

Parameter space dimension reduction of an adaptive interpolator during multidimensional signal differential compression

A I Maksimov¹, M V Gashnikov^{1,2}

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

²Image Processing Systems Institute of RAS - Branch of the FSRC "Crystallography and Photonics" RAS, Molodogvardejskaya street 151, Samara, Russia, 443001

e-mail: aleksei.maksimov.ssau@gmail.com

Abstract. We propose a new adaptive multidimensional signal interpolator for differential compression tasks. To increase the efficiency of interpolation, we optimize its parameters space by the minimum absolute interpolation error criterion. To reduce the complexity of interpolation optimization, we reduce the dimension of its parameter range. The correspondence between signal samples in a local neighbourhood is parameterized. Besides, we compare several methods for such parameterization. The developed adaptive interpolator is embedded in the differential compression method. Computational experiments on real multidimensional signals confirm that the use of the proposed interpolator can increase the compression ratio

1. Introduction

There are many interpolation [1, 2] methods have been already developed, and they continue to evolve. It is worth mentioning techniques based on context modeling [3], least mean squares [4], Kronecker bases [5], matrix pencil method [6], compressed sensing [7], etc. However, these methods are recourse consuming and computationally complex, therefore are not suitable for differential compression algorithms.

Differential compression of multidimensional signals, also called DPCM (differential pulse-code modulation [8 - 11]), is based on interpolation (prediction) of signal samples based on already processed samples and further interpolation error encoding (post-interpolation residues encoding). A high correlation usually characterizes real digital signals, so a transition to a differential representation entails a significant non-uniformity of the probability distribution of post-interpolation residues, which, in turn, leads to a decrease [12 - 14] of compressed data entropy, and a compression ratio increase.

In his paper, we propose a new adaptive DPCM interpolator of multidimensional signals [2, 15 – 16], for which three different ways of parameterization of already processed samples correspondence are proposed. The complexity of optimizing the parameters of the proposed interpolator is decreased by reducing the dimension of its parameters space. An experimental study of the proposed adaptive interpolator on a test set of multidimensional signals of the SpecTIR spectrometer was made; its results demonstrate that the proposed interpolation method outperforms the most common one.

The article is structured as follows: first, a general method of DPCM is given. After which the proposed interpolator is described: the general scheme for constructing adaptive interpolators, the procedure for reducing the parameter space and the interpolation procedure. After that, the results of an experimental study are presented.

2. Multidimensional signals differential compression

During differential compression of a multidimensional signal, the samples are processed in the order of some scan, which generalizes the line-by-line scan of the two-dimensional case. Let $C(\vec{x})$ be the multidimensional signal and \vec{x} be the vector of its arguments. Every sample of $C(\vec{x})$ is processed in the following way:

1. Interpolation.

The interpolated value $P(\vec{x})$ of the current sample is calculated using the already processed (compressed and decompressed) samples $\{C_k(\vec{x})\}$:

$$P(\vec{x}) = \mathbf{P}\left(\left\{\hat{C}_k(\vec{x})\right\}\right), \quad (1)$$

via interpolation function \mathbf{P} .

2. Calculation of the differential signal f (the difference between the interpolated and the real value):

$$f(\vec{x}) = C(\vec{x}) - P(\vec{x}). \quad (2)$$

where $f(\vec{x})$ - is the differential signal.

3. Quantization of differential signal:

$$f_q(\vec{x}) = Q(f(\vec{x})), \quad (3)$$

where $f_q(\vec{x})$ - quantized difference, Q - quantization function.

In this work, we used a uniform quantization scale:

$$f_q(\vec{x}) = \text{sign}(f(\vec{x})) \left\lfloor \frac{|f(\vec{x})| \varepsilon_{\max}}{2\varepsilon_{\max} + 1} \right\rfloor, \quad (4)$$

where ε_{\max} - preset maximum error (compression algorithm parameter), $\lfloor \dots \rfloor$ means an integral part of the number. This quantization scale controls the maximum error between the initial $C(\vec{x})$ and decompressed $\hat{C}(\vec{x})$ signals:

$$\varepsilon_{\max} = \max_{\vec{x}} \left| C(\vec{x}) - \hat{C}(\vec{x}) \right|. \quad (5)$$

4. Reconstruction of current value:

$$\hat{C}(\vec{x}) = P(\vec{x}) + f_q(\vec{x})(2\varepsilon_{\max} + 1). \quad (6)$$

i.e. calculation of decompressed value $\hat{C}(\vec{x})$. This value will be used in the compression stage during the interpolation (3) of the next samples. This feedback is necessary to ensure that the interpolators work identically during compression and decompression.

5. Encoding of the quantized differential signal. In this work, Huffman encoder was used [8 - 10].

3. Multidimensional signals adaptive interpolation

3.1. Approach to adaptive interpolator construction

We have developed proposed interpolator according to the following general structure of adaptive interpolation of multidimensional signals. It generalizes two-dimensional interpolation algorithms described in [2, 8].

Let us consider a current sample $C(\vec{x})$ which is to be interpolated on the basis of neighbor ones $\{C_k(\vec{x})\}$. Let $\{P_i(\{C_k(\vec{x})\})\}$ – be the set of simple and fast interpolation functions. Therefore, for every sample a set of interpolated values can be calculated:

$$P_i(\vec{x}) = P_i(\{C_k(\vec{x})\}). \quad (7)$$

Resulting interpolated value is chosen via the parameterized decision rule R :

$$P(\vec{x}) = C_i(\vec{x}), \quad i = R(\eta(\vec{x}), \eta^{\text{lim}}), \quad (8)$$

which uses the vector of local features $\eta(\vec{x})$ calculated on the basis of neighbor samples $\{C_k(\vec{x})\}$. The decision rule is also dependent on the parameter η^{lim} , which is calculated during the optimization of some criterion (for example, interpolation error). The certain form of the criterion is determined by the method application area.

Averaging over the nearest processed samples interpolators are usually used [2] to reduce the computational complexity in differential compression:

$$P(\vec{x}) = \frac{1}{N} \sum_{i=1}^N \hat{C}_k(\vec{x}), \quad (9)$$

where $\hat{C}_k(\vec{x})$ – are the neighbor samples, N – is the number of neighbor samples.

The interpolator of this type work quite accurately on relatively smooth signal parts due to averaging over noised samples, but it has a large error on the contours (extended brightness differences). When interpolating contour readings, interpolators based on interpolation “along” the contour work more accurately. An example of these interpolators for the two-dimensional case is, for example, Graham interpolator [2, 8].

During Graham interpolation, the interpolated value is equal to the neighbor sample value which lies in the direction of the contour. However, this interpolation is less accurate on smooth parts of the signal.

In this paper, we propose an adaptive interpolator that combines the advantages of both described approaches. Proposed interpolator automatically switches between averaging interpolation and interpolation “along the contour” depending on the presence and intensity of the contour in the local neighborhood of each signal sample.

Let us describe in more detail interpolation function (7) and decision rule (8). Let L be the number of possible contour directions (usually, it is greater than or equal to signal dimension). Let $\{\lambda_i(\vec{x}): 0 \leq i < L\}$ be the set of averaged difference moduli $|\hat{C}_i(\vec{x}) - \hat{C}_r(\vec{x})|$ between already processed samples $\hat{C}_k(\vec{x})$ in every direction possible (this set is computed in every signal sample). Difference values $\lambda_i(\vec{x})$ determine the presence and intensity of the contour in the current signal sample local neighbourhood. The least of these differences might be used to determine the direction of the contour if one passes through the processed sample. To decide whether there is a contour in a signal and what direction it goes in, several threshold values η_i^{lim} are used. Thresholds are compared with local features values $\eta(\vec{x})$. If there are no contours detected, averaging interpolation (9) is used:

$$P(\vec{x}) = P^{(1)}(\vec{x}) = \frac{1}{N} \sum_{k=1}^N \hat{C}_k(\vec{x}), \quad \text{if } i = \arg \min_{\lambda} \{\lambda_i(\vec{x})\} \ \& \ \eta(\vec{x}) \leq \eta_i^{\text{lim}}, \quad i \in [0, N_c). \quad (10)$$

Otherwise, the neighbouring sample, located in the differences minimum direction is used as an interpolating value (interpolation “along the contour” is performed):

$$P(\vec{x}) = P^{(2)}(\vec{x}) = \hat{C}_j(\vec{x}), \quad \text{if } i = \arg \min_{\lambda} \{\lambda_i(\vec{x})\} \ \& \ \eta(\vec{x}) > \eta_i^{\text{lim}}, \quad i \in [0, N_c). \quad (11)$$

Therefore, to calculate the best η_i^{lim} values, we have to perform the optimization in N_c -dimensional parameter space.

The application task determines the number of dimensions N_c of parameter space λ_i^{lim} . For differential compression, the complexity of the task is determined not only by the signal dimension but also by the number of contour directions taken into account. For example, for differential compression of two-dimensional signals (images), $N_c = 2$ if we consider horizontal and vertical contours $N_c = 4$ if we consider horizontal, vertical and diagonal contours. Given the limitations on computational complexity, even for $N_c = 2$ searching the η_i^{lim} parameters can be an overly time-consuming task.

3.2. Interpolator parameter space dimension reduction

In this paper, we propose to reduce the parameter space dimension to lower the computational complexity of finding these parameters. To do that we propose to use decision rules based on the relationships between $\lambda_i(\vec{x})$ instead of their absolute values. If there are no contours in a sample's neighborhood, differences λ_i will have close values. If there is a distinct contour, the desired λ_j will be not only the least but also distant from the other ones. Figure 1 shows described situations pictorially.

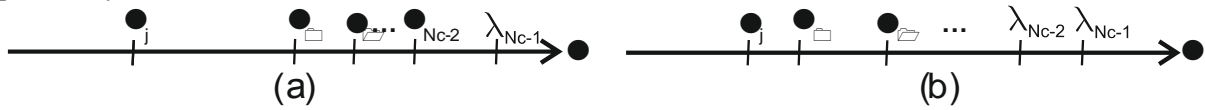


Figure 1. Differences distribution during the interpolation: (a) – a contour passes through the processed sample, (b) – there are no contours in the sample's neighbourhood.

We propose three new contour features based on described statements:

$$\eta_{\text{I}}(\vec{x}) = \lambda_r(\vec{x}) - \lambda_j(\vec{x}), \quad j(\vec{x}) = \arg \min_i \{\lambda_i(\vec{x})\}, \quad r(\vec{x}) = \arg \min_{i: i \neq j} \{\lambda_i(\vec{x})\}, \quad (12)$$

$$\eta_{\text{II}}(\vec{x}) = \frac{\lambda_j(\vec{x})}{\bar{\lambda}(\vec{x})}, \quad \bar{\lambda}(\vec{x}) = \frac{1}{N_c} \sum_{k=0}^{N_c-1} \lambda_k(\vec{x}), \quad (13)$$

$$\eta_{\text{III}}(\vec{x}) = \frac{\lambda_1^V(\vec{x}) - \lambda_0^V(\vec{x})}{\frac{1}{N_c - 1} \sum_{i=0}^{N_c-2} \lambda_{i+1}^V(\vec{x}) - \lambda_i^V(\vec{x})}, \quad (14)$$

where $\lambda_i^V(\vec{x})$ – is the element of the difference ordered series.

During the interpolation procedure, proposed contour features are compared with a single threshold value η^{lim} . Thus, by reducing the parameter space dimension, it is possible to reduce the multi-parameter optimization problem to a single-parameter one, where the only parameter is η^{lim} . To calculate this parameter, an automatic optimization procedure is used, similar to one described in [16].

3.3. Adaptive interpolation algorithm for differential compression

We considered a three-dimensional signal during experimental research $C(\vec{x}) = C(x, y, z)$, which is processed “layer-wise” with line-by-line scan in each layer. We have considered the following set of differences, each difference in accordance with a contour direction:

$$\lambda_1 = \lambda_- = \left(\left| \hat{C}(x, y-1, z) - \hat{C}(x-1, y-1, z) \right| + \left| \hat{C}(x-1, y-1, z) - \hat{C}(x-2, y-1, z) \right| + \left| \hat{C}(x-1, y+1, z) - \hat{C}(x-2, y+1, z) \right| \right) / 3 \quad (20)$$

$$\lambda_2 = \lambda_1 = \left(\left| \hat{C}(x-1, y-1, z) - \hat{C}(x-1, y-2, z) \right| + \left| \hat{C}(x-1, y, z) - \hat{C}(x-1, y-1, z) \right| + \left| \hat{C}(x-1, y+1, z) - \hat{C}(x-1, y, z) \right| \right) / 3 \quad (21)$$

$$\lambda_3 = \lambda_1 = \left(\left| \hat{C}(x, y-1, z) - \hat{C}(x-1, y, z) \right| + \left| \hat{C}(x-1, y, z) - \hat{C}(x-2, y-1, z) \right| + \left| \hat{C}(x-1, y-1, z) - \hat{C}(x-2, y, z) \right| \right) / 3, \quad (22)$$

$$\lambda_4 = \lambda_1 = \left(\left| \hat{C}(x, y-1, z) - \hat{C}(x-1, y-2, z) \right| + \left| \hat{C}(x-1, y, z) - \hat{C}(x-2, y-1, z) \right| + \left| \hat{C}(x+1, y-1, z) - \hat{C}(x, y-2, z) \right| \right) / 3, \quad (23)$$

$$\lambda_5 = \lambda_* = \left(\left| \hat{C}(x-1, y, z) - \hat{C}(x-1, y, z-1) \right| + \left| \hat{C}(x, y-1, z) - \hat{C}(x, y-1, z-1) \right| + \left| \hat{C}(x-1, y-1, z) - \hat{C}(x-1, y-1, z-1) \right| + \left| \hat{C}(x+1, y-1, z) - \hat{C}(x+1, y-1, z-1) \right| \right) / 4 \quad (24)$$

Figure 2 demonstrates the discussed differences.

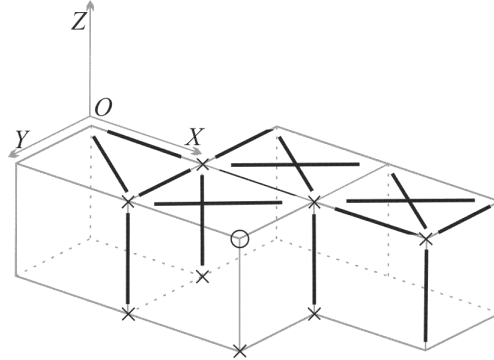


Figure 2. Discrete differences for five contour directions (bold lines) during sample «⊗» interpolation with the use of «×» samples as reference.

As one may notice $\lambda_-, \lambda_1, \lambda_2, \lambda_3, \lambda_4$ are the averaged differences of vertical, horizontal and diagonal directions inside the current layer $C(x, y, z)$, and λ_* is the “cross-layer” averaged difference, i.e., the difference between samples from layer $C(x, y, z)$ and layer $C(x-1, y, z)$.

In this case, contour features (12- 14) can be described in the following way:

$$\eta_I(x, y, z) = \lambda_r(x, y, z) - \lambda_j(x, y, z),$$

$$j(x, y, z) = \arg \min_i \{ \lambda_i(x, y, z) \}, \quad r(x, y, z) = \arg \min_{i: i \neq j} \{ \lambda_i(x, y, z) \}, \quad i, j \in [1, 5] \quad (25)$$

$$\eta_{II}(x, y, z) = \frac{\lambda_j(x, y, z)}{\bar{\lambda}(x, y, z)}, \quad \bar{\lambda}(x, y, z) = \frac{1}{5} \sum_{k=1}^5 \lambda_k(x, y, z), \quad (26)$$

$$\eta_{III}(x, y, z) = \frac{\lambda_1^V(x, y, z) - \lambda_0^V(x, y, z)}{\frac{1}{4} \sum_{i=1}^4 \lambda_{i+1}^V(x, y, z) - \lambda_i^V(x, y, z)}, \quad (27)$$

In this work, we have compared the adaptive interpolator with proposed contour features (25 - 27) with an averaging one (9). The averaging interpolator for this case can be described in the following way:

$$P(x, y, z) = \frac{1}{8} \sum_{i=1}^8 \hat{C}_k(x, y, z). \quad (28)$$

4. Experimental study of the proposed adaptive interpolator

A set of 6 hyper-spectral Earth remote sensing signals were used during the experimental research. The set was obtained via SpecTIR hyper-spectrometer. Signals are $160 \times 200 \times 356$ with 16-bit color depth. Figure 3 demonstrates several channels of test signals.

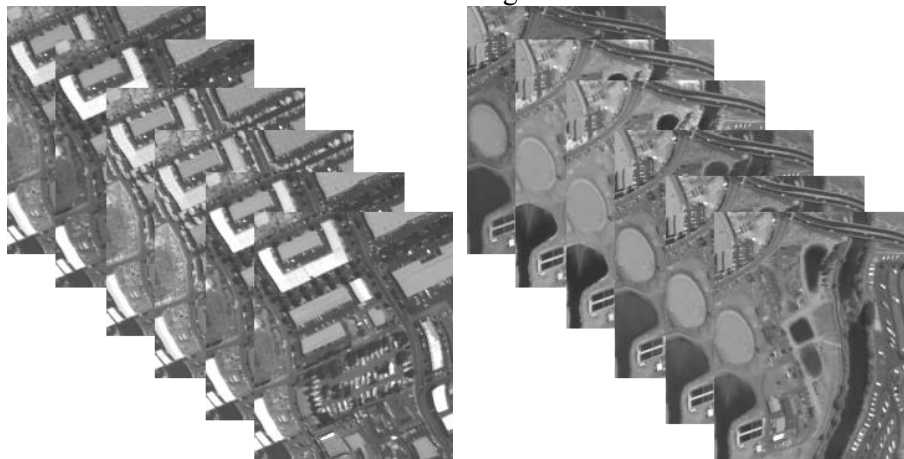


Figure 3. Several contrasted signal channels of the test set.

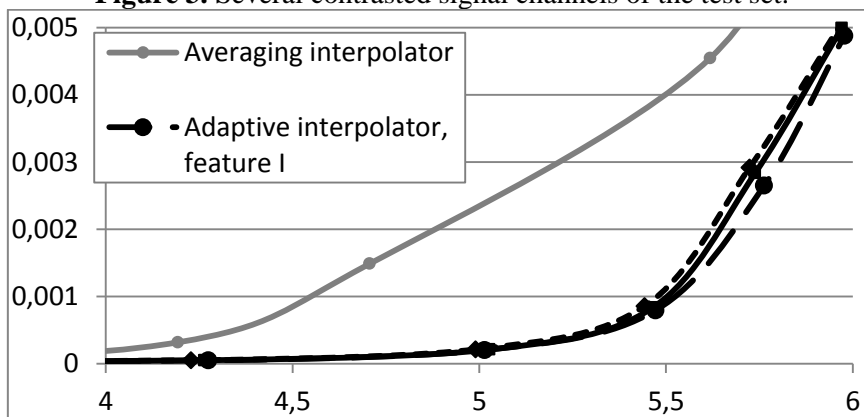


Figure 4. Dependence of decompressed signal RMS on compression coefficient with the use of the adaptive (with different contour features) and averaging interpolators.

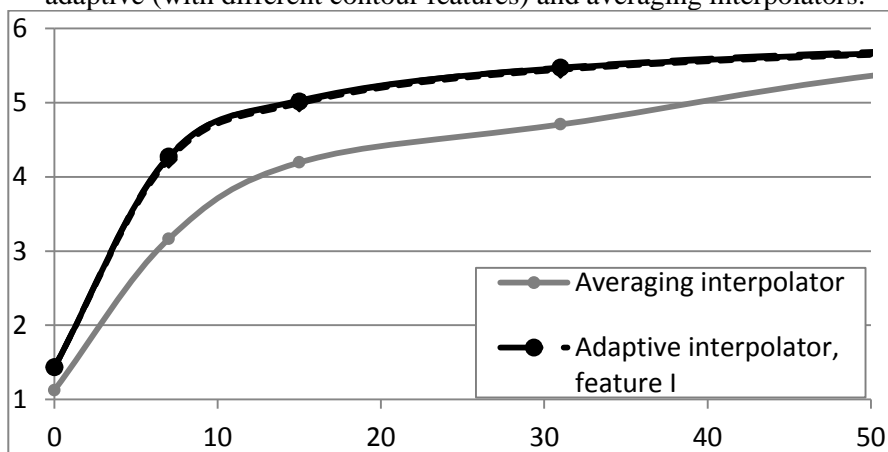


Figure 5. Dependence of the averaged over the test set compression ratio on the absolute error.

Experimental research was held in the following way - each test signal was compressed and then decompressed with a different maximum error value after that root mean square error (RMS) was

calculated for the decompressed signal. Results were averaged over all test set. Figure 4-7 demonstrates the results of the experimental study.

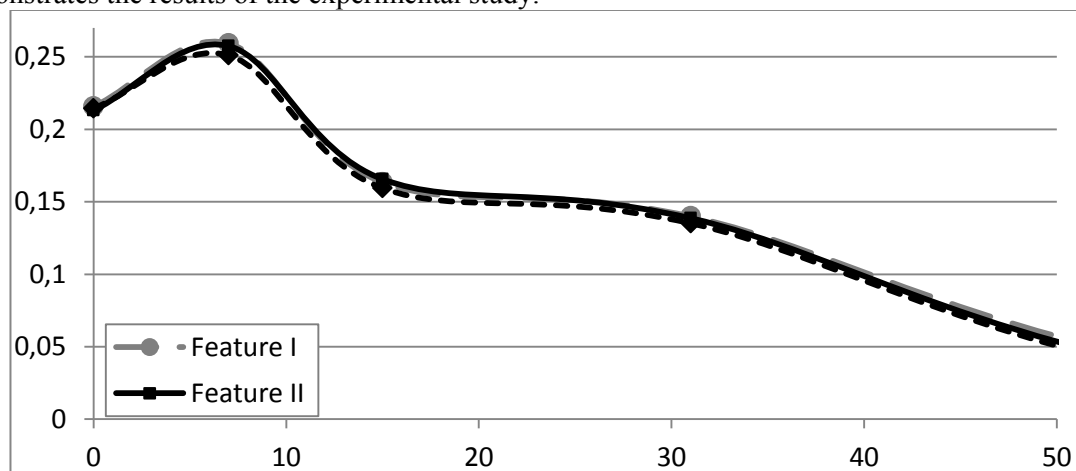


Figure 6. Dependence of the relative gain in archive size from the averaging interpolator on the absolute error.

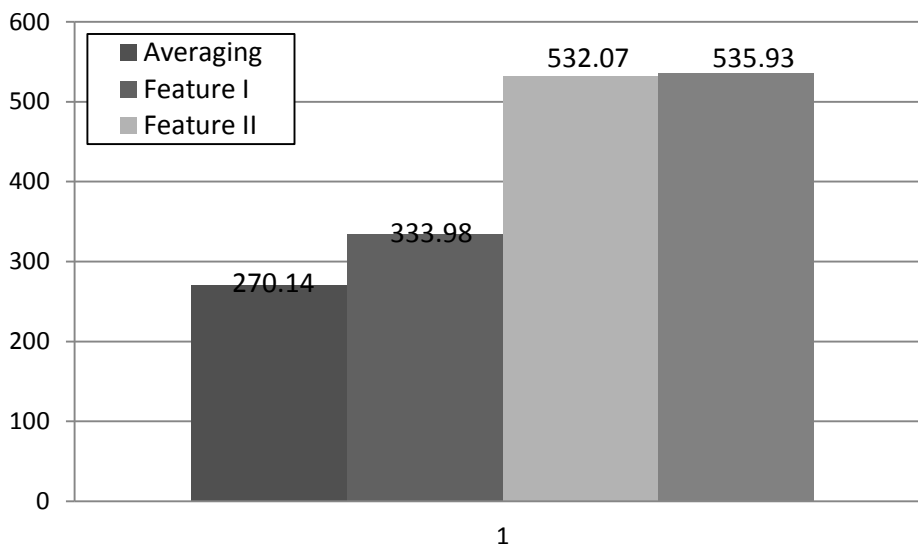


Figure 7. Averaged over the test set time of processing of one test image.

As one may notice, proposed adaptive interpolator significantly reduces decompressed signal RMS. for example, it gives 11 times less RMS then averaging interpolator for compression coefficient 5. The least RMS is obtained with the use of the contour feature that uses the difference between two minimum values (see 25 and its general form 12). One may notice, that the use of feature I (12, 25) results in faster image processing with the same performance (in terms of RMS and compression ratio) as features II (13, 26) and III (14, 27).

5. Conclusions

In this paper, new adaptive interpolator for differential compression tasks was proposed. It uses the contour features to switch between interpolation methods in every processed signal sample. The parameter space dimension reduction procedure was performed to lower the computational complexity of the proposed interpolator. Three methods for the contour feature calculation were proposed.

During the experimental research, every setup of the proposed interpolator outperformed the averaging one. The least RMS is obtained with the use of contour feature that uses the difference between two minimum values.

5. References

- [1] Sergeev V V and Chicheva M A 2013 *Digital processing of signals and images theory* (Samara: Samara University Press)
- [2] Gashnikov M V 2016 Parameterization of the nonlinear Graham predictor for digital image compression *Computer Optics* **40(2)** 225-231 DOI: 10.18287/2412-6179-2016-40-2-225-231
- [3] Gashnikov M V 2018 Interpolation based on context modeling for hierarchical compression of multidimensional signals *Computer Optics* **42(3)** 468-475 DOI: 10.18287/2412-6179-2018-42-3-468-475
- [4] Cohen A, Davenport M A and Leviatan D 2013 On the stability and accuracy of least squares approximations *Computational mathematics* **13** 819-834
- [5] Caiafa C F and Cichocki A 2016 Computing Sparse Representations of Multidimensional Signals Using Kronecker Bases *Neural Computation* **25** 186-220
- [6] Verstakov E V and Zakharchenko V D 2015 The comparative analysis of approximation algorithms of two-dimensional signals by Proni method and matrix bunches method *Radio engineering and telecommunication systems* **1** 26-31
- [7] Bigot J, Boyer C and Weiss P 2016 An analysis of block sampling strategies in compressed sensing *IEEE Transactions on Information Theory* **62** 2125-2139
- [8] Soifer V A et al. 2009 *Computer Image Processing, Part II: Methods and algorithms* (Saarbrücken: VDM Verlag)
- [9] Salomon D 2007 *Data Compression. The Complete Reference* (Berlin: Springer-Verlag)
- [10] Sayood K 2012 *Introduction to Data Compression.* (Burlington: The Morgan Kaufmann Series in Multimedia Information and Systems)
- [11] Vaseghi S V 2000 *Advanced Digital Signal Processing and Noise Reduction* (Hoboken: John Wiley & Sons)
- [12] Fursov V A 2011 *Information theory* (Samara: Samara University Press)
- [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] Woods J 2011 *Multidimensional Signal, Image, and Video Processing and Coding* (Cambridge: Academic Press)
- [15] Donoho D L 2006 Compressed sensing *IEEE Trans. Inform. Theory* **52** 1289-12306
- [16] Maksimov A I and Gashnikov M V 2018 Adaptive interpolation of multidimensional signals for differential compression *Computer Optics* **42(4)** 679-687 DOI: 10.18287/2412-6179-2018-42-4-679-687

Acknowledgments

The reported study was funded by:

- RFBR according to the research projects 18-01-00667, 18-07-01312;
- the RF Ministry of Science and Higher Education within the state project of FSRC “Crystallography and Photonics” RAS under agreement 007-GZ/Ch3363/26.