# Utilization of Tensor Decompositions for Video-compression

Sergey Matveev[1,2], Aleksandr Kurilovich[3]

[1] *Lomonosov Moscow State University, faculty of Computational Mathematics and Cybernetics, Leninskie Gory, MSU, 2-nd educational building, Moscow, 119889, Russia*

[2] *Institute of Numerical Mathematics RAS, Moscow, Russia 2, Gubkin st. 8Address, Moscow, 119333, Russia*

[3] *Center for Energy Science and Technology, Skolkovo Institute of Science and Technology, 121205, Moscow, Russia*

**Abstract**

In this work, we provide a study of video compression with the use of tensor train and Tucker decompositions. We measure the quality of compression with classical PSNR and SSIM metrics. Our approach allows us to control the quality of compressed video through the analytical evaluation of tensor decomposition ranks using the target value of PSNR. We achieve this aim because the PSNR is naturally related to the value of relative error in the Frobenius norm, which can be controlled for both tensor train and Tucker decompositions. In case of tensor train decomposition, we evaluate the idea of adding additional virtual dimensions and show that this trick allows us to improve the quality of compression without adding non-negligible additional errors. We discuss the advantages and visible artifacts introduced by the tensor-based algorithms to video compression and compare our results with industrial standards.

**Keywords**

video compression, tensor decomposition, tensor train, Tucker decomposition.

## 1. Introduction

Video compression is an important task in many multimedia applications due to the exponential growth in video data being generated nowadays [1]. In the research field, we seek to reduce the amount of data required to represent a video preserving its natural structure. Hence, one has to apply encoding to a video sequence and obtain a compressed format that can be efficiently used. Lossy video compression aims to achieve high compression ratios and minimize the distortion introduced in the video. Hence one has to find the optimal trade-off between the compression ratio (CR) and the distortion. It is typically measured within various quantitative quality metrics such as Peak Signal-to-Noise Ratio (PSNR) [2], Structured Similarity Index Measure (SSIM) [3], and Video Multi-Method Assessment Fusion (VMAF) [4].

Miscellaneous video compression standards have been introduced over the several last decades. E.g. there exist very popular standards based on “classical” techniques such as H.264/AVC [5],

*GraphiCon 2023: 33rd International Conference on Computer Graphics and Vision, September 19–21, 2023, V.A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences, Moscow, Russia*

matseralex@cs.msu.ru (S. Matveev); a.a.kurilovich@yandex.ru (A. Kurilovich)
0000-0001-8000-8595 (S. Matveev)

H.265/HEVC [6], and AV1 [1]. At the same time, we observe an extremely fast and promising growth of the deep learning-based methods [1]. These methods are mostly based on neural networks which are trained to achieve the compression process directly from data. For example, the VVC standard (Versatile Video Coding) [7] and Neural Video Compression (NVC) framework [8] seem to be well-developed. Unfortunately, the deep-learning approaches may suffer from adversarial attacks as well as from the extremely high computational cost of the training process. Hence, it seems that there is still space for complementary methods for video compression.

One of the alternative approaches could be built upon the various tensor decompositions such as tensor train (TT) and Tucker decompositions. Even though, there are quite many possible approaches for low-rank tensor decomposition such as canonical polyadic, tensor ring, and miscellaneous tensor networks (including the PEPS and MERA [19]). TT and Tucker formats seem to be especially interesting due to the existence of the computationally-effective algorithms st-HOSVD [13] and TTSVD [14] allowing to select analytically the accuracy of the constructed approximation. Both of these algorithms are based on the singular value decomposition (SVD). In this work, we apply the Tucker and TT formats for lossy video compression. We combine them with the standard preprocessing techniques in order to explain their advantages for this specific problem.

## 2. Problem setting

We propose the framework relying on TTSVD and st-HOSVD algorithms. In our work we use the open-source implementation `x264` of the h264 standard as a baseline. We concentrate on the following issues in our researh:

(1) Minimal preprocessing of the video data. We convert the video into `RGB24` format and split it into chunks of fixed length (30 frames per chunk) across the time dimension. Additionally, we use the virtual increase of dimensionality [12] for the TTSVD and investigate if it could improve the compression.

(2) Selection of the st-HOSVD and TTSVD ranks is based on the target value of the PSNR. The detailed description is provided in Section 3.

(3) We consider the optimization of the encoding speed out of the scope of this work even though many fruitful ideas and tricks can be developed in this direction.

The original video is stored in `yuv420p` format in the file system. Then it is converted to `RGB24` with the `opencv-python` wrapper for `ffmpeg`. Then it is compressed with some tensor decomposition approach (either TTSVD or st-HOSVD). The ranks are selected analytically with respect to the target PSNR for the `RGB24` representation of video. We use the `float32` data type in order to to reduce the size of the matrices/tensors instead of `float64`. We serialize the compressed video by the Python `pickle` module. We compute CR as the ratio of compressed and original video sizes in bytes. We use `ffmpeg` for the evaluation of the PSNR and SSIM metrics.

In this work, we use the well-known video-dataset from the xiph.org[2] website. We choose three videos with different temporal and spatial properties following [15]. Namely, we use *Crowd*

[1] Specification is available by link

[2] https://media.xiph.org/video/derf/

*Run* and *Red Kayak*) and also the black/white ultra-sound video `Cov_emdocs_vid2.avi` from the dataset supporting the paper [16].

Further, we discuss the procedure for analytical rank evaluation.

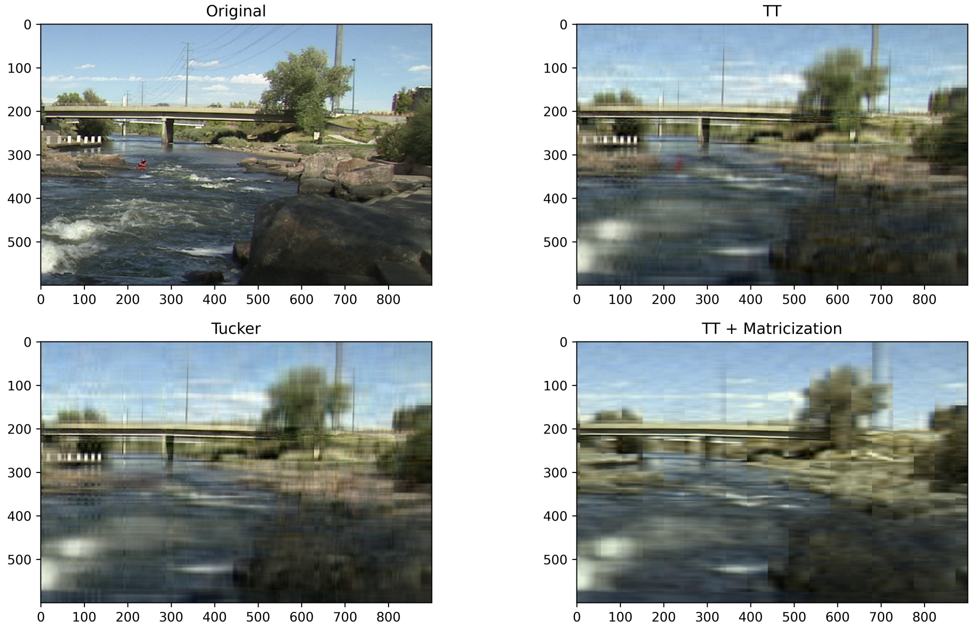

**Figure 1:** Compression artifacts. We set 291st frame of *Red Kayak* video and target PSNR of 25 dB for our compression algorithms. Row-wise, left-to-right frames correspond to the original video, videos restored from TT, Tucker, and TT (with preliminary matrixization) formats.

## 3. Analytical selection of the compression ranks

We refer to original sources of st-HOSVD [13] for the Tucker decomposition [9, 10]

$$A(i,j,k) = \sum_{\alpha_1=1}^{r_1}\sum_{\alpha_2=1}^{r_2}\sum_{\alpha_3=1}^{r_3} G(\alpha_1,\alpha_2,\alpha_3)\; U_1(i,\alpha_1)U_2(j,\alpha_2)U_3(k,\alpha_3). \tag{1}$$

for the description of its implementation and algorithmic complexity as well as for TTSVD [14] in case of tensor train [11] format

$$A(i,j,k) = \sum_{\alpha_1}^{r_1}\sum_{\alpha_2}^{r_2} G_1(i,\alpha_1)G_2(\alpha_1,j,\alpha_2)G_3(\alpha_2,k). \tag{2}$$

The complexity of st-HOSVD for $d$-dimensional array is $O(N^4 + N^3R + \min(N^2R^2, NR^4))$ operations, where $R$ is the maximal rank in the constructed decomposition and $N$ is its maximal mode size. Application of the TTSVD requires $O(N^4)$ operations. We implement the analytical rank selection procedure for both st-HOSVD and TTSVD as follows:

1. First of all, we use the fact that PSNR is closely related to approximation error in Frobenius norm:

$$PSNR = 10 \cdot \log_{10}\left(\frac{(2^n-1)^2}{MSE}\right) = -10 \cdot \log_{10}\left(\frac{\|\bar{A}-\bar{A}^*\|_F^2}{255^2 \cdot H \cdot W \cdot C \cdot T}\right) \tag{3}$$

where $\bar{A}$ - is the unfolded tensor representing the original video, $\bar{A}^*$ - is the unfolded tensor comprising the video decoded from the compressed format, $n$ - the number of bits for encoding the pixel intensity per pixel per channel ($n = 8$ for `RGB24`), $H$ - height, $W$ - width, $T$ - number of frames, and $C$ - number of color channels.

2. Then we compute the Frobenius norm of the residuals:

$$\|\bar{A}-\bar{A}^*\|_F = \sqrt{255^2 \cdot H \cdot W \cdot C \cdot T \cdot \exp\left(-\frac{PSNR}{10}\right)} \tag{4}$$

3. We select the ranks for each truncated singular value decomposition within the st-HOSVD or TTSVD algorithms. We apply SVD to the unfolded tensor $\bar{B}$ representing the compressed video at each stage of the algorithms. Further, we remind the well-known relation between the Frobenius norm and singular values getting the expression for the residuals:

$$\|\bar{B}-\bar{B}^*\|_F = \sqrt{\sum_{i=r_{tr}+1}^{R} \sigma_i^2} \tag{5}$$

Here, $\bar{B}^*$ is best approximation of matrix $\bar{B}$ with rank $r_{tr}$. In order to obtain the target PSNR, we select a maximum $r_{tr}$ obtaining the Frobenius norm of the residuals lower than the threshold recalculated from PSNR. Given the number $d$ of SVD applied within the st-HOSVD or TTSVD, we set the thresholds, which uniformly distribute the errors, among these operations. We suppose that the adaptive algorithms for error distribution may be applied, which is a matter of further research. Overall, in this work, we compute the ranks of tensor decompositions with the following rule:

$$r_{tr} = \max_{r_{tr}}\left[\sqrt{\sum_{i=r_{tr}+1}^{R} \sigma_i^2} > \sqrt{\frac{1}{d}} \cdot \|\bar{A}-\bar{A}^*\|_F\right] = \tag{6}$$

$$\max_{r_{tr}}\left[\sqrt{\sum_{i=r_{tr}+1}^{R} \sigma_i^2} > \sqrt{\frac{255^2}{d} \cdot H \cdot W \cdot C \cdot T \cdot \exp\left(-\frac{PSNR}{10}\right)}\right]$$

We apply it independently to each color channel if the virtual dimensions are not used. For the st-HOSVD, this means that $r_{tr} = 3$ during the utilization of the truncated SVD along the color mode of the tensor.

## 4. Results and Discussion

The video compression artifacts might significantly influence the visual perception of the processed data. They are especially visible at encoding with medium-low target PSNR (size/bitrate minimization). We concentrate on a low target PSNR of 25dB in order to show a visual explanation of the typical artifacts arising after the application of the st-HOSVD and TTSVD (with or without matrixization). An illustrative example is presented in Fig. 1.

For both TTSVD and st-HOSVD, we see artificial stripes and lines. They seem to have a non-local influence on the whole column/row (see Fig. 1, top-right and bottom-left images). The introduction of virtual dimensions along the space dimensions incorporates block structures (see Fig. 1 bottom-right image) and reduces the non-local errors. The blocks appear after the quantization of dimensions along the H-W modes. It means that we split each frame into blocks and stack them by the introduction of two additional modes of reshaping the initial tensor. Hence, two more SVDs have to be applied leading to different row-column basis sets for each block. It improves the visual perception of the compressed video.

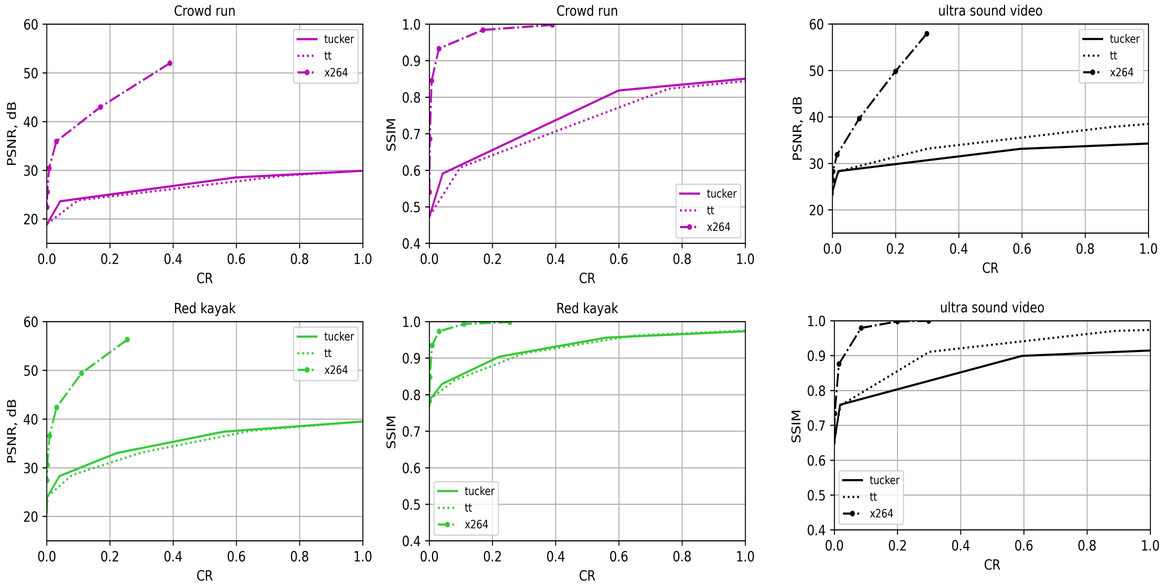

**Figure 2:** RD plots. Left column: PSNR-CR RD plots, right column: SSIM-CR RD plots. Figures on the 1st, 2nd, and 3rd rows show results for *Crowd run*, *Red kayak* and ultra-sound videos respectively. Dotted line: st-HOSVD, solid line: TTSVD, and dash-dotted line: `ffmpeg` (x264, medium preset, CRF). All quality metrics are calculated by `ffmpeg`.

We also perform the additional analysis for the RD plot in Fig. 2. The st-HOSVD allows to obtain better results than TTSVD for color videos but shows worse performance in case of black/white ultrasound video. In some sense, application of tensor decomposition to the compression of black/white video is simpler because we do not have to process the distinct color channels. On the other side, this ultra-sound video contains the natural noise itself. Such a noise causes an increase of the ranks of the corresponding unfolding matrices but for the target PSNR value of 30 dB we obtain an acceptable compression by 3 times and subjective visual quality of the compressed video. In this case, there might be an additional fruitful direction of future research devoted to the extraction of the low-rank based structures from noisy biomedical data [17].

However, the open-source implementation of the `h264` standard (`x264`) outperforms our approach in all three cases. It means that tensor formats might find applications within the video-processing, preconditioning of neural networks and completion [18] pipelines but not self-sufficient for the compression task. All in all, we apply the matrixization trick known as Quantized Tensor Train Decomposition [12]. We demonstrate these results in Fig. 3. For the TT format this trick allows us to small additional compression of the studied data and almost reach the RD for the st-HOSVD but not `x264`.

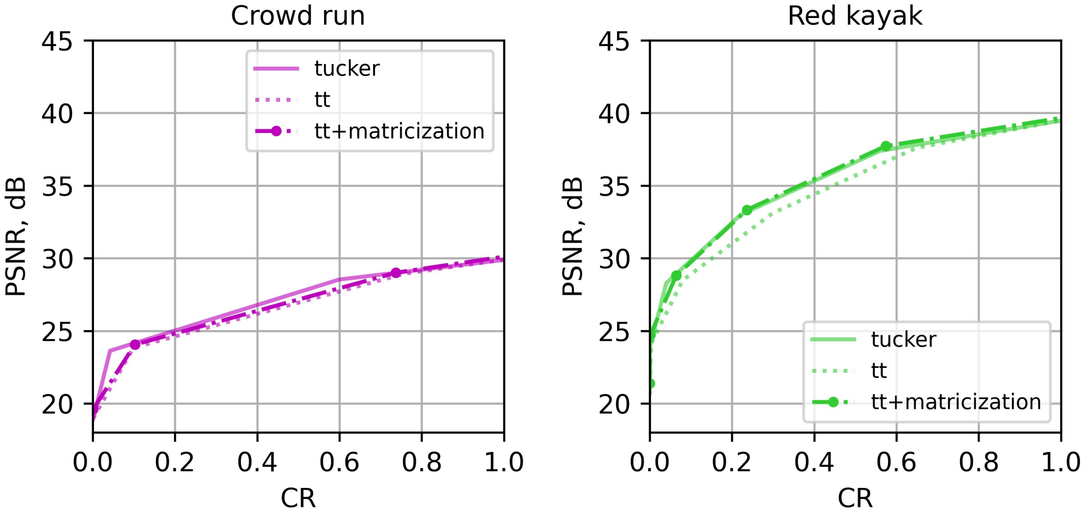

**Figure 3:** RD plots. PSNR-CR plots are shown left-to-right for *Crowd run* and *Red kayak* videos respectively. Dotted line: TTSVD, solid line: st-HOSVD, and dash-dotted line: TTSVD with video matrixization (30-frame video chunk with the shape (30, 1080, 1920, 3) is reshaped to the tensor of shape (30, 30, 36, 40, 48, 3)). All quality metrics are calculated by `ffmpeg`.

## 5. Conclusion

In this work, we have studied the opportunities of video data compression with the use of tensor train and Tucker tensor formats. We have tested the classical SVD-based methods for compressing of the selected videos with ranging spatiotemporal complexity. Even though the compression ratio does not reach state-of-the-art video compression standards, we consider this research useful due to the much more structured tensor train and Tucker formats. The applied tensor decompositions might become useful for preconditioning of the video processed by neural networks in order to decrease their inference time.

## 6. Acknowledgements

S. A. Matveev was supported by Russian Science Foundation (project 21-71-10072), A. A. Kurilovich acknowledges the VK MADE Big Data Academy [3]. Our codes are openly available at https://github.com/Lemm1ng/TensorVision.

## References

[1] Antsiferova, A., Lavrushkin, S., Smirnov, M., Gushchin, A., Vatolin, D., Kulikov, D., Video compression dataset and benchmark of learning-based video-quality metrics, arXiv preprint arXiv:2211.12109, 2022

[2] Huynh-Thu, Q., Ghanbari, M., Scope of validity of PSNR in image/video quality assessment, Electronics letters, **44**, 13 800–801, 2008

[3] Wang, Z., Bovik, A. C., Sheikh, H. R., Simoncelli, E. P, Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing, IEEE transactions on image processing, **13(4)**, 600–612, 2004

[4] Li, Z., Bampis, C., Novak, J., Aaron, A., Swanson, K., Moorthy, A., Cock, J. D., VMAF: The journey continues, Netflix Technology Blog. **25**, (1), 2018

[5] Sullivan, G. J., Topiwala, P. N., Luthra, A., The H. 264/AVC advanced video coding standard: Overview and introduction to the fidelity range extensions, Applications of Digital Image Processing XXVII, **5558**, 2004, 454–474

[6] Sullivan, G. J., Ohm, J. R., Han, W.J., Wiegand, T., Overview of the high efficiency video coding (HEVC) standard, IEEE Transactions on circuits and systems for video technology, **22** 12, 1649-1668, 2012

[7] Bross, B., Wang, Y. K., Ye, Y., Liu, S., Chen, J., Sullivan, G.J., Ohm, J.R., Overview of the versatile video coding (VVC) standard and its applications, IEEE Transactions on Circuits and Systems for Video Technology, **31**, (10), 3726–3764, 2021

[8] Liu, H., Chen, T., Lu, M., Shen, Q., Ma, Z., Neural video compression using spatio-temporal priors, arXiv:1902.07383, 2019

[9] Tucker, L.R., The extension of factor analysis to three-dimensional matrices, Contributions to mathematical psychology, 110119, 1964

[10] Tucker, L.R., Some mathematical notes on three-mode factor analysis, Psychometrika, **31**(3), 279–311, 1966

[11] Oseledets, I.V, Tensor-train decomposition. SIAM Journal on Scientific Computing **33**(5), 2295–2317 (2011)

[12] Oseledets, I.V., Approximation of matrices with logarithmic number of parameters. Doklady Mathematics **80**(2), 653–654 (2009)

[3] https://data.vk.company/

[13] Badeau, R., Boyer, R., Fast multilinear singular value decomposition for structured tensors. SIAM Journal on Matrix Analysis and Applications **30**(3), 1008–1021 (2008)

[14] Oseledets, I. V., Tyrtyshnikov, E.E., Breaking the curse of dimensionality, or how to use SVD in many dimensions. SIAM Journal on Scientific Computing **31**(5), 3744–3759 (2009)

[15] Zvezdakova, A.V., Kulikov, D.L., Zvezdakov, S.V., Vatolin, D.S., BSQ-rate: a new approach for video-codec performance comparison and drawbacks of current solutions, Programming and computer software, **46**, 183–194 2020

[16] Accelerating Detection of Lung Pathologies with Explainable Ultrasound Image Analysis, Born, J., Wiedemann, N., Cossio, M., Buhre, C., Brändle, G., Leidermann, K., Aujayeb, A., Moor, M., Rieck, B., Borgwardt, K. Applied Sciences, **11**, 2, 672, 2021

[17] Robust matrix completion with complex noise, Tang, L., Guan, W., Multimedia Tools and Applications, 2020, **79**, =2703–02717

[18] Ahmadi-Asl, S., Asante-Mensah, M. G., Cichocki, A., Phan, A. H., Oseledets, I., Wang, J., Fast Cross Tensor Approximation for Image and Video Completion, Signal Processing, 109121, 2023

[19] Evenbly, G., Vidal, G., Tensor network states and geometry, Journal of Statistical Physics, 2011, **145**, 891-918