

UDC 681.327

Victor Poriev

**IMPROVING THE METHOD  
OF RUN LENGTH ENCODING**

Віктор Порєв

**ВДОСКОНАЛЕННЯ МЕТОДА КОДУВАННЯ  
ДОВЖИН ПОВТОРЕНЬ**

The article analyzes modifications of the classical method of encoding repeat lengths for compression of raster data. As a result of the analysis, the usefulness of constructing optimal code sequences for fragments of raster images is noted. Proposals for organizing direct access have been made to accelerate the decoding of large-scale raster data fragments.

**Key words:** compression, prefix codes, run length encoding.

Fig.: 3. Tabl.: 0. Bibl.: 7.

В статті проаналізовані модифікації класичного метода кодування довжин повторень для компресії растрових даних. У результаті аналізу відзначається корисність побудування оптимальних кодових послідовностей для фрагментів растрових зображень. Зроблено пропозиції щодо організації прямого доступу для прискорення декодування фрагментів растрових даних великих обсягів.

**Ключові слова:** кодування довжин повторень, компресія, префіксні коди.

Рис.: 3. Табл.: 0. Бібл.: 7.

**Relevance of the research topic.** The growth in the amount of information processed by modern information systems prompts the search for effective forms of data storage and transmission. The improvement of methods of compression of information is actual.

**Formulation of the problem.** For information systems that store large volumes of information in raster formats, it is necessary to provide both a high degree of compression and a high decompression rate. To a large extent, these factors are contradictory.

**Actual scientific researches and issues analysis.** The method of compressing information based on encoding repeat lengths (RLE – *run length encoding*) has been known for a long time [1]. This is a very simple compression method, according to which each sequence of identical values is encoded by a pair (number of repetitions, values). This method has received wide popularity for recording images in a variety of file formats. The main known implementations of the RLE method are the PackBits method used in TIFF, TGA and others [2], as well as the version of the RLE method for the PCX file format [3].

Advantages of RLE:

- additional memory is not required (for example, for the dictionary)
- simplicity and high speed of unpacking (decoding)
- in simple implementations of the RLE method, the highest packaging speed is achieved
- the possibility of independent coding of individual lines, or other blocks, creates prerequisites for:
  - possibility of organizing quick direct access to any parts of the image
  - parallel (multi-thread) organization of coding-decoding

The disadvantage of known implementations of the RLE method is the small degree of compression. To increase the compression, some authors, for example [4], propose to combine the Huffman encoding with RLE [5].

In order to increase the RLE compression, authors [6, 7] proposed special prefix codes to represent the color values of raster images, as well as independent encoding of individual raster fragments for the optimal codes for these fragments. Such a version of the compression method is called RLE-БП. The developed adaptive encoder RLE-БП made it possible to increase compression by 1.5-2 times compared with the PackBits, PCX implementations while maintaining high decompression rates. This allowed it to compete with more powerful vocabulary LZ-like compression methods.

One of the advantages of the RLE method over vocabulary compression methods is that there is no need to accumulate a predefined decoding (dictionary). This allows you to encode independent raster fragments without losing compression that generates the ability to organize fast direct access to the desired parts of the image without decompressing the previous ones. Such opportunities are useful, in particular, in geographic information systems [6, 7].

**Uninvestigated parts of general matters defining.** The aspects of using RLE method together with methods of efficient bit sequence coding are not investigated.

**The research objective.** The main tasks are to search for encoding bit sequences for RLE that can provide both high compression rates and high decompression speed with minimal memory requirements.

**The statement of basic materials.** To estimate the number of bits needed to encode some image, we will write the following formula

$$Bits_{image} = Bits_{pixel} \times N_{pixels} ,$$

where:  $Bits_{pixel}$  - number of bits per pixel,  $N_{pixels}$  - number of pixels in the image. For example, for a 256-color image ( $Bits_{pixel} = 8$ ) of  $1000 \times 2000$  pixels ( $N_{pixels} = 2000000$ ), you need  $8 \times 2000000 = 16000000$  bits.

You can write such an estimate for RLE encoding

$$Bits_{RLE} = M_{bit\ single} \times N_{single} + M_{bit\ chains} \times N_{chains} ,$$

where:  $M_{bit\ single}$  - the number of bits needed to represent a single pixel,  $N_{single}$  - the

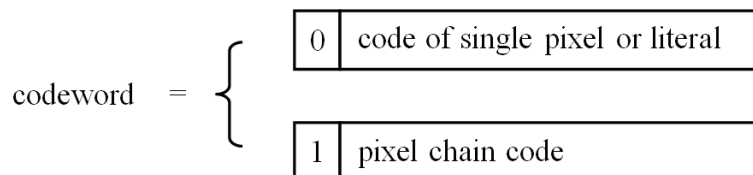
number of single pixels in the image,  $M_{\text{bit chains}}$  - the number of bits needed to represent the pixel chain,  $N_{\text{chains}}$  - the number of pixel chains in the image.

For compression it is necessary that  $Bits_{\text{RLE}} < Bits_{\text{image}}$ . The amount of compression is determined, firstly, by the number of chains of identical pixels in a particular image, and secondly by the coding of chains and single pixels.

The number of bits per  $M_{\text{bit chains}} = 16$  for the PackBits and PCX methods regardless of the particular image. For the RLE\_BII method, the value of  $M_{\text{bit chains}}$  can be substantially less than 16 and takes into account the features of a particular image.

The number of bits per single pixel depends on the image. For PCX and PackBits, the  $M_{\text{bit single}}$  value is in the range of 8 to 16. For the RLE\_BII method, the  $M_{\text{bit single}}$  value may be less than 8.

**General model structure.** In general, it seems that for all known varieties of the RLE method, the following coding structure is used. Bitstream contains codewords. The first bit 0 of each codeword of them means that the next is the single pixel code or the code of the set of unique pixels (literals). If the first bit = 1, then the code of the chain of the same pixels is contained.



**Fig. 1.** Codeword structure

The single pixel code can be a normal binary code of the color index, or may be in the form of a prefix code, such as Huffman. There may be a combination of prefix and common codes used in RLE\_BII, such as:

0 c . . c - m0 bits c

10 c . . c - m1 bits c

110 c . . c - m2 bits c

1110 c . . c - m3 bits c

1111 c . . c - m4 bits c

where: c . . c is the bits of the usual binary code of color indices. This RLE method improvement improves the compression of images that have many single pixels.

The pixel chain code must one way or another contain the chain length (n) and the pixel color index (s). For example, for the PackBits method, the pixel string code has the form: nnnnnnnccccccc, and for RLE\_BII it is usually the color of the bits first and then the bits of length. One of the moments in improving the code structure in RLE\_BII is the different number of bits of the chain code for different lengths of chains - for short lengths less bits, for longer ones. Some RLE\_BII submethods use different versions of prefix codes for different lengths of pixel strings.

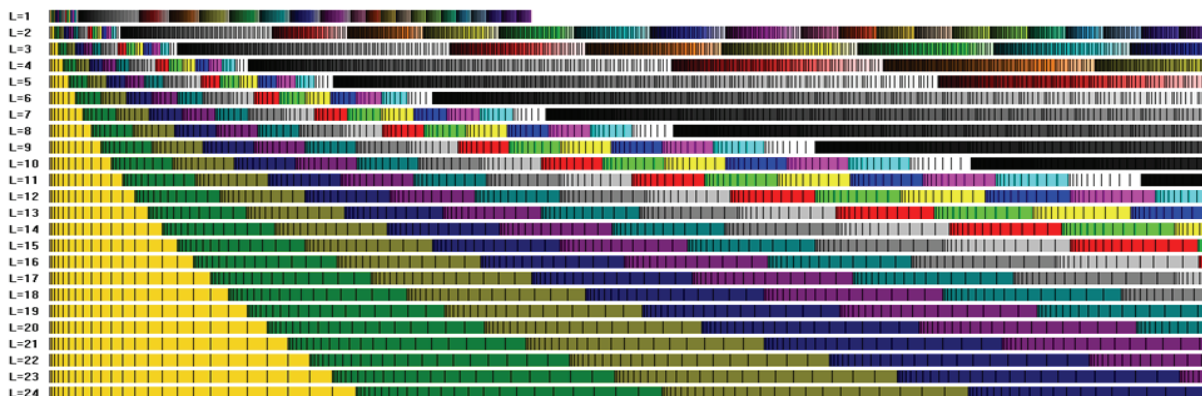
Another aspect of improving the RLE method is to control the length of the codewords using a plurality of encoding parameters. Such parameters can be, for

example, the number of bits of repeat length, the number of bits of color indices, types of code formats, etc. This allows each piece of the image to be written down by a code of minimum length, so that RLE\_БП provides the highest degree of compression among known implementations of the RLE method.

**Experiments.** You can give an example of a specific image of  $1125 \times 1115$ , in which 74% of single pixels, 26% of the pixels in the chains. For him, Bitsimage is 10,003,500. This image is poorly compressed by all known versions of the RLE method. PCX encoding gives 10,821,000 bits, that is, even more so than Bitsimage. The value of  $M_{\text{bit single}} = 9.43$ ,  $M_{\text{bit chains}} = 16$ . Encoding of PackBits gives 9710384 bits (compression is),  $M_{\text{bit single}} = 8.2$ ,  $M_{\text{bit chains}} = 16$ . The RLE\_БП method gives the greatest effect: 8590579 bits,  $M_{\text{bit single}} = 7.69$ ,  $M_{\text{bit chains}} = 9.93$ .

In the following example, the image is better compressed, because the single pixels have only 8.4% of it, and the rest in the chains. The dimensions are  $11680 \times 10,000$ , which means Bitsimage = 934.4 million bits. Encoding PCX gives 192.0 million bits,  $M_{\text{bit single}} = 10.32$ ,  $M_{\text{bit chains}} = 16$ . PackBits encoding gives 188.4 million bits,  $M_{\text{bit single}} = 9.66$ ,  $M_{\text{bit chains}} = 16$ . The RLE\_БП method here gives the biggest effect: 151.1 million bits,  $M_{\text{bit single}} = 8.37$ ,  $M_{\text{bit chains}} = 12.96$ .

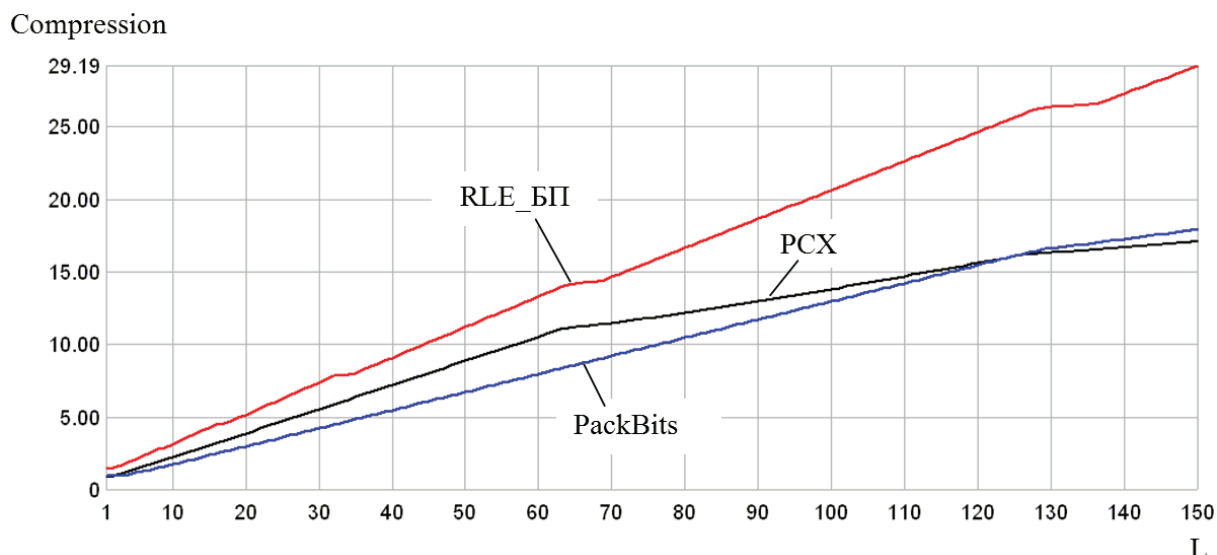
With regard to tests. For testing, you should use both real images and specially synthesized tests. As the simplest tests to assess the capabilities of RLE can recommend the image of vertical bands of different thicknesses of different colors. For example, the following tests:



*Fig. 2.* A set of tests of vertical stripes of different widths

Fig. 2 shows 20 test variants for different maximum bandwidths (L). In each test, the set of stripes of all 256 colors, each color strip is separated by a black vertical line. On these tests, different versions of RLE show a significantly different degree of compression (Fig. 3).

**Conclusions.** The problems of improving the method of RLE based on the methods of optimal encoding of bit sequences of individual fragments of raster images are investigated. The comparative testing of realization of the modified method with known versions of RLE implementations is carried out.



**Fig. 3.** The degree of compression in the tests for the width of the bands  $L = 1 \dots 150$

### References

1. S.W.Golomb. Run-Length Encodings, IEEE Trans. Information Theory, 12:3 (1966) pp. 399-401.
2. PackBits. From Wikipedia, the free encyclopedia. URL: <https://en.wikipedia.org/wiki/PackBits>.
3. PCX File Format Summary. FileFormat.Info. URL: <https://www.fileformat.info/format/pcx/egff.htm>.
4. Bayadir Abbas Al-Himyari. Role of Run Length Encoding on Increasing Huffman Effect in Text Compression // Journal of Kerbala University, 2008, Vol. 6 No.2 Scientific. pp.199-204.
5. D.A. Huffman. A Method for the Construction of Minimum-Redundancy Codes. Proceedings of the IRE. 40 (1952): 1098–1101. doi:10.1109/JRPROC.1952.273898.
6. Блинова Т.А., Порев В.Н. Некоторые способы кодирования растров в геоинформационных системах // Электронное моделирование. – 2008. – Т.30, №1. – С. 119-128
7. Blinova T., Porev V. Some Methods Of The Raster Encoding In Geographic Information Systems // Proc. int. conf. “CODATA`21”, Kyiv, 2008. – p.153.

### Autors

Victor Porev – associate professor, Department of Computer Engineering, National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”.

E-mail: [v\\_porev@ukr.net](mailto:v_porev@ukr.net)

Порев Віктор Миколайович – доцент, кафедра обчислювальної техніки, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського».

## РОЗШИРЕНА АНОТАЦІЯ

В. М. Порєв

### ВДОСКОНАЛЕННЯ МЕТОДА КОДУВАННЯ ДОВЖИН ПОВТОРЕНЬ

**Актуальність теми дослідження.** Зростання обсягів інформації, яку обробляють сучасні інформаційні системи спонукає шукати ефективні форми зберігання та передачі даних. Актуальним є вдосконалення методів компресії інформації.

**Постановка проблеми.** Для інформаційних систем, які зберігають великі обсяги інформації у растрових форматах, необхідно забезпечувати як високу ступінь компресії, так і високу швидкість декомпресії. Значною мірою ці фактори суперечливі.

**Аналіз останніх досліджень і публікацій.** Метод компресії інформації на основі кодування довжин повторень (RLE) відомий вже давно. Широку популярність цей метод отримав для запису зображень у різноманітних файлових форматах. Недоліком відомих реалізацій метода RLE є невелика компресія, проте інтерес до цього метода не зменшується. Дослідники шукають можливості його вдосконалити, щоб підвищити компресію і зберегти високу швидкодію.

**Виділення недосліджених частин загальної проблеми.** Недослідженими є аспекти використання метода RLE спільно зі способами ефективного кодування бітових послідовностей.

**Постановка завдання.** Основними завданнями є пошук процедур кодування бітових послідовностей для RLE, які здатні забезпечити одночасно і високу ступінь компресії і високу швидкість декомпресії при мінімальних вимогах до пам'яті.

**Викладення основного матеріалу.** Запропонованої формулу для оцінки основних властивостей метода RLE, яку використано для порівняння різних версій реалізації цього метода. Визначено загальну модель, яка охоплює відомі різновиди реалізації метода RLE і дозволяє проаналізувати можливі модифікації метода. Запропоновані способи модифікації метода на основі оптимального кодування бітових послідовностей RLE з використанням префіксних кодів.. Виконане порівняльне тестування відомих версій реалізацій RLE та запропонованою автором, яка отримала назву RLE\_БП.

**Висновки.** Досліджені питання вдосконалення метода RLE на основі способів оптимального кодування бітових послідовностей окремих фрагментів растрових зображень. Виконане порівняльне тестування реалізації модифікованого метода з відомими версіями реалізацій RLE.

**Ключові слова:** кодування довжин повторень, компресія, префіксні коди.