UDC 681.327

Victor Porev.

ADJUSTABLE RUN-LENGTH ENCODING AND ITS APPLICATION

The paper considers the method of lossless compression, which is a modification of the run len encoding. The proposed method allows to achieve a high rate of decompression of raster images with a competitive degree of compression due to the variativity of the construction of code sequences. An example of application of this method for distance on-line learning is given.

Key words: adaptive coding, compression, lossless, run length encoding.

Fig.: 1. Bibl.: 7.

Relevance of the research topic. Continuously increasing volumes of stored and transmitted information are a characteristic feature of the development of society. The improvement of methods of compression of information is actual.

Formulation of the problem. The main idea of the compression method under consideration is the ability to configure lossless compression parameters to achieve optimal characteristics. Optimization can be carried out taking into set of such conflicting factors as code length, compression speed, decompression speed, direct access speed to individual fragments of the unpacked array and other.

It should be noted that the ability to configure encoding parameters here does not mean adaptive coding in the generally accepted interpretation [1]. It so happened that adaptive coding was called a coding model that adapts to changes in information flow when it is encoded by a stream in real time. In other words, it is single pass compression. A prominent representative, the very known example of such methods of this is adaptive Huffman coding [2] As a rule, adaptive coding is mentioned in the context of lossless compression. Although, obviously, there is no ban for the discuss about adaptive coding for lossy compression.

In contrast to adaptive streaming coding, the methodology discussed in this article is called "adjustable" coding. Typically, this means having to complete two or more coding passes. Obviously, this does not allow compression in synchronization with the input data stream (or it makes it very difficult).

However, in addition to streaming coding tasks, there are many tasks for which using flexible adjusting coding can be useful.

Actual scientific researches and issues analysis. If you try to reduce the problem to choosing optimal parameters of some well-known methods and presenting it as something new, then it is difficult to achieve success. In this sense, this approach is trivial when discussing well-known lossy compression methods. Indeed, many lossy compression methods use some transformations that, as a rule, have control parameters, for example, the degree of loss. A vivid representative of this is the JPEG compression, based on discrete cosine transform. The basic compressing-decompressing scheme here is this: direct transformation - quantization - packaging - recording – reading - unpacking - dequantization - inverse transformation [3]. The coefficients of the quantization table are the main variable parameter, varying which you can change the compression properties over a wide range. Simplistically, the compression ratio parameter can be represented with just one number - the quantization factor by which the coefficients of the quantization table are divided. Another factor that may have an impact is the encoding methods of quantized quantities. Here, adjusting compression by choosing a coding method usually gives a smaller range of possibilities. A similar situation exists for some other known lossy compression methods, for example, JPEG2000 based on wavelet transform [4].

A substantially different situation takes place for the lossless compression. methods. We can say that the well-known lossless compression methods do not allow you to adjust the degree of compression simply by changing the quantization factor - this parameter is not at all in lossless compression. Often there are only fairly limited options for choosing code parameters. A fairly high level of ability to configure code parameters for the Huffman compression method. For each block of information, you can individually define a tree of code building [2]. Actually, this was the basis of adaptive coding. However, Huffman's compression level is low for applications such as image compression.

Among the lossless compression methods known today, LZ methods dictionary compression have the highest compression ratio. As an example, consider the LZW method [5,6,7]. The dictionary does not need to be transmitted along with compressed data. When decompressing, a dictionary is created automatically. This is very convenient for working in a data stream. Such an algorithm also shows high compression ratios for raster palette images, for example, 256-color ones. But there are also disadvantages to dictionary LZ methods. The main disadvantage can be called a low speed decompression. Especially if you need quick direct access to individual code fragments, for example, when viewing a fragment of a large raster image. To do this, unpack all the previous code. To some extent, the properties of the process and the result of LZ compression can be modified by the choice of code length and dictionary size. But such "adjusting" does not significantly speed up the decompressing. A significant restriction on the volume of the dictionary will lead to a loss of advantage in compression rate.

Another widely known lossless compression method is Run-Length Encoding (RLE) [8]. As a rule, it is inferior to dictionary methods in compression rate, but provides a much higher decompression speed. This is especially noticeable when working with images. 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, as well as the version of the RLE method for the PCX file format [7].

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
  - fully parallel or multi-thread organization of coding-decoding

The disadvantage of known implementations of the RLE method is the small degree of compression. In order to increase the RLE compression, authors [9, 10] 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 [9, 10].

Uninvestigated parts of general matters defining. At present many aspects of using RLE method together with methods of efficient bit sequence coding are not investigated. In particular, the regulation of the encoder, the choice of parameter values for optimal compression of various raster images. In other words, the creation of an "adjustable", that is, essentially adaptive but multi-pass encoder.

The research objective. The main tasks are to search for optimal coding bit sequences for RLE, which can provide both a high compression ratio and a high decompression speed, preferably with minimal memory requirements. The solution to these problems seems possible on the basis of the proposed RLE-БП method, which allows to achieve high flexibility ("adjustability") of encoder.

The statement of basic materials. Here, the description of the RLE-БП method, first published in 2008 [9,10], will not be repeated in detail. This method has been used successfully since 2005 in a variety of software systems using image compression.

Further development of RLE-БП has taken place in recent years. In particular, the following version of direct access to individual fragments of compressed rasters was implemented in 2017 [11].

In 2008, some new methods for generating bit sequences for describing both repeated and single pixels were proposed by the authors [9]. The features of the proposed code sequences is the ability to modify codes over a wide range. So, for example, one of the packaging methods (method3) as part of the RLE-БП, has the form:

0c ... c - for single pixels of arbitrary color. First, the prefix (0), then *M* color bits.

1p ... pnn ... n - for chains of pixels of the same color.

First the prefix (1p ... p), then the bits (nn ... n) of the chain length.

Each *i*-th primary color corresponds to an individual number (*Ni*) of bits n.

Encoding prefix codes according to method 3, consider this example

0c ... c - single pixels;

10nn ... n - a chain of pixels of color 0 (*N*0 bits n);

110nn ... n - a chain of pixels of color 1 (*N*1 bits of n);

1110nn ... n - a chain of pixels of color 2 (*N*2 bits of n);

1111nn ... n - a chain of pixels of color 3 (*N*3 bits n)

In this example, the prefix has up to 4 bits, but the encoder can use prefixes of both shorter and longer lengths, depending on the situation. Thus, the parameters of this encoding method3 are the number of primary colors (*M*), the prefix length, and the set of *Ni* values of the chain code lengths.

In total, the RLE-БП describes 4 methods-varieties of a combination of encoding of repeated and single pixels. For optimal coding, it is required to choose not only the most suitable method, but also to find the optimal values of the parameters of each method. For example, to achieve the minimum length of the packed code. The encoder automatically searches for the optimal set of methods and the set of values of their parameters automatically for each specific encoded image (or other array).

The best result is obtained when the optimal parameters are found automatically individually for each line of the raster image. Then the total amount of packed code is the smallest. As a result, in terms of compression, such an encoder can outperform (although not by much) LZW compression. At the same time, the main advantage of RLE is preserved - decompression speed.

Putting theory into practice. Initially, the compression methods of RLE-БП and the corresponding software adjustable encoder were created for the Geographical Information System. But nowadays it is unexpectedly needed in another sphere of interest. Because of COVID-19, the on-line distance learning activity has greatly increased. The author made an electronic board that could leaf through a lot of packed slides for presentation, on which you still need to make notes and explanations during the lesson. Since video conferencing with many participants requires a lot of resources, a program has been made that spends these resources economically. First of all, this concerns the decompression speed of compressed images when they are scaled over a wide range - from showing on a larger scale a fragment of one slide to showing at the same time many slides in the mosaic form without using additional memory (Fig. 1).

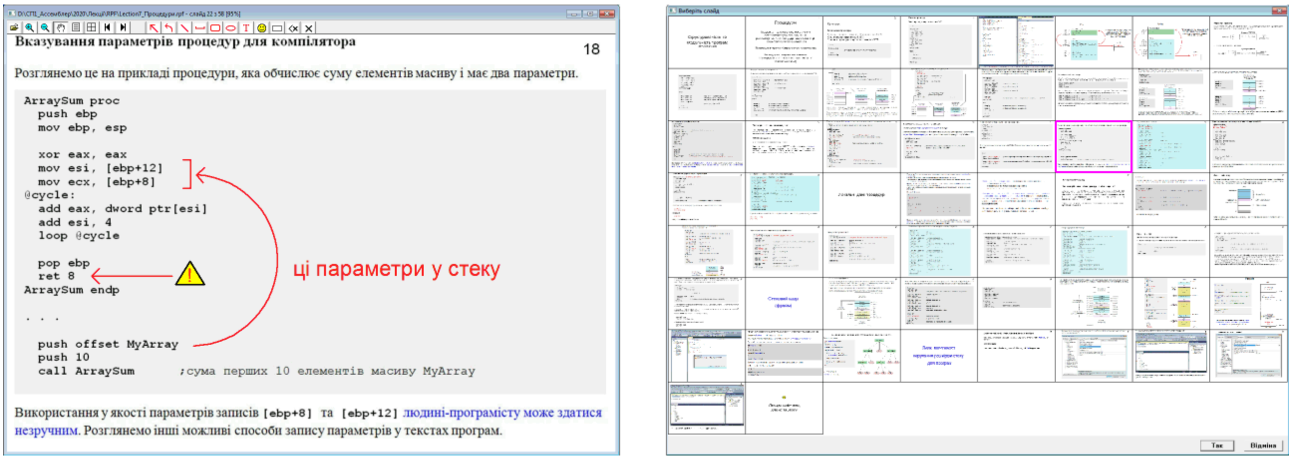

Fig. 1. Electronic board for the on-line distance learning which is based on the RLE-БП fast decompression software

Conclusions. The problems of using lossless compression methods in the presence of the requirement to ensure a high speed of decompression of raster images are investigated. It is proposed to use a modified run length encoding method RLE-БП, which allows for a competitive compression ratio at a high decompression rate. This is achieved by improving the code sequences with the ability to adjust encoding parameters over a wide range.

References

1. Adaptive coding. From Wikipedia, the free encyclopedia. URL: https://en.wikipedia.org/wiki/Adaptive_coding

2. 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.

3. Gregory K. Wallace. The JPEG Still Picture Compression Standard // IEEE Transactions on Consumer Electronics. Vol. 38, No 1, Feb 1992, pp. xviii-xxxiv

4. High Throughput JPEG 2000 (HTJ2K) and the JPH file format: a primer // ISO/IEC JTC 1/SC 29/WG1 | Document N87018.

5. URL: http://ds.jpeg.org/whitepapers/jpeg-htj2k-whitepaper.pdf

6. Ziv J., Lempel A. Compression of Individual Sequences via Variable-Rate Coding // IEEE Trans. Inform. Theory, 1978, V. 24 (5), pp. 530–536

7. Welch T. A Technique for High-Performance Data Compression // Computer, 1984, V. 17 (6), pp. 8–-19.

8. Ватолин Д., Ратушняк А., Смирнов М., Юкин В. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео. - М.: ДИАЛОГ-МИФИ, 2003. - 384 с. ISBN 5-86404-170-х

9. S.W.Golomb. Run-Length Encodings // IEEE Trans. Information Theory, 12:3 (1966) pp. 399-401.

10. Блинова Т.А., Порев В.Н. Некоторые способы кодирования растров в геоинформационных системах // Электронное моделирование. – 2008. – Т.30, №1. – С. 119-128

11. Blinova T., Porev V. Some Methods Of The Raster Encoding In Geographic Information Systems // Proc. int. conf. “CODATA`21”, Kyiv, 2008. – p.153.

12. Порєв В.М. Стосовно модифікації метода кодування довжин повторень // Вісник НТУУ ”КПІ”. Сер. Інформатика, управління та обчислювальна техніка. – К.: ВЕК +, 2017. № 65. С.82-89.

AUTHORS

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

E-mail: v_porev@ukr.net