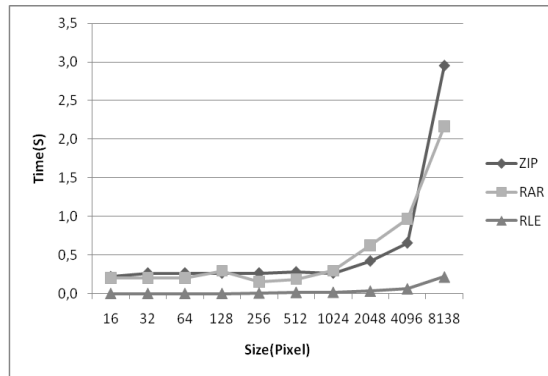


Рис. 4. Зависимости времени кодирования изображений от их размера для алгоритмов Zip, Rar, RLE и его предложенных модификаций в операционной системе Windows 7

Оценка эффективности вероятностного кодирования длин серий для сжатия полутоновых изображений с потерями

На рис. 5 приведены тестовые изображения, для которых, а также для тестовых изображений, приведенных на рис. 3,б и 3,в, произведен анализ эффективности алгоритмов кодирования RLE и его модификаций при сжатии с потерями по отношению к алгоритмам Zip, Rar, Хаффмана, JPEG и JPEG2000 (рис. 6).



Рис. 5. Тестовые полутоновые изображения:
а – Lena (256x256 пикселей); б – cameraman (256x256 пикселей)

Для сжатия с потерями на основе алгоритма RLE и его модификаций I2/B/N и I3/B/N, алгоритмов Zip, Rar и Хаффмана использован алгоритм двухпорогового построчного квантования изображений, рассмотренный выше. Как следует из рис. 6, предложенные алгоритмы вероятностного кодирования длин серий обеспечивают в 1,1 – 1,6 раза и 1,2 – 1,5 раза меньший коэффициент сжатия с потерями (в зависимости от типа изображения) по сравнению с алгоритмами Rar и Zip соответственно. При этом они превосходят базовый алгоритм RLE в 1,4 – 2 раза и алгоритм Хаффмана в 1,2 – 2,3 раза. При сжатии изображений в 2 – 5 раз предложенные алгоритмы обеспечивают уменьшение среднеквадратической ошибки в 3 – 10 раз в зависимости от типа изображения по сравнению с алгоритмом JPEG 2000. При одинаковой среднеквадратической ошибке

предложенные алгоритмы проигрывают алгоритму JPEG в коэффициенте сжатия в 1,2 – 1,8 раза.

В табл. 5 приведено время сжатия с потерями тестовых изображений с помощью различных алгоритмов. Из табл. 5 следует, что предложенные алгоритмы вероятностного кодирования длин серий обеспечивают уменьшение времени сжатия в 1,7 – 6 раз по сравнению с RLE; в 170 – 1750 раз по сравнению с RAR; в 230 – 2200 раз по сравнению с ZIP; в 50 – 400 раз по сравнению с JPEG; в 70 – 670 раз по сравнению с JPEG 2000.

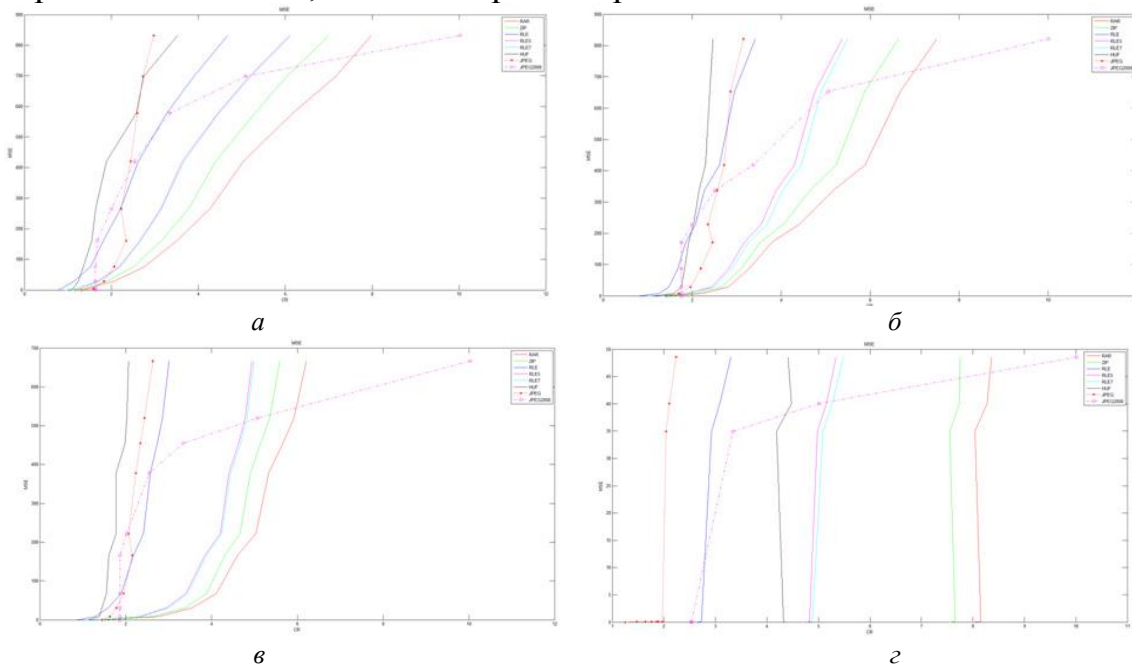


Рис. 6. Зависимости среднеквадратической ошибки сжатия тестовых полутоновых изображений от коэффициента сжатия:

а – для тестового изображения Lena; б – для тестового изображения cameraman;
в – для тестового изображения M2; г – для тестового изображения M3

Табл. 5. Время сжатия с потерями тестовых изображений

Алгоритм	Время сжатия изображений, с			
	M2	M3	Lena	cameraman
RLE	0.000000855	0.000000855	0.00000241	0.000000855
I2/B/N	0.000000285	0.000000285	0.00000144	0.000000285
I3/B/N	0.000000285	0.00000142	0.000000962	0.000000285
RAR	0.258	0.249	0.219	0.300
ZIP	0.300	0.324	0.322	0.682
JPEG	0.080	0.073	0.200	0.116
JPEG 2000	0.100	0.102	0.182	0.190

Заключение

Предложены алгоритмы вероятностного кодирования длин серий для сжатия полутоновых изображений без потерь и с потерями, отличающиеся от базового алгоритма RLE учетом вероятности повтора значений пикселей в строках изображения. Показано, что при сжатии без потерь предложенные алгоритмы обеспечивают уменьшение размера закодированных

изображений на 57% и 8% по сравнению с алгоритмами RLE и Хаффмана соответственно; уменьшение времени кодирования в 10–31 раз и в 13–53 раза при изменении размера изображений от 256x256 пикселей до 8192x8192 пикселей раз по сравнению с алгоритмами Rar и Zip соответственно. При сжатии с потерями предложенные алгоритмы обеспечивают повышение коэффициента сжатия изображений до 2 раз по сравнению с алгоритмом RLE, до 2,3 раза по сравнению с алгоритмом Хаффмана, уменьшение среднеквадратической ошибки до 10 раз по сравнению с алгоритмом JPEG 2000 при коэффициенте сжатия до 2 – 5 раз. Установлено, что предложенные алгоритмы обеспечивают уменьшение времени кодирования до 6 раз по сравнению с RLE, 1750 раз по сравнению с RAR, до 2200 раз по сравнению с ZIP, до 400 раз по сравнению с JPEG, до 670 раз по сравнению с JPEG 2000.

Список литературы

1. Pennebaker, W. B. JPEG Still Image Compression Standard / W. B. Pennebaker, J. L. Mitchell. – New York : Van Nostrand Reinhold, 1993. – 412 p.
2. Ebrahimi, T. JPEG2000 still image coding versus other standards / T. Ebrahimi [et al.] // Proc. of the SPIE, San Diego, CA, USA, July/August 2000. – Vol. 4115. – P. 446–454.
3. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео / Д. Ватолин [и др.]. – М: Диалог-МИФИ, 2003. – 384 с.
4. Golomb, S.W. Run-Length Encoding / S.W. Golomb // IEEE Transactions on Information Theory. – 1966. – July. – P. 399–401.
5. Миано, Дж. Форматы и алгоритмы сжатия изображений в действии / Дж. Миано. – М.: Изд. Триумф, 2003 – 336 с.
6. Сэломон, Д. Сжатие данных, изображений и звука / Д. Сэломон. – М: Техносфера, 2004. – 368 с.

Аль Бахдили Хасан, аспирант кафедры сетей и устройств телекоммуникаций факультета телекоммуникаций Белорусского государственного университета информатики и радиоэлектроники, Hassan.kassim@yahoo.com

Цветков Виктор Юрьевич, доцент кафедры сетей и устройств телекоммуникаций факультета телекоммуникаций Белорусского государственного университета информатики и радиоэлектроники, кандидат технических наук, доцент, vtsvet@bsuir.by

Конопелько Валерий Константинович, заведующий кафедрой сетей и устройств телекоммуникаций факультета телекоммуникаций Белорусского государственного университета информатики и радиоэлектроники, доктор технических наук, профессор, kafsiut@bsuir.by