Evaluation of Video Compression Methods for Network Transmission on Diverse Data: A Case Study

Roman Strukov
University of Texas at Arlington
Arlington, Texas, USA
roman.strukov@uta.edu

Vassilis Athitsos
University of Texas at Arlington
Arlington, Texas, USA
athitsos@uta.edu

ABSTRACT

There is a variety of video compression methods proposed in the literature. Choosing the best method for a specific type of video can involve a lot of experimentation and trial-and-error. This paper offers a quantitative evaluation of three popular compression methods, namely HEVC, MJPEG, MJPEG2000. Our evaluation uses five different types of video data that are commonly transmitted over a network. We use two evaluation metrics, namely PSNR and encoding time. While our evaluation is still preliminary, and will eventually be expanded to include more compression methods, larger datasets, and more metrics, it already includes features not encountered in evaluations that have appeared in prior literature. Even in its current form, to the best of our knowledge, our evaluation is the first one to evaluate both PSNR and encoding time, of both inter-frame and intra-frame codecs, on both natural and synthetic videos.

CCS CONCEPTS

• **Computing methodologies → Image compression**; • **General and reference → Evaluation**; **Metrics**.

KEYWORDS

video compression, compression ratio, peak signal-to-noise ratio, encoding time

ACM Reference Format:
Roman Strukov and Vassilis Athitsos. 2023. Evaluation of Video Compression Methods for Network Transmission on Diverse Data: A Case Study. In *Proceedings of the 16th International Conference on PErvasive Technologies Related to Assistive Environments (PETRA '23), July 05–07, 2023, Corfu, Greece.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3594806.3594863

1 INTRODUCTION

Transmitting and processing video information streams is a part of many important applications, in diverse fields such as telesurgery, remote rescue operations, video chats, online presentations, etc. High-quality video with minimal time delay is necessary for performing remote surgery or rescue operations. Live video presentations require the synchronization of a speaker and the audience with minimal possible latency. Surveillance systems must be able to provide the user with high quality images while minimizing, depending on the context, latency and/or storage requirements. Given a fixed transmission rate, the two defining properties of a video stream are the quality of the transmitted video and the latency, which is at least equal to the time it takes to encode the data.

The purpose of this paper is to offer an experimental evaluation of some popular codecs for video compression, on videos from five categories with significantly different characteristics. The five categories are natural videos, videos with a human face, cartoons, video games, recorded videos of computer desktop. This is a more diverse set of data compared to evaluations previously published in the literature. The results demonstrate that different codecs are advantageous for different categories.

It is important to have in mind that different application contexts place different demands on codecs. Encoding a movie for streaming is a task where non-real time performance is acceptable, as long as decoding can be done in real time and video quality is visually appealing. On the other hand, teleconferencing requires real-time performance both for the encoding and the decoding part. Reliable transmission channels permit the use of inter-frame compression, where knowledge of the previous video frame is used in reconstructing the current frame. Unreliable transmission channels, such as Internet-based teleconferencing and wireless communication among low-end devices, make inter-frame compression less appealing, because once a frame has been lost, all subsequent frames that are dependent on that one lost frame cannot be recovered. Therefore, in that case intra-frame compression is preferred. Our evaluation includes one inter-frame method, namely H.265 (HEVC), and two intra-frame methods, namely MJPEG and MJPEG2000.

2 RELATED WORK

Motion JPEG, MPEG and H.2x video codecs families are compared in [1] based on compression factor, system requirements, and bitrate. However, there is no comparison of video quality, and a single doorway scene video is used. H.264 achieves three times smaller bitrate than MPEG-4 and six time smaller bitrate than MJPEG. However, the H.264 is resource demanding, which is a disadvantage for embedded systems, ASICS and FPGA-based solutions.

In [2], x264 and x265, and their respective LCEVC implementations are investigated and evaluated using image quality metrics such as Peak Signal-to-Noise Ratio and Mean Opinion Score. A few game screenshots are used as test data. The paper indicates that, in terms of PSNR, while LCEVCx264 outperformed x264, x265 outperformed LCEVC-x265. Meanwhile, LCEVC-264 outperforms x264 in terms of overall MOS scores for the bitrates considered and x265 is more efficient than LCEVC-265 at lower bitrates.

PETRA '23, July 05–07, 2023, Corfu, Greece

ACM ISBN 979-8-4007-0069-9/23/07.
https://doi.org/10.1145/3594806.3594863

Dirac and Theora I, open and free video compression systems are highlighted and compared in [3]. The primary benchmark criterion is PSNR. The Akiyo video, showing a close-up of a single person, with varying frame rates, is used as test data. The evaluation shows that there is a substantial performance gap between Theora I and Dirac on the one hand, and H.264- and Motion JPEG2000-compliant reference systems on the other hand. However, a simplified version of Dirac, called Dirac Pro, achieves performance comparable to that of Motion JPEG2000, which can be better than one dB below the PSNR performance of H.264 with TV-size and HD video data.

The H.264, VP8 and MPEG4 video codecs are compared over software defined networks (SDN) in [4]. The performance of the codecs is evaluated in terms of transmission speed and image quality, using end-to-end delay and PSNR. Only interframe coding algorithms are presented in the paper. The paper concludes that MPEG4 provides the best image quality and the worst video latency over VP8 and H264.

The author of [5] proposes a method of comparing video codecs while also taking into account objective quality assessment metrics. The paper also shows the process of assessing the quality for pairs of the following video codecs: H.264, H.265, AV1. All of the codecs are inter-frame based. The metrics utilized in the paper are PSNR, SSIM and a metric suggested by the author of the paper. FFMPEG environment was used for compression and the parameters of the codecs were tuned based on ITU-T and YouTube recommendations.

In [6], several generations of video coding standards are compared by means of peak signal-to-noise ratio (PSNR). The paper includes H.262, H.263, but the main focus is on the H.264 and H.265 video codecs, all of which are inter-frame-based. For roughly comparable subjective quality, the bit-rate was lower for H.265 compared to H.264.

The video compression comparisons in [7] consider the encoding or decoding time needed for each algorithm, in real-time streaming conditions. All video test sequences are either HD or UHD resolution. The authors compare both inter-frame and intra-frame codecs. The criteria used in the paper are bitrate and encoding time, however, the quality of the test video is not measured.

The author of [8] gives an experimental primer on the current affairs in state-of-the-art video compression, focusing purely on rate-distortion performance under quantization and bitrate constraints, and disregarding domain-specific factors. The codecs are enabled as recommended by their developers for maximum performance.

Table 1 lists previously published evaluations of video compression methods, and compares them to the evaluation conducted in this paper. As we see on that table, our evaluation is the only one to use both natural and synthetic videos. Several prior evaluations do not report both PSNR and encoding time, whereas our evaluation reports both. Also, several prior evaluations do not include both inter-frame and intra-frame codecs, and our evaluation includes both types.

Table 1: Comparison of features of related work.

	Metric		Type of codec		Type of test video	
	PSNR	Encoding time	Inter-frame codecs	Intra-frame codecs	Natural video	Synthetic video
This paper	✓	✓	✓	✓	2	3
[1]	✗	✗	✓	✓	1	0
[2]	✓	✗	✓	✗	0	14
[3]	✓	✗	✓	✓	3	0
[4]	✓	✓	✓	✗	1	0
[5]	✓	✗	✓	✗	2	0
[6]	✓	✓	✓	✗	0	20
[7]	✗	✓	✓	✓	3	0
[8]	✓	✗	✓	✓	6	0

3 OVERVIEW OF THE CONSIDERED ALGORITHMS

3.1 MJPEG

MJPEG is one of the first ISO/ITU-T video compression standards [17]. MJPEG gives good compression results in both lossy and lossless compression with minimal encoding time. MJPEG uses a purely intra-frame compression scheme (compared to more complex inter-frame compression calculation schemes). An average compression ratio of 15:1 is achieved with lossy coding based on discrete cosine transform (DCT) blocks [17]. The absence of inter-frame compression in MJPEG generally prevents compression ratios greater than 1:20 from being obtained, depending on the tolerance of spatial distortion in the decoded frames of the video sequence. Since frames are compressed independently of each other, MJPEG requires less computing resources and RAM at the encoding stage. Lossless compression is implemented using interpolation compression techniques, which include differential coding, run length coding, and Huffman coding [18].

MJPEG also uses a quantization matrix with an adjustable ratio, which allows configuring the degree of image compression.

Zigzag scanning of the pixel matrix is performed on quantized coefficients, since it allows entropy to be encoded in order from low-frequency to high-frequency coefficients.

The frame compression process consists of the following elements:

- Color model transformations (most often to YCbCr) for color images.
- Discrete cosine transform (working with blocks of 8x8 pixels):

$$G_{u,v} = \frac{1}{4}(\alpha(u)\alpha(v)\sum_{x=0}^{7}\sum_{y=0}^{7} g_{x,y}\cos[\frac{(2x+1)u\pi}{16}]\cos[\frac{(2x+1)u\pi}{16}] \quad (1)$$

,where
u - horizontal spatial frequency, for integers $0 \leq u \leq 8$.
v - vertical spatial frequency, for integers $0 \leq v \leq 8$
$\alpha(u) = \{\frac{1}{\sqrt{2}, u=0}\}$ - normalizing scaling vector.

$g_{x,y}$ - pixel value at the point with coordinates (x,y).
$G_{u,v}$ - the DCT coefficient at the point with coordinates (u,v).

- Quantization.
- Zigzag traversal of a matrix of pixels.
- Entropy coding using Huffman tables [18].

Different quantization matrices are used for the luma and chrominance components. The quality factor is set using quantization tables.

3.2 MJPEG2000

The MJPEG2000 [11] format is a video compression standard that supports lossy and lossless compression of grayscale or color images. MJPEG2000 has high performance at low bit rate without sacrificing performance at high bit rate, region-of-interest coding, EBCOT (Embedded Block Coding with Optimal Truncation) [11], which is less restrictive than the EZW (Embedded Zerotrees of Wavelet transforms, coding using a nested null tree) [11], namely random access to individual areas of the video frame and error resistance. This algorithm is based on transformation and uses wavelet decomposition.

The wavelet transform has a 3dB advantage over DCT-based compression. Lossless compression is achieved as a result of transformations and entropy coding. In addition, at the beginning of the compression process, MJPEG2000 divides the frame into non-overlapping rectangular parts ("tiles"). Partitioning into "tiles" - one of the most important stages of the algorithm [11].

The algorithm supports the construction of wavelets using both convolution and Lifting Scheme. For both modes, the signal must first be expanded periodically. This is done so that the filtering operations that are performed on the signal boundaries operate on a single signal instance that matches the filter coefficient. Thus, the number of filter coefficients determines how far the signal propagates from both sides relative to the boundary. Convolution-based filtering is performed by performing a series of point operations between the high and low pass filter coefficients and the extended one-dimensional signal.

Filtering by the lifting scheme is carried out by replacing the values of odd pixels in a row or column with a weighted sum of the values of neighboring even pixels. For the lossless case, the results are rounded to integer values. The formulas for filtering based on the lifting scheme are given below:

$$y(2n+1) = x_{ext}(2n+1) - [\frac{x_{ext}(2n) + x_{ext}(2n+2)}{2}] \quad (2)$$

$$y(2n) = x_{ext}(2n) - [\frac{x_{ext}(2n-1) + x_{ext}(2n+1) + 2}{4}] \quad (3)$$

where x_{ext} is the extended input signal, and y is the output signal.

3.3 H.265

H.265 (HEVC) is a video compression algorithm developed by the ISO/IEC MPEG (Moving Picture Experts Group) and ITU-T VCEG (Video Coding Experts Group) working group [12]. It was created with the goal of significantly improving compression efficiency compared to existing standards (particularly H.264 [13]), reducing the required bit rate by 50 percent without losing video quality. HEVC has an identical scope to H.264/MPEG-4 AVC [13] and focuses on two key issues: increasing video resolution and expanding the use of parallel processing architectures.

First of all, the algorithm is aimed at consumer applications, since the formats are limited: 4:2:0 8 bits and 4:2:0 10 bits. The following modification of the algorithm supports 4:2:2 and 4:4:4 formats with color depth greater than 10 bits.

The H.265 video encoder follows the steps below:

- Splitting each frame into multiple blocks.
- Interpolate each block and subtract the predicted part from the block.
- Transformation and quantization of the residual (difference between the original video frame block and the predicted one).
- Output entropy encoding result, prediction information, mode information and headers.

The algorithm decoder does the following:

- Entropy decoding and extraction of elements of the encoded sequence.
- Scaling and inverting the transformation stage.
- Interpolate each block and add the predicted part to the result of the inverse transformation.
- Restoration of the decoded video image.

In the transform step, the residual data after the interpolation is transformed using a block transform based on the discrete cosine transform. Only 4x4 luma is applied to the residual data in each block.

4 SELECTION OF BENCHMARK CRITERIA

Solving the problems of performing remote surgical or rescue operations involves working with high-quality images, which, in the context of video compression algorithms, is inextricably linked with the peak signal-to-noise ratio (PSNR) criterion [14].

On top of that, many applications, for example broadcasting of sport events, require real-time responsiveness, so timing is also a high priority alongside quality. Accordingly, encoding time was chosen as the second comparison criterion, which reflects the time spent by the video codec on data processing.

Below is a brief description of the selected codec comparison criteria:

- Peak signal to noise ratio: The peak signal-to-noise ratio is an expression of the ratio between the maximum possible value (power) of the signal and the power of the distorting noise, which affects the quality of its presentation. Since many signals have a very wide dynamic range (the ratio between the largest and smallest possible values of a variable), PSNR is usually expressed in terms of a logarithmic dB scale and should not exceed 35 dB, otherwise distortion becomes visually noticeable [15]. PSNR in dB is defined as:

$$PSNR = 20\log(\frac{MAX_I}{\sqrt{MSE}}) \quad (4)$$

, where
MAX_I - maximum value accepted by the image pixel,
MSE - the root mean square error:

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}|I(i,j) - K(i,j)|^2 \qquad (5)$$

, where
I - original image, K - a noisy approximation of the original image, m - the number of pixel rows, n - the number of pixel columns. Formula used to calculate $Y`C_bC_r$ $PSNR$ for 4:2:0 format:

$$PSNR_{Y`C_bC_r} = \frac{6PSNR_{Y`} + PSNR_{C_b} + PSNR_{C_r}}{8} \qquad (6)$$

- Encoding time. The second criterion for comparative analysis was chosen the encoding time of algorithms. This indicator is critical for software and hardware systems which have a real-time workflow, for example, broadcasting of sport live events.

5 RESULTS OF ANALYSIS OF THE ALGORITHMS

In the process of conducting a comparative analysis of the considered algorithms, five test videos were used. The videos have fundamental structural and color differences, and these differences help highlight the dis/advantages of each codec when measuring PSNR. Furthermore, utilizing different frame rates allows testing the second criterion, encoding time, in the most efficient way. The videos that were used are presented below:

Figure 1: Left: Frame from Crowd video. Right: Frame from Person video.

The frames of the first test video: Crowd, 2560x1600, 30fps, and the second test video: Person, 1280x720, 60fps, are shown in Fig. 1. The Crowd video is natural and contains a large number of small-sized details, in fact, there are no smooth color transitions. The Person video is a structural antipode of Crowd and includes several monotonous zones with sharp color transitions. Also, there are objects with a large set of components on the Person video's frame.

Figure 2: Left: frame from Animation video. Right: frame from Videogame video

Fig. 2 represents a snapshot of the synthetic animated video sequence (1920x1080, 60fps). The video's structure includes mostly low-frequency regions and little motion. 2 also demonstrates a frame of a computer game's gameplay (1280x720, 60fps). The peculiarity of this video is that it contains a large amount of motion, which complicates the compression process, especially for inter-frame algorithms, including HEVC.

A snapshot of the recorded PC's desktop (2160x3840, 60fps) is shown on Fig. 3. The video sequence is synthetic and has an extremely complex structure because of a multi-color spreadsheet and a text document both of which add high frequency to every frame of the video sequence.

Below are the results of a comparative analysis of the H.265, MJPEG and MJPEG2000 algorithms using the test video sequences described above, according to the selected criteria.

Peak signal-to-noise ratio.

Table 2: Comparison of video codecs by PSNR.

PSNR (dB)			
	MJPEG	MJPEG2000	HEVC
Crowd	29.8	32.1	34.1
Person	32.2	34.9	36.1
Animation	40.4	39.6	42.6
Video game	27.6	27.2	29
Recorded desktop	25.7	33.6	37.4

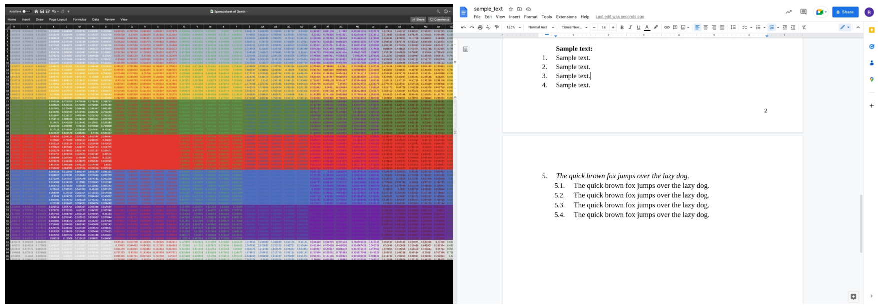

Figure 3: Frame from Recorded desktop video

To begin with, it is seen that the video with a large number of high-frequency regions (Crowd) is more difficult to compress for codecs based on both the wavelet transform and the discrete cosine transform, which is directly reflected in decibels: an average of 34.1 dB for H. 265, 32.1 dB for MJPEG2000 and 29.8 dB for MJPEG. However, the performance of H.265 is the most optimal in terms of peak signal-to-noise ratio in general and this comparative analysis in particular. The video with large low-frequency regions (Person) is processed more efficiently by all the codecs, however, the H.265 still has a significant advantage compared to the others. In addition, all three video codecs demonstrated outstanding results for the synthetic video with animation and the PSNR is around 40 dB. The video has distinct colors and mostly low-frequency regions which are easier to compress, however, it is worth to be noted that the HEVC showed the best result as well as for the previous videos. Furthermore, the video sequence of a video game contained a lot of motion and turned out to be a tall order for all three codecs. Meanwhile, the HEVC still has the biggest PSNR among others. In conclusion, the video sequence of a recorded PC's desktop was compressed efficiently by HEVC and MJPEG2000 but was problematic fir MJPEG which is a DCT-based codec and had to deal with a lot of high-frequency regions containing different colors.

Encoding time.

Table 3: Comparison of video codecs by encoding time of 60 frames in seconds.

Encoding time (sec)			
	MJPEG	MJPEG2000	HEVC
Crowd	2.5	4.4	3.8
Person	2.8	3.8	3.3
Animation	3.1	1.7	3.2
Video game	1.9	1.2	3.5
Recorded desktop	2.4	1.3	2.4

As per Table 3, both the Crowd and the Person video sequences led to similar results: the H.265 has a high latency, which can hardly be overcome, because of its interframe nature. Meanwhile, MJPEG and MJPEG2000 compress much faster. The large number of high frequency regions of the Crowd also makes the H.265 unstable in terms of latency and the time for processing every next frame fluctuates which might result in dramatic consequences for applications, related to IP-networks. MJPEG2000, which utilizes a Wavelet-based compression algorithm, showed the worst results for the Animation video sequence in terms of encoding time compared to MJPEG and HEVC, which signals about the weakness of this algorithm for the synthetic video with a lot of motion. On top of that, MJPEG2000 was the slowest with the Video game and the Recorded desktop sequences. Meanwhile, MJPEG demonstrated sufficiently good results.

6 CONCLUSION

This paper has considered three popular video compression algorithms, and offers a comparative analysis based on a certain set of criteria. The results of the analysis allow us to draw the following conclusions:

- For applications requiring high image quality, like surveillance systems, telesurgery and remote rescue tasks, the H.265 codec tends to work best, as shown by our PSNR results. At the same time, with respect to encoding time, the H.265 codec has a less appealing performance.
- The shortest encoding time is achieved using MJPEG, compared to H.265 and MJPEG2000. This can be a significant consideration for broadcasting of sport events and online video presentations.
- In situations, where high image quality is required, and low latency is not critical, the H.265 codec seems preferable. In situations demanding real-time performance and immediate responsiveness, our results give MJPEG the advantage.

In our future work, we plan to significantly expand this study, including more codecs such as MPEG, VP9, HT2K, and more metrics, such as VMAF, SSIM, and VSNR. We also plan to create a more comprehensive dataset, including several videos for each category, to get a better sense of statistical fluctuations in performance among videos of the same category.

ACKNOWLEDGMENTS

This work was partially supported by a grant from Macnica Americas.

REFERENCES

[1] S. Ponlatha, R. S. Sabeenian, Comparison of Video Compression Standards, International Journal of Computer and Electrical Engineering, Vol. 5, No. 6, December 2013

[2] Nabajeet Barman, Steven Schmidt, Saman Zadtootaghaj, and Maria G. Martini, Codec Compression Efficiency Evaluation of MPEG-5 part 2 (LCEVC) using Objective and Subjective Quality Assessment, 12, April, 2022
[3] Halbach T. (2009). COMPARISON OF OPEN AND FREE VIDEO COMPRESSION SYSTEMS - A Performance Evaluation. In Proceedings of the First International Conference on Computer Imaging Theory and Applications, pages 74-80
[4] Shiva Rowshanrad, Sahar Namvarasl, Behnaz Jamasb, Manijeh Keshtgary, Video Codec Standards Comparison for Video Streaming Over SDN, ARPN Journal of Systems and Software Vol. 5, No. 1, February 2015
[5] Klink, J. A Method of Codec Comparison and Selection for Good Quality Video Transmission Over Limited-Bandwidth Networks. Sensors 2021, 21, 4589
[6] Jens-Rainer Ohm, Member, IEEE, Gary J. Sullivan, Fellow, IEEE, Heiko Schwarz, Thiow Keng Tan, Senior Member, IEEE, and Thomas Wiegand, Fellow, IEEE Comparison of the Coding Efficiency of Video Coding Standards—Including High Efficiency Video Coding (HEVC). IEEE Transactions On Circuits And Systems For Video Technology, Vol. 22, No. 12, December 2012
[7] Georgios Minopoulos, Vasileios A. Memos, Konstantinos E. Psannis, Yutaka Ishibashi, Comparison of Video Codecs Performance for Real-Time Transmission, University of Macedonia Thessaloniki, 2022
[8] Etito Ohwovoriole, Yiannis Andreopoulos, Rate-Distortion Performance Of Contemporary Video Codecs: Comparison Of Google/WebM VP8, AVC/H.264 and HEVC TmuC, University College London, 2010
[9] Wallace J.K., "The JPEG still picture compression standard" Communication of the ACM, no.4, April 1991, p.31-44.
[10] Vatolin D. Methods of data compression: Arrangement of archivers, image and video compression / D. Vatolin [et al.] .- M. : Dialog-MEPhI, 2003 .- 381 p.
[11] Topiwala P., "Performance comparison of JPEG2000 and H.264/AVC high profile intra-frame coding on HD video sequences," Proc. SPIE International Symposium, Digital Image Processing, August 2006, p. 67-72.
[12] Sullivan J.J., "Overview of the high efficiency video coding (HEVC) standard", IEEE Transactions on Circuits and Systems for Video Technology, No.12, December 2012, p.1649-1668.
[13] JVT Draft ITU-T recommendation and final draft international standard of joint video specification (ITU-T Rec. H.264-ISO/IEC 14496-10 AVC), March 2003, JVT-G050
[14] Lukin V.V., Ponomarenko N.N., Krivenko S.S. Analysis of the efficiency of image compression methods in accordance with various quality criteria // Radioelectronics and Informatics. 2007. No. 2.
[15] Drozdov S.N., Zhiglaty A.A., Kravchenko P.P., Skorokhod S.V., Khusainov N.Sh. On the experience of practical implementation of the standards of the JPEG2000 family in the development of a software system for broadcasting a video stream in real time under conditions of a limited network resource. technical systems". - Taganrog: Publishing House of the Southern Federal University, 2014. - S. 459-472.
[16] HM 16.9 Reference Software: http://hevc.hhi.fraunhofer.de/
[17] JPEG reference software: ftp://ftp.simtel.net/pub/simtelnet/msdos/graphics/jpegsr6.zip
[18] JPEG2000 latest reference software: http://www.ece.uvic.ca/ mdadams/jasper/