

Interpolation for differential and hierarchical compression of multidimensional signals

S A Denisov¹ and M V Gashnikov¹

¹Samara National Research University, Moskovskoe Shosse 34A, Samara, Russia, 443086

Abstract. A comparative research of interpolation algorithms for hierarchical and differential methods of signal compression is performed. The hierarchical method of signal compression is based on hierarchical grid interpolation. The differential method of signal compression is based on differential pulse-code modulation (DPCM). Versions with maximum error control are used for both methods. Computational experiments are performed in natural test signals. The dependence of the compression ratio on the maximum error and on the standard error is given for both methods.

1. Introduction

To date, digital systems contain a huge amount of multidimensional information. First of all, these are hyperspectral data [1-2], the results of remote sensing [3], as well as video, charts, diagrams, drawings, photographs, etc. Such a variety of information requires a huge size of space to store this information on servers. Accordingly, to reduce the size of stored signals, effective algorithms for signal compression are required.

There is large number of algorithms for signal compression [4-11]. The compression method JPEG [8] is currently the most common method for signal compression. This method is based on a discrete cosine transformation [9]. The JPEG-2000 method [10] is more effective [11]. This method is based on wavelet. But this method is much less common.

Both of these methods are based on transformation the signal into a certain space of coefficients. This allows you to achieve high compression ratios. However, error control in the transformed space is difficult. If it is necessary to control the error, other methods of signal compression are promising. First of all, these are differential [4-5] and hierarchical [12-13] methods of signal compression. These methods allow you to control the maximum error [14].

In addition, the computational complexity of these methods is much less, since there is no transformation into a spectral space. These methods also have many other advantages (noise immunity, buffer memory management in real-time systems, etc.). Therefore, the task of investigating such compression methods is topical.

Differential and hierarchical compression methods decorrelate a signal in the same way. Transformation to a difference representation of the signal occurs. The difference signal is calculated as the difference between the original and interpolated samples of the signal. The difference between differential and hierarchical compression methods lies in interpolation algorithm. In differential methods, a progressive scan is used, and the signal samples are interpolated on the basis of previous samples. In hierarchical methods, the signal is interpolated based on more resampled versions of the same signal. The question of which interpolation algorithm is more effective requires research.

In this paper, computational experiments are performed comparing differential and hierarchical methods of signal compression with each other. The research is performed in the ordinates "error/compression ratio". The compression method of JPEG is used as a basis for comparison.

2. Differential compression of signals

Differential compression of signals is also called differential pulse-code modulation (DPCM) [4-5]. Here is a brief description of this compression method (see Figure 1). The multidimensional signal $\mathbf{X} = \{x(\vec{c})\}$ is processed in the order of some scan (\vec{c} is the vector of the signal's arguments).

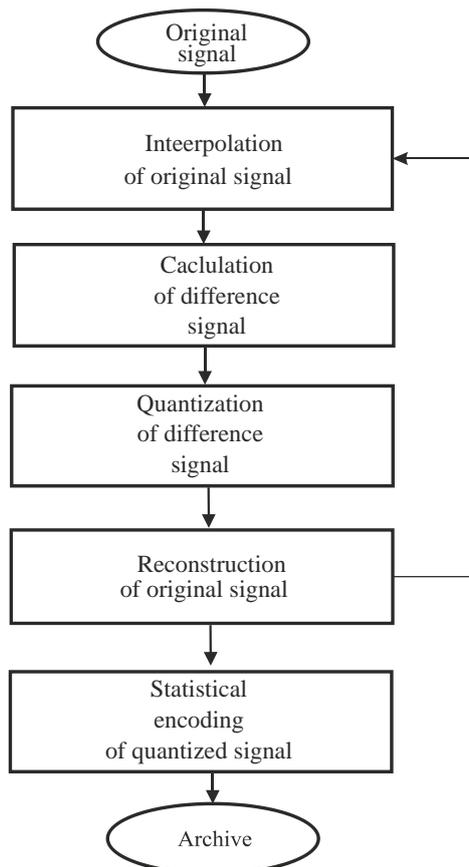


Figure 1. Differential compression of signal.

First, for each sample, an interpolated value is calculated

$$\hat{x}(\vec{c}) = I\left(\{\bar{x}(\vec{k})\}\right), \quad (1)$$

where $I(\cdot)$ is some interpolator, and $\{\bar{x}(\vec{k})\}$ are the previous (already processed) signal samples.

These samples have already been compressed and decompressed.

After interpolation, the sample of difference signal is calculated:

$$f(\vec{c}) = x(\vec{c}) - \hat{x}(\vec{c}). \quad (2)$$

If the original signal has a high correlation, the variance of the difference signal is much smaller than the variance of the original signal. This makes it possible to compress the difference signal much more than the original signal. To increase the compression ratio, the difference signal is quantized using the quantization function $Q(\cdot)$:

$$q(\vec{c}) = Q(f(\vec{c})). \quad (3)$$

The entropy [4] of the quantized signal $q(\vec{c})$ is much smaller than the entropy of the difference signal $f(\vec{c})$, so the quantized signal is compressed much more. In this paper, a uniform scale is used for quantization. In this situation, the actual quantizer can be written in the form:

$$q = Q(f) = \text{sign}(f) \left[\frac{e_{\max} + |f|}{2e_{\max} + 1} \right], \quad (4)$$

where [...] denotes the integer part of a number, and $\text{sign}(f)$ is calculated as follows:

$$\text{sign}(f) = \begin{cases} 1, & \text{если } f > 0 \\ 0, & \text{если } f = 0 \\ -1, & \text{если } f < 0 \end{cases}$$

The use of this quantizer allows us to control the maximum error e_{\max} [5] for each signal sample:

$$e_{\max} = \max_{\vec{c}} |f(\vec{c})| = \max_{\vec{c}} |x(\vec{c}) - \bar{x}(\vec{c})| \quad (5)$$

The quantized signal $q(\vec{c})$ is processed by a statistical encoder and stored in the archive.

After the quantization, the signal sample is restored:

$$\bar{x}(\vec{c}) = \hat{x}(\vec{c}) + (1 + 2e_{\max})q(\vec{c}), \quad (6)$$

where $\bar{x}(\vec{c})$ are the reconstructed signal samples. Exactly the same values will be obtained after decompression. With compression, these values are needed for interpolation (1) of subsequent samples. The use of reconstructed signal samples rather than the original signal samples for interpolation (1) makes it possible to ensure that the interpolated values are the same at the stages of compression and decompression.

3. Hierarchical compression of signals

Hierarchical compression of signals is based on a special hierarchical representation of the signal [12-13]. This representation of the signal is similar to a quad tree. A multidimensional signal $\mathbf{X} = \{x(\vec{c})\}$ is represented as a set of hierarchical levels \mathbf{X}_l (see also Figures 2-3):

$$\begin{aligned} \mathbf{X} &= \bigcup_{l=0}^{L-1} \mathbf{X}_l, \\ \mathbf{X}_{L-1} &= \{x_{L-1}(\vec{c})\}, \\ \mathbf{X}_l &= \{x_l(\vec{c})\} \setminus \{x_{l+1}(\vec{c})\}, \quad l < L-1, \end{aligned}$$

where L is the number of hierarchical levels \mathbf{X}_l , $\{x_l(\vec{c})\}$ is the signal resampled with step 2^l :

$$x_l(\vec{c}) = x(2^l \vec{c}) \quad (7)$$

This hierarchical representation of the signal makes possible a sequential compression of the hierarchical levels. We compress the hierarchical levels in the following sequence: \mathbf{X}_{L-1} , \mathbf{X}_{L-2} , ..., \mathbf{X}_2 , \mathbf{X}_1 , \mathbf{X}_0 . The hierarchical level \mathbf{X}_{L-1} is stored in the archive without any changes, since the data size of this level is very small. The samples of all other hierarchical levels are interpolated on the basis of samples of more resampled hierarchical levels:

$$\hat{x}_l(\vec{c}) = I \left(\bigcup_{k=l+1}^{L-1} \bar{\mathbf{X}}_k \right), \quad (8)$$

where $\hat{x}_l(\vec{c})$ are interpolating values, $I(\cdot)$ is interpolation function, and \bar{X}_k are the hierarchical levels that have already been compressed and decompressed.

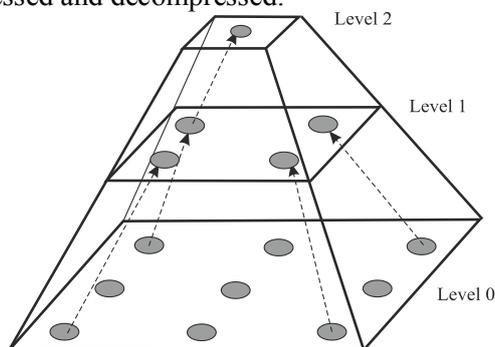


Figure 2. Hierarchical representation of a two-dimensional signal.

After the interpolation, the difference signal is calculated:

$$f_l(\vec{c}) = x_l(\vec{c}) - \hat{x}_l(\vec{c}) \quad (9)$$

Then this difference signal is quantized by means of some quantization function $Q(\cdot)$:

$$q_l(\vec{c}) = Q(f_l(\vec{c})) \quad (10)$$

The quantized signal $q_l(\vec{c})$ is processed by a statistical encoder and stored in an archive. The quantizer (4) with uniform scale is used in this paper, which allows us to control (5) the maximum error e_{\max} for each signal sample.

After quantization, the signal sample recovery is performed:

$$\bar{x}_l(\vec{c}) = \hat{x}_l(\vec{c}) + (1 + 2e_{\max})q_l(\vec{c}) \quad (11)$$

where $\bar{x}(\vec{c})$ are the reconstructed signal samples. These samples are subsequently used to interpolate (8) the samples of less resampled hierarchical levels of the signal.

3	1	2	1	3
1	1	1	1	1
2	1	2	1	2
1	1	1	1	1
3	1	2	1	3

Figure 3. Samples numbers of hierarchical levels of a two-dimensional signal (empty cells correspond to the hierarchical level number zero).

4. Interpolation of digital signals during compression

With differential and hierarchical compression, the already processed samples are used for interpolation. To reduce computational complexity, simple averaging functions are used in this case [4-5]. To simplify the explanation, we describe interpolation functions for the case of a two-dimensional signal $\mathbf{X} = \{x(\vec{c})\} = \{x(m, n)\}$.

During differential compression, the following interpolator is used in this work [15-17]:

$$\hat{x}(m, n) = \frac{1}{2}\bar{x}(m-1, n) - \frac{1}{2}\bar{x}(m-1, n-1) + \bar{x}(m, n-1)$$

With hierarchical compression, the following expression was used to interpolate samples with indices of the form $(2m+1, 2n+1)$:

$$\hat{x}_l(2m+1, 2n+1) = \left[\frac{1}{4} (\bar{x}_{l+1}(m, n) + \bar{x}_{l+1}(m+1, n) + \bar{x}_{l+1}(m, n+1) + \bar{x}_{l+1}(m+1, n+1) + 1) \right], \quad (12)$$

where $[\dots]$ denotes the integer part of the number, $\{x_l(m, n)\}$ is a signal resampled in 2^l times:

$$x_l(m, n) = x(2^l m, 2^l n) \quad (13)$$

Samples with indices $x_l(2m+1, 2n)$ and $x_l(2m, 2n+1)$ are interpolated as follows:

$$\hat{x}_l(2m+1, 2n) = \left[\frac{1}{2} (\bar{x}_{l+1}(m, n) + \bar{x}_{l+1}(m+1, n)) \right] \quad (14)$$

The interpolator (13-14) is shown in Figure 4a. Next to Figures 4b-4c two more hierarchical interpolators are shown in the same notation.

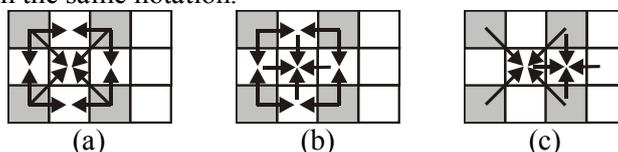


Figure 4. Interpolators with hierarchical compression.

The described averaging interpolators for hierarchical and differential compression are very simple and have low computational complexity.

5. Experimental research of interpolation algorithms of signals

The considered interpolators were investigated within the framework of the appropriate methods of signal compression. For this research, computational experiments were performed. Nature test signals were used in computational experiments. Some examples of decompressed test signals are shown in Figures 5-6.

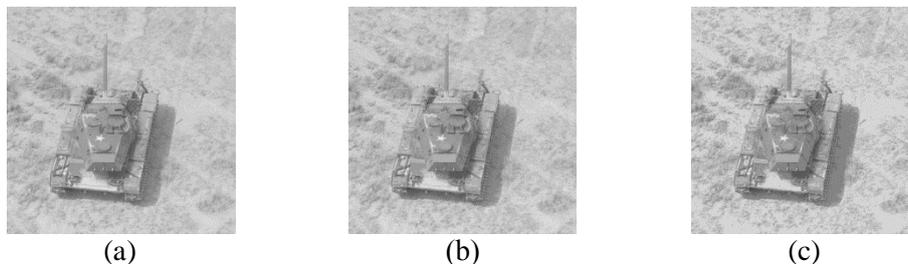


Figure 5. Decompressed signals for differential compression (the maximum error e_{\max} is 2, 10, 20).

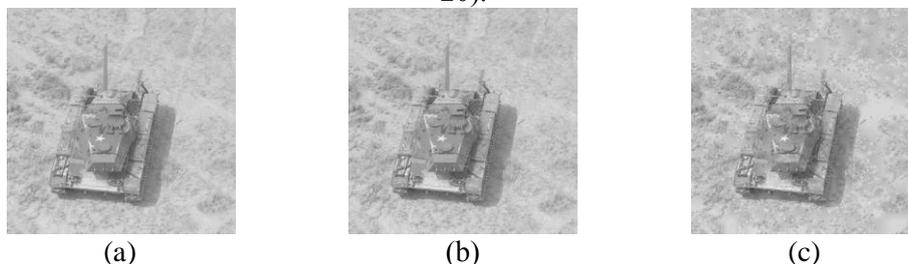


Figure 6. Decompressed signals for hierarchical compression (the maximum error e_{\max} is 2, 10, 20).

The algorithms were compared in the coordinates "error/compression ratio". As a basis for comparison, the JPEG method was used. In this case, the maximum (4) error e_{\max} was used as an error

between the original and decompressed signals. We also used the root mean square (RMS) error as a measure of this error:

$$e_{RMS} = \sqrt{\frac{1}{S} \sum_c (x(\vec{c}) - \bar{x}(\vec{c}))^2} \quad (15)$$

where S is the number of signal samples, $x(\vec{c})$ is the original signal, $\bar{x}(\vec{c})$ is the decompressed signal.

Typical results of computational experiments are shown in Figures 7-8. Based on the results of computational experiments, the following conclusions were made.

1. Compression ratio K of hierarchical compression and differential compression is greater than the compression ratio of the JPEG method with small RMS error. With large RMS error, the compression ratio of the JPEG method is better.
2. The maximum error e_{max} hierarchical compression is less than the maximum error of the differential compression and the maximum error of the JPEG method.
3. RMS error e_{RMS} of hierarchical compression is less than the RMS error of differential compression.

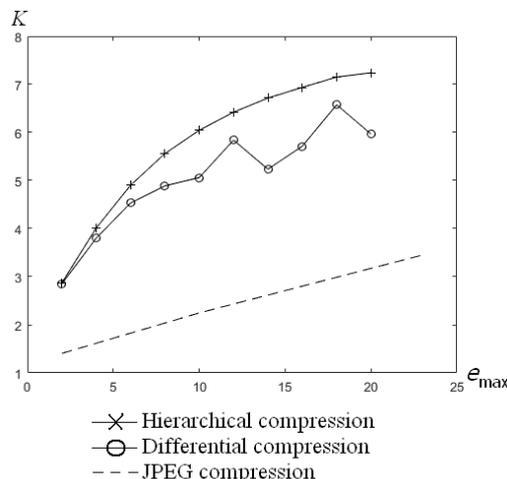


Figure 7. Dependences of the compression ratio K on the maximum error e_{max} .

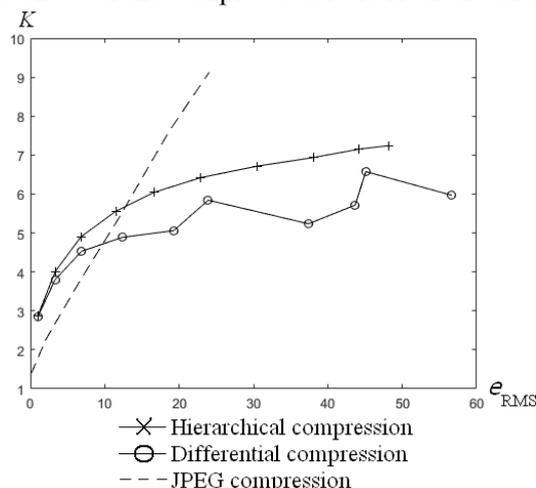


Figure 8. Dependences of the compression ratio K on the RMS error e_{RMS} .

6. Conclusion

Differential and hierarchical methods of multidimensional signals compression are considered. Algorithms of signals interpolation for differential and hierarchical signal compression are considered. Algorithms of interpolation and signal compression are implemented as software. Computational experiments were performed to investigate signal interpolation algorithms within the framework of

signal compression methods. Computational experiments were performed comparing hierarchical and differential methods of signal compression on a set of natural test signals. As a basis for comparison, the JPEG compression method was used. The results of the computational experiments are shown in the coordinates "error/compression ratio". As a measure of error, RMS and maximum errors are used. The conditions under which hierarchical compression has an advantage over differential compression and the JPEG method are described.

7. References

- [1] Chang C 2013 *Hyperspectral Data Processing: Algorithm Design and Analysis* (New York: Wiley Press) p 1164
- [2] Chang C 2007 *Hyperspectral data exploitation: theory and applications* (New York: Wiley-Interscience) p 440
- [3] Borengasser M 2004 *Hyperspectral Remote Sensing – Principles and Applications* (CRC Press) p 128
- [4] Sayood K 2012 *Introduction to Data Compression* (The Morgan Kaufmann Series in Multimedia Information and Systems) p 743
- [5] Salomon D 2007 *Data Compression. The Complete Reference* (Springer-Verlag) p 1118
- [6] Gupta V, Sharma A and Kumar A 2014 Enhanced Image Compression Using Wavelets *International Journal of Research in Engineering and Science (IJRES)* **2** 55-62
- [7] Woon W M, Ho A T S, Yu T, Tam S C, Tan S C and Yap L T 2000 Achieving high data compression of self-similar satellite images using fractal *Proc. of IEEE Int. Geoscience and Remote Sensing Symposium (IGARSS)* 609-611
- [8] Wallace G 1991 The JPEG Still Picture Compression Standard *Communications of the ACM* **34(4)** 30-44
- [9] Plonka G and Tasche M 2005 Fast and numerically stable algorithms for discrete cosine transforms *Linear Algebra and its Applications* **394(1)** 309-345
- [10] Li J 2003 Image Compression: The Mathematics of JPEG-2000 *Modern Signal Processing* **46** 185-221
- [11] Ebrahimi F, Chamik M and Winkler S 2004 JPEG vs. JPEG2000: An Objective Comparison of Image Encoding Quality *Proc. of SPIE Applications of Digital Image Processing XXVII* **5558** 300-308
- [12] Gashnikov M V, Glumov N I and Sergeyev V V Compression Method for Real-Time Systems of Remote Sensing *Proc. 15th Int. Conf. on Pattern Recognition* **3** 232-235
- [13] Gashnikov M V 2017 Minimizing the entropy of post-interpolation residuals for image compression based on hierarchical grid interpolation *Computer Optics* **41(2)** 266-275 DOI: 10.18287/2412-6179-2017-41-2-266-275
- [14] Lin S 2004 *Error Control Coding: Fundamentals and Applications, second edition* (New Jersey: Prentice-Hall) p 1260
- [15] Efimov V M and Kolesnikov A N 1997 Effectiveness estimation of the hierarchical and line-by-line lossless compression algorithms *Proc. of the III conf. Pattern Recognition and Image Analysis* **1** 157-161
- [16] Gashnikov M V 2016 Interpolation for hyperspectral images compression *CEUR Workshop Proc.* **1638** 327-333
- [17] Gashnikov M V and Glumov N I 2015 Hyperspectral images repository using a hierarchical compression *Posters proc. of 23 Int. Conf. on Computer Graphics, Visualization and Computer Vision* 1-4

Acknowledgments

This paper was funded by RFBR according to the research projects 18-01-00667, 18-07-01312.