УДК 519.722

DOI: 10.15827/0236-235X.118.282-285

Дата подачи статьи: 08.11.16

2017. Т. 30. № 2. С. 282–285

ЭФФЕКТИВНЫЙ МЕТОД БЛОЧНОГО КОДИРОВАНИЯ ДВУХУРОВНЕВЫХ ИЗОБРАЖЕНИЙ

М.П. Бакулина, к.ф.-м.н., научный сотрудник, marina@rav.sccc.ru

(Институт вычислительной математики и математической геофизики СО РАН,
просп. Академика Лаврентьева, 6, г. Новосибирск, 630090, Россия)

Эффективное сжатие изображений без потери качества – одна из важных задач теории информации, имеющая широкое практическое применение. Известно, что любое цифровое изображение можно представить последовательностью сообщений. Для выбора таких сообщений существуют многочисленные способы. Единственное требование к ним – обеспечение возможности восстановления точной копии исходного изображения из последовательности сообщений.

Один из способов выбора сообщений состоит в том, что смежные элементы изображения группируются в блоки, которые затем кодируются в соответствии с вероятностями их появления. При этом для наиболее вероятных конфигураций блоков используются короткие кодовые слова, а для менее вероятных – длинные, в результате чего в среднем достигается сжатие данных. Такой тип кодирования называется блочным.

Применяя метод блочного кодирования, можно добиться эффективного сжатия без потери качества двухуровневых (бинарных) изображений, где каждый элемент изображения может быть либо черным, либо белым. Решению этой задачи и посвящена данная статья.

Любое двухуровневое изображение рассматривается как множество примыкающих друг к другу прямоугольных блоков определенного размера. Эти блоки образуют совокупность сообщений, характеризующую изображение. Применяя к совокупности блоков оптимальный код Хаффмана, можно добиться наибольшего сжатия данных. Однако для блоков размером более 3×3 совокупность сообщений оказывается очень большой, и использование кода Хаффмана становится неэффективным. Кроме того, статистический анализ конфигураций блоков для двухуровневых изображений показывает, что блок, состоящий из белых элементов, имеет достаточно высокую вероятность.

Исходя из этого наблюдения и применяя известный оптимальный код, в данной работе предлагается эффективный двухэтапный метод блочного кодирования для двухуровневых изображений. Найдены оптимальные размеры блока, используемые на первом этапе кодирования, а также проведено сравнение экспериментальных результатов сжатия построенного алгоритма с результатами сжатия известного блочного алгоритма JPEG, которое подтвердило эффективность предложенного метода.

Ключевые слова: двухуровневое изображение, блочное кодирование, степень сжатия.

Эффективное сжатие изображений без потери качества – одна из важных задач теории информации, имеющая широкое практическое применение. Ее решению посвящен ряд исследований (например [1–3]). Любое цифровое изображение можно представить последовательностью сообщений. Для выбора таких сообщений существуют многочисленные способы. Единственное требование к ним – обеспечение возможности восстановления точной копии исходного изображения из последовательности сообщений. Один из способов выбора сообщений заключается в том, что смежные элементы изображения группируются в блоки размером $n \times m$, где n и m – число элементов в горизонтальном и вертикальном направлениях соответственно. Затем полученные блоки кодируются в соответствии с вероятностями их появления, причем для наиболее вероятных конфигураций блоков используются короткие кодовые слова, а для менее вероятных – длинные кодовые слова, в результате чего в среднем достигается сжатие данных. Такой тип кодирования называется блочным и рассматривается в работах [4, 5]. Блочное кодирование с использованием адаптивных методов рассмотрено в работах [6, 7].

Любое двухуровневое (бинарное) изображение мы будем рассматривать как совокупность примы-

кающих друг к другу прямоугольных блоков размером $n \times m$. Каждый элемент изображения может быть либо черным (1), либо белым (0), поэтому общее число конфигураций блоков, то есть число возможных расположений элементов в пределах блока, составляет 2^{nm} . Эти блоки образуют совокупность сообщений, которая характеризует изображение. Применяя к совокупности блоков оптимальный код Хаффмана [8], можно добиться наибольшего сжатия данных. Однако для блоков, размер которых более 3×3 , совокупность сообщений оказывается очень большой, и использование кода Хаффмана становится нецелесообразным.

В данной работе предлагается эффективный двухэтапный метод блочного кодирования для двухуровневых изображений. Найдены оптимальные размеры блока на первом этапе кодирования, а также проведено сравнение экспериментальных результатов сжатия построенного алгоритма с результатами сжатия блочного алгоритма JPEG [9], подтверждающее эффективность предложенного метода.

Метод блочного кодирования

Статистический анализ конфигураций блоков для двухуровневых изображений показывает, что

блок, состоящий из белых элементов, имеет достаточно высокую вероятность. Исходя из этого наблюдения и предложенного в [10] субоптимального кода, построим эффективный метод блочного кодирования. Кодирование будем осуществлять в два этапа.

Рассмотрим первый этап. Кодовое слово для блока, состоящего из одних нулей, будет 0. Кодовые слова для других конфигураций блоков образуются из nm двоичных разрядов, соответствующих данному блоку, которым предшествует префикс 1. Распределение вероятностей для источника, при котором этот код оптимален, равно

$$P(i, n, m) = \begin{cases} P(0, n, m) & \text{при } i = 1, \\ \frac{1 - P(0, n, m)}{2^{nm} - 1} & \text{при } i = 2, \dots, 2^{nm}, \end{cases} \quad (1)$$

где $P(0, n, m)$ – вероятность полностью белого блока.

Тогда средняя длина кодового слова определяется в виде

$$L = P(0, n, m) + (1 + nm)(1 - P(0, n, m)) = nm(1 - P(0, n, m)) + 1. \quad (2)$$

Пусть теперь $y_1 y_2 \dots y_l$ – последовательность, полученная после первого этапа кодирования, $y_i \in \{0, 1\}$. Рассмотрим второй этап кодирования, осуществляемый арифметическим кодом из [10]. Обозначим через $p = p(1)$ и $q = p(0)$. Выделим в этой

последовательности серии длины $l = \left\lfloor \frac{1}{\sqrt{p}} \right\rfloor$, кото-

рые следуют после появления 1, и особые символы $\underline{0}$ и $\underline{1}$, не входящие в блоки, то есть представим последовательность $y_1 y_2 \dots y_l$ в виде

$$\underline{0} \dots \underline{0} \underline{1} \underbrace{y_1 \dots y_l}_{i-1} \underline{0} \dots \underline{0} \underline{1} \underbrace{y_1 \dots y_l}_i.$$

Кодирование различных y_i осуществляется арифметическим кодом из [10] с помощью различных кодеров, настроенных на различные вероятности появления нулей и единиц, и может быть описано следующим образом. Особые символы $\underline{0}$ и $\underline{1}$ кодируются с помощью кодера K_0 с вероятностями q^l и $1 - q^l$ для $\underline{0}$ и $\underline{1}$ соответственно. Рассмотрим кодирование символов, находящихся внутри блока $y_1 \dots y_l$ длины l . Пусть $y_1 \dots y_{i-1} = \underbrace{0 \dots 0}_{i-1}$ ($i = 1, \dots, l$).

Тогда символ y_i , находящийся в i -й позиции после появления $i - 1$ нулей, кодируется с помощью кодера K_i с вероятностями π_i и $(1 - \pi_i)$ для $\underline{0}$ и $\underline{1}$ соответственно, где

$$\pi_i = \frac{1 - q}{1 - q^{l-i+1}} = \frac{1}{1 + q + q^2 + \dots + q^{l-i}}. \quad (3)$$

Появление этих вероятностей объяснено в [11]. Наконец, символы в блоке $y_1 \dots y_l$, следующие после появления в этом блоке 1, кодируются с помощью кодера $\bar{K}$ с исходными вероятностями q и p для 0 и 1 соответственно. Важно отметить, что ве-

роятности π_i не хранятся в памяти кодера и декодера, а вычисляются по следующей рекуррентной формуле:

$$\frac{1}{\pi_{i+1}} = \frac{1}{\pi_i} - q^{l-i}. \quad (4)$$

Следовательно, вычисление π_i , определяемых формулой (3), можно организовать по следующей схеме: $\bar{q} := \bar{q} / q$, $\bar{\pi}^{-1} := \bar{\pi}^{-1} - \bar{q}$ с начальными данными $\bar{q} := q^l$, $\bar{\pi}^{-1} := 1 + q + \dots + q^{l-1}$.

Таким же образом кодируется следующий блок, причем перед каждым новым блоком начальные данные обновляются. Для нахождения оптимальных размеров блока на первом этапе кодирования найдем теоретический коэффициент сжатия, полученный после первого этапа. Под коэффициентом сжатия C будем понимать отношение числа двоичных разрядов, необходимых для представления заданного изображения до кодирования, к числу двоичных разрядов после кодирования. Учитывая (2), получаем

$$C = \frac{nm}{\bar{L}} = \frac{nm}{nm(1 - P(0, n, m)) + 1}. \quad (5)$$

Ограничимся рассмотрением квадратных блоков $n \times n$. В таблице 1 приведены результаты зависимости коэффициента сжатия, полученные теоретически $C_{\text{теор}}$ и экспериментально $C_{\text{эксп}}$, от размера квадратного блока n для различных двухуровневых изображений A1–A5. Отметим, что для теоретических и экспериментальных результатов размер n был взят в интервале от 2 до 6. Это объясняется тем, что для $n > 6$ коэффициент сжатия начинает уменьшаться, поэтому дальнейшее увеличение размера блока становится нецелесообразным.

Таблица 1

Зависимость теоретических и экспериментальных результатов коэффициента сжатия от размера квадратного блока

Table 1

Dependence of theoretical and experimental results of a compression ratio on the size of a square block

Изображение	Коэффициент сжатия	Размер блока, n				
		2	3	4	5	6
A1	$C_{\text{теор}}$	3,01	5,22	6,91	7,15	6,87
	$C_{\text{эксп}}$	3,26	5,54	7,13	7,44	7,08
A2	$C_{\text{теор}}$	3,25	5,19	6,26	6,61	6,30
	$C_{\text{эксп}}$	3,15	5,06	6,11	6,54	6,19
A3	$C_{\text{теор}}$	3,21	5,15	5,68	5,60	5,24
	$C_{\text{эксп}}$	3,06	4,89	5,53	5,48	5,07
A4	$C_{\text{теор}}$	2,99	3,68	3,91	3,81	3,57
	$C_{\text{эксп}}$	2,83	3,56	3,81	3,66	3,48
A5	$C_{\text{теор}}$	2,52	3,23	3,31	3,14	2,95
	$C_{\text{эксп}}$	2,45	3,14	3,19	3,03	2,87

Из таблицы 1 видно, что наилучшие значения коэффициента сжатия дают размеры $n = 4$ и $n = 5$, которые и являются оптимальным размером блока.

Сравнение экспериментальных результатов сжатия

Для подтверждения эффективности предложенного метода было проведено сравнение экспериментальных результатов сжатия построенным алгоритмом с результатами наиболее распространенного и известного стандарта сжатия изображений – блочным методом JPEG. В качестве тестовых изображений были взяты рассмотренные выше двухуровневые изображения A1–A5. Сравнение проводилось по степени сжатия. Под степенью сжатия в данном случае понимаем количество бит, которым представляется в сжатом файле один байт (8 бит) исходного (несжатого) изображения. Результаты степени сжатия для предложенного метода k_{NEW} и для блочного метода JPEG – k_{JPEG} представлены в таблице 2.

Таблица 2

Результаты степеней сжатия различных двухуровневых изображений для предложенного алгоритма и известного блочного метода JPEG

Table 2

The results of a compression ratio of various binary images for the proposed algorithm and the famous JPEG decomposition method

Изображение	k_{NEW}	k_{JPEG}
A1	2,94	3,16
A2	4,58	4,81
A3	3,53	3,77
A4	4,91	5,14
A5	5,15	5,39

Из таблицы видно, что степень сжатия k_{NEW} примерно на 22–24 % лучше степени сжатия блочным методом JPEG, что подтверждает эффективность предложенного метода.

Выводы

Полученные экспериментальные данные показывают эффективность предложенного в данной работе метода блочного кодирования двухуровневых изображений: он дает сжатие таких изображений на 22–24 % лучше, чем известный и широко применяемый метод JPEG. Показано, что оптимальные размеры блока для предложенного метода – 4×4 или 5×5 . Построенный алгоритм кодирования может быть использован на практике для эффективного сжатия картографических и факсимильных изображений, спутниковых изображений земной поверхности и т.д.

Литература

1. Todd S., Langdon G.G., Rissanen J. Parameter reduction and context selection for compression of the gray-scale images. IBM Jour. Res. Develop, 1985, vol. 29, no. 2, pp. 188–193.
2. Howard P.G., Vitter J.S. Fast and efficient lossless image compression. Proc. IEEE Data Compression Conf. Snowbird, Utah, USA, 1993, pp. 351–360.
3. Babu P., Sathappan S. Efficient lossless image compression using modified hierarchical prediction and context adaptive coding. Indian Jour. of Sc. and Tech., 2015, vol. 8, pp. 1–6.
4. Tan D.S., Turner L.F. Sequential block interleave coding of two-tone facsimile data. Computers and Digital Techniques, 1988, vol. 38, pp. 95–107.
5. Horlander F.J. Incremental scanning for facsimile. IBM Tech. Disc. Bull, 1972, vol. 14, pp. 3311–3313.
6. Coulon F., Jonsen O. Adaptive block scheme for source coding of black-and-white facsimile. Electron. Lett., 1976, vol. 12, no. 3, pp. 61–62.
7. Bracamonte J., Ansorge M., Pellandini F. Adaptive block-size transform coding for image compression. IEEE Intern. Conf. on Acoustics, Speech, and Signal Processing, 1997, vol. 4, pp. 2721–2724.
8. Кормен Т.Х., Лейзерсон Ч.И., Ривест Л.Р., Штайн К. Алгоритмы: построение и анализ. М.: Вильямс, 2007. 459 с.
9. Weinberger M.J., Seroussi G., Sapiro G. The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS. IEEE Trans. Image Process, 2000, vol. 9, no. 8, pp. 1310–1324.
10. Ryabko B.Ya., Fionov A.N. Homophonic coding with logarithmic memory size. Algorithms and Computation, Springer, Berlin, 1997, pp. 253–262.
11. Рябко Б.Я., Шарова М.П. Быстрое кодирование низкоэнтропийных источников // Проблемы передачи информации. 1999. Т. 35. № 1. С. 49–60.

AN EFFICIENT BLOCK CODING METHOD FOR BINARY IMAGES

M.P. Bakulina¹, Ph.D (Physics and Mathematics), Research Associate, marina@rav.sssc.ru

Institute of Computational Mathematics and Mathematical Geophysics SB RAS, Academician Lavrentev Ave. 6, Novosibirsk, 630090, Russian Federation

Abstract. Efficient image compression without quality loss is one of important problems of the information theory. This problem has a wide practical application. It is known that any digital image can be represented by a sequence of messages. There are numerous ways to select messages. The only requirement for these methods is the possibility of restoring an exact copy of the original image from a sequence of messages.

One of the ways of choosing messages is that the adjacent picture elements are grouped into blocks. Then, these blocks are encoded according to the probabilities of their occurrence. Thus, short code words are used for the most probable configurations of blocks, and long code words are used for less probable configurations. The result is an average ratio of data compression. This coding is called block coding.

The block coding method makes it possible to obtain efficient compression two-level (binary) images without losing quality. This paper considers the solution of this problem.

We consider any binary image as many adjacent rectangular blocks of a certain size. These blocks form a set of messages, which characterizes the image. Using the optimal Huffman coding we can achieve the highest data compression. However, in blocks larger than 3×3 a set of messages is very large, and the Huffman code becomes inefficient. In addition, statistical analysis of two-level images shows that a block consisting of white elements has high probability.

Based on this observation and applying the known optimal code, this paper proposes an efficient two-step block coding method for binary images. We found the optimal block size at the first stage of coding. We have also compared the experimental results of the compression ratio with the proposed algorithm and the block algorithm JPEG. The results have confirmed the efficiency of the proposed method.

Keywords: binary image, block coding, compression ratio.

References

1. Todd S., Langdon G.G., Rissanen J. Parameter reduction and context selection for compression of the gray-scale images. *IBM Jour. Res. Develop.* 1985, vol. 29, no. 2, pp. 188–193.
2. Howard P.G., Vitter J.S. Fast and efficient lossless image compression. *Proc. IEEE Data Compression Conf.* Snowbird, Utah, 1993, pp. 351–360.
3. Babu P., Sathappan S. Efficient lossless image compression using modified hierarchical prediction and context adaptive coding. *Indian Jour. of Science and Technology.* 2015, vol. 8 (34), pp. 1–6.
4. Tan D.S., Turner L.F. Sequential block interleave coding of two-tone facsimile data. *Computers and Digital Tech-nique.* 1988, vol. 38, pp. 95–107.
5. Horlander F.J. Incremental scanning for facsimile. *IBM Tech. Disc. Bull.* 1972, vol. 14, pp. 3311–3313.
6. Coulon F., Jonsen O. Adaptive block scheme for source coding of black-and-white facsimile. *Electron. Lett.* 1976, vol. 12, no. 3, pp. 61–62.
7. Bracamonte J., Ansorge M., Pellandini F. Adaptive block-size transform coding for image compression. *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing.* 1997, vol. 4, pp. 2721–2724.
8. Cormen T.H., Leiserson Ch.E., Rivest R.L., Stein C. *Introduction to Algorithms.* 2nd ed., MIT Press & McGraw-Hill Publ., 2001 (Russ. ed.: Moscow, Vilyams Publ., 2007, 459 p.).
9. Weinberger M.J., Seroussi G., Sapiro G. The LOCO-I lossless image compression algorithm: Principles and standarti-zation into JPEG-LS. *IEEE Trans. Image Process.* 2000, vol. 9, no. 8, pp. 1310–1324.
10. Ryabko B.Ya., Fionov A.N. Homophonic coding with logarithmic memory size. *Algorithms and Computation.* Berlin, Springer Publ., 1997, pp. 253–262.
11. Ryabko B.Ya., Sharova (Bakulina) M.P. Fast coding of low-entropy sources. *Problemy peredachi informatsii* [Prob-lems of Information Transmission]. 1999, vol. 35, no. 1, pp. 49–60 (in Russ.).

Примеры библиографического описания статьи

1. Бакулина М.П. Эффективный метод блочного кодирования двухуровневых изображений // Программные продукты и системы. 2017. Т. 30. № 2. С. 282–285; DOI: 10.15827/0236-235X.118.282-285.
2. Bakulina M.P. An efficient block codig method for binary images. *Programmnye produkty i sistemy* [Software & Systems]. 2017, vol. 30, no. 2, pp. 282–285 (in Russ.); DOI: 10.15827/0236-235X.118.282-285.