MSU Codec Comparison 2018

Part IV: FullHD Content, High Quality Use Case

Video group head	Dr. Dmitriy Vatolin
Project head	Dr. Dmitriy Kulikov
Measurements & analysis	Dr. Mikhail Erofeev,
	Sergey Zvezdakov,
	Anastasia Antsiferova,
	Denis Kondranin

Free version

Codecs:

H.265

- sz265
- x265

Non H.265

- AV1
- SIF Encoder
- sz264
- VP9
- x264

CS MSU Graphics & Media Lab, Video Group

April 4, 2019

http://www.compression.ru/video/codec_comparison/index_en.html

videocodec-testing@graphics.cs.msu.ru

1. REPORT VERSIONS

	Free version	Enterprise version
Use cases	High quality use case (partically)	High quality use case
Metric: YUV-SSIM	✓	✓
Description of video sequences	✓	✓
Other objective metrics (Y-VMAF, Y-SSIM, U-SSIM, V-SSIM, YUV-PSNR, Y-PSNR, U-PSNR, V-PSNR)	✓	✓
Codec info (developer, version number, website link)	✓	✓
Relative quality analysis	✗	✓
Per-sequence-results	2 of 5 sequences	All 5 sequences for all use cases (in interactive charts)
Download links for video sequences	✗	✓
Encoders presets description	✗	✓
PDF report	42 pages	50 pages
HTML report	64 interactive charts	150+ interactive charts

Contents

1 Report Versions **2**

2 Acknowledgments **5**

3 Overview **6**
3.1 Sequences 6
3.2 Codecs 6

4 Objectives and Testing Rules **7**

5 RD Curves **8**

6 Encoding Speed **10**

7 Speed/Quality Trade-Off **12**

8 Bitrate Handling **15**

9 Relative Quality Analysis **16**

10 Conclusion **17**

A Sequences **19**
A.1 Bay Time-Lapse 19
A.2 Fire 20
A.3 Quiz Trailer 21
A.4 Talking Pair 22
A.5 Tractor 23

B Codecs **24**

C Sequence Selection **25**

D Figure Explanation **30**
D.1 RD Curves 30
D.2 Relative-Bitrate/Relative-Time Charts 30
D.3 Graph Example 30
D.4 Bitrate Ratio for the Same Quality 30
D.4.1 When RD Curves Fail to Cross the Quality Axis 32
D.4.2 When RD Curves Are Non-monotonic 33
D.5 Relative Quality Analysis 33

E Objective-Quality Metric Description **35**
E.1 SSIM (Structural Similarity) 35
E.1.1 Brief Description 35

E.1.2 Examples 36
E.1.3 Measurement method 38

F About the Graphics & Media Lab Video Group 40

2. ACKNOWLEDGMENTS

The Graphics & Media Lab Video Group would like to thank the following companies for providing the codecs and settings used in this report:

- AV1 developer team
- MulticoreWare, Inc.
- Nanjing Yunyan
- SIF Encoder Team
- VP9 developer team
- x264 Developer Team

We're also grateful to these companies for their help and technical support during the tests.

3. OVERVIEW

3.1. Sequences

	Sequence	Number of frames	Frame rate	Resolution
1.	Bay Time-Lapse	489	25	1920×1080
2.	Fire	601	25	1920×1080
3.	Quiz Trailer	600	30	1920×1080
4.	Talking Pair	782	24	1920×1080
5.	Tractor	690	25	1920×1080

Table 1: Summary of video sequences

Brief descriptions of the sequences used in our comparison appear in Table 1. Appendix A provides more-detailed descriptions of these sequences.

3.2. Codecs

Codec	Developer	Version
AV1	AV1 developer team	1.0.0-1276-g625cded05
SIF Encoder	SIF Encoder Team	1.66.0
sz264	Nanjing Yunyan	0.200.2668+100M 08e3761
sz265	Nanjing Yunyan	
VP9	VP9 developer team	v1.7.0-gcde3da57b
x264	x264 Developer Team	r2901-7d0ff22
x265	MulticoreWare, Inc.	2.7+19-1fafca24a399

Table 2: Short codecs' descriptions

Brief descriptions of the codecs used in our comparison appear in Table 2. We used x264 as a good-quality AVC reference codec. Appendix B provides detailed descriptions of all codecs in our comparison.

4. OBJECTIVES AND TESTING RULES

In this report we use objective assessment methods to compare the encoding quality of recent HEVC encoders as well as encoders implementing other standards. This effort employed 5 video sequences at 1080p resolution to evaluate codec performance. To choose out test set, we analyzed 539,765 video sequences and selected representative examples (a detailed description of the selection process appears in Appendix C).

This report contains one use cases: high-quality encoding. For this use case we offered the codec developers the option to provide encoding parameters for our tests. If they declined to provide any, we either used the same parameters from our prior study or, if none were available, did our best to choose good parameters ourselves. Nevertheless, the parameters had to satisfy a minimum speed requirements: no more than 120 seconds per frame.

Our comparison used a computer with the following configuration: based on an Intel Core i7-8700K (Coffee Lake) processor @ 3.7GHz with 16 GB of RAM running Windows 10. For objective quality measurements we used the YUV-SSIM metric (see Appendix E.1).

5. RD CURVES

Next figures show RD curves for video sequences. Judging from the mean quality scores (computed using the method described in Section D), first place in the quality competition goes to **AV1**, second place goes to **VP9**, **x265**, and **sz265**, and third place to **sz264** and **x264** . Please note that **AV1** encoder does not show the best quality among all encoders at *Bay Time-Lapse* sequence.

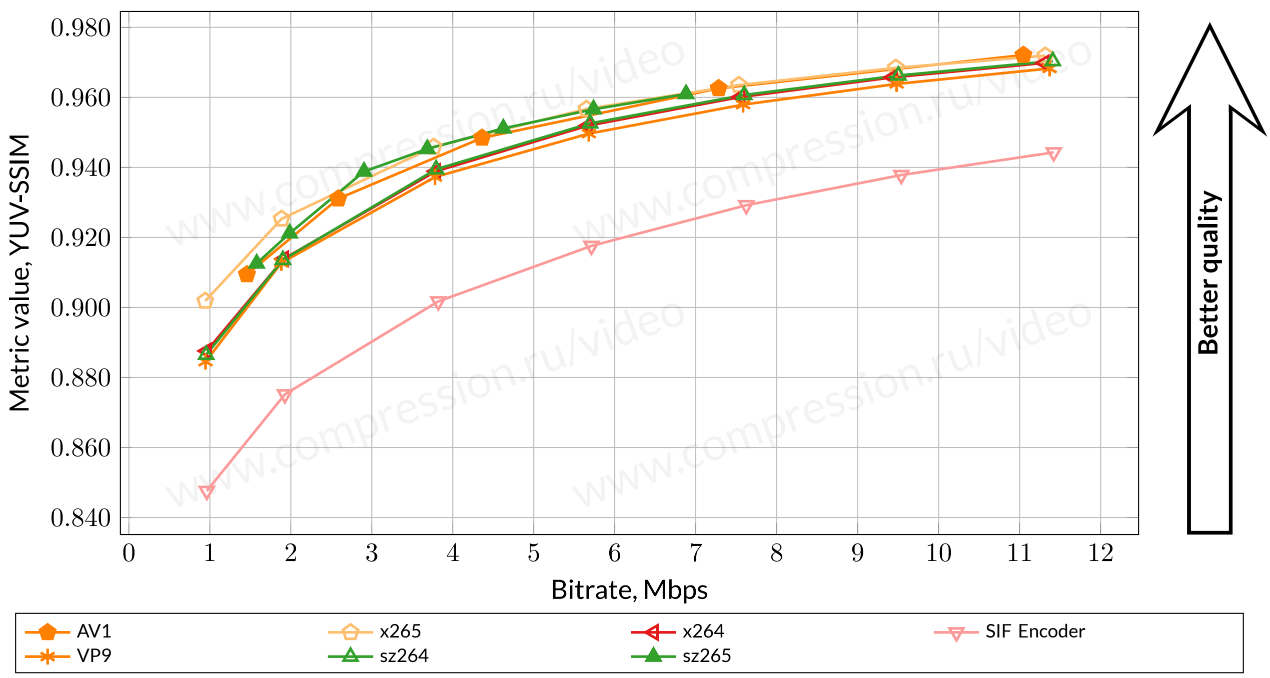


Figure 1: Bitrate/quality—use case "High quality use case," *Bay Time-Lapse* sequence, YUV-SSIM metric.

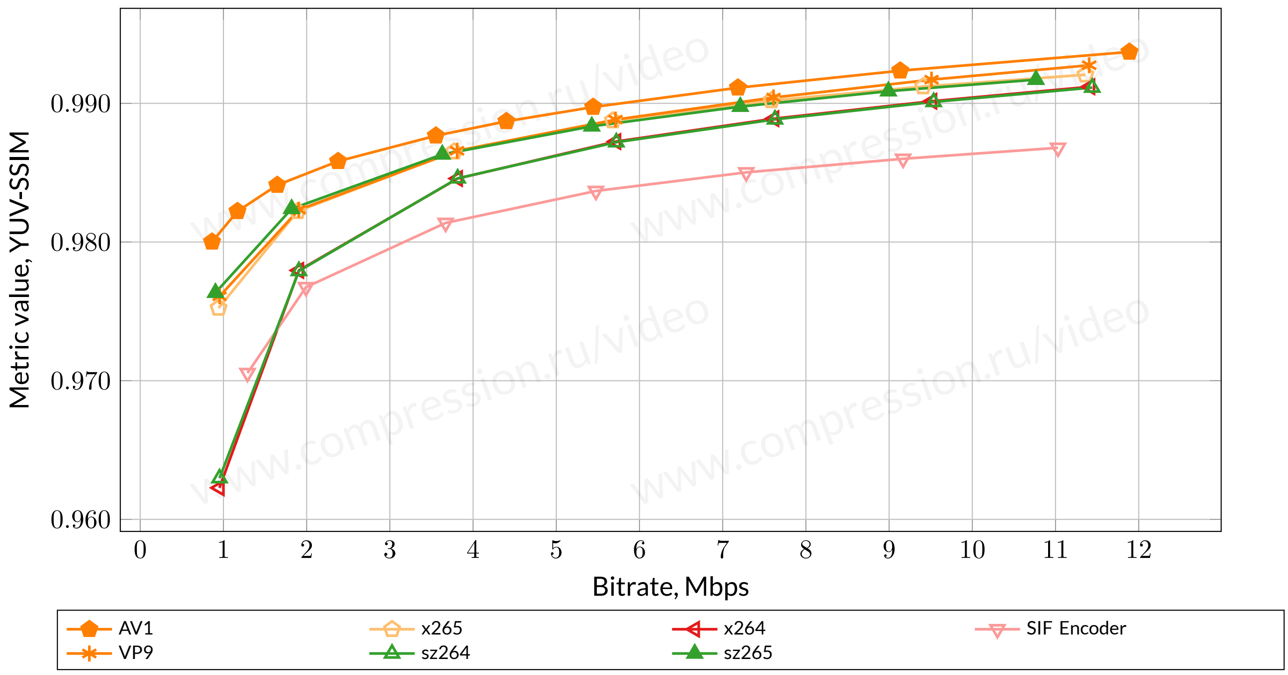
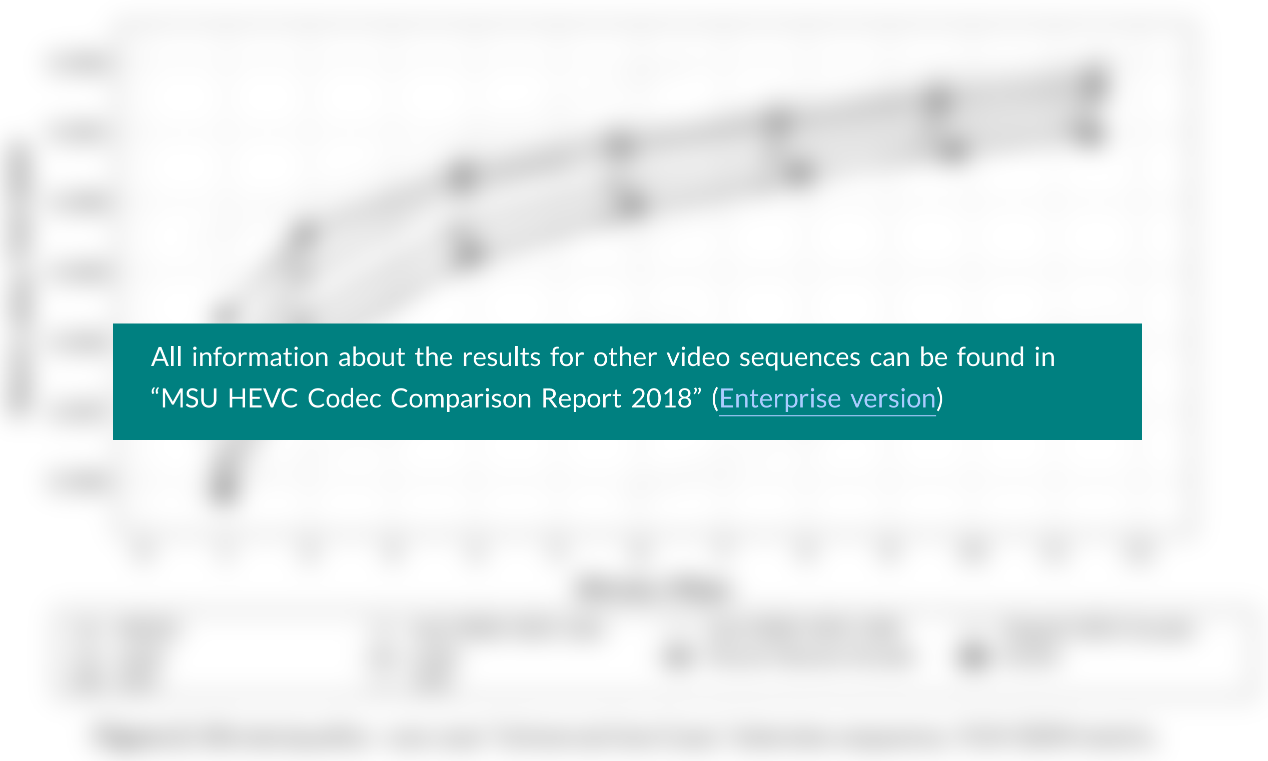


Figure 2: Bitrate/quality—use case "High quality use case," *Fire* sequence, YUV-SSIM metric.



All information about the results for other video sequences can be found in "MSU HEVC Codec Comparison Report 2018" (Enterprise version)

6. ENCODING SPEED

Judging from the mean speed scores (computed using the method described in Section D), first place in the speed competition goes to **SIF Encoder**, second place goes to **x264**, and third place to **vp9** and **x265**. **AV1** encoder shows 10-50 times lower encoding speed comparing to most participants in this use-case. The slowest encoder in this use-case (**sz265**) is 2,45 times faster than **AV1**.

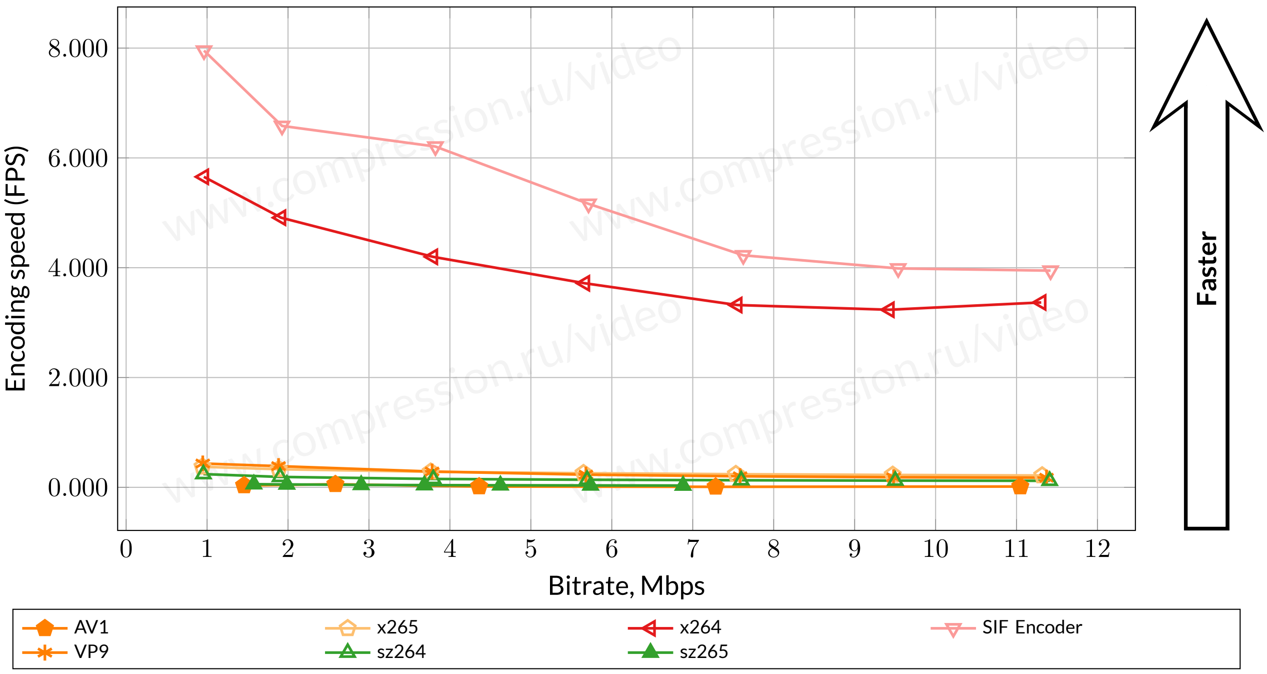


Figure 3: Encoding speed—use case "High quality use case," *Bay Time-Lapse* sequence.

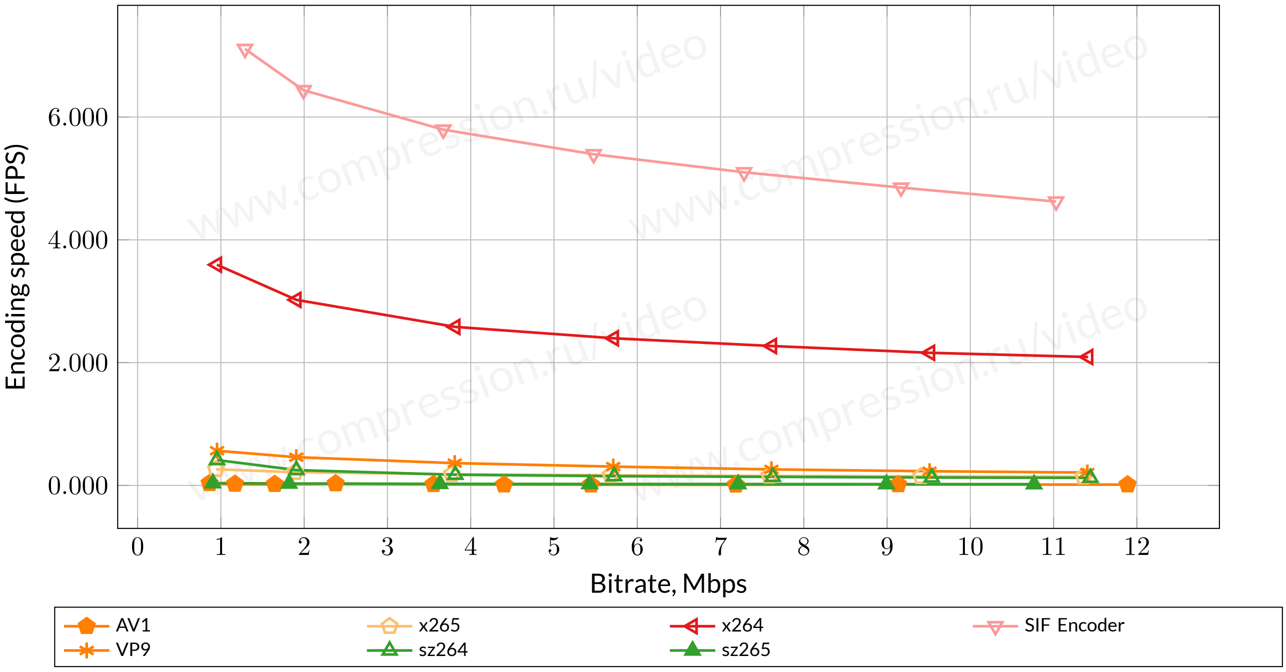
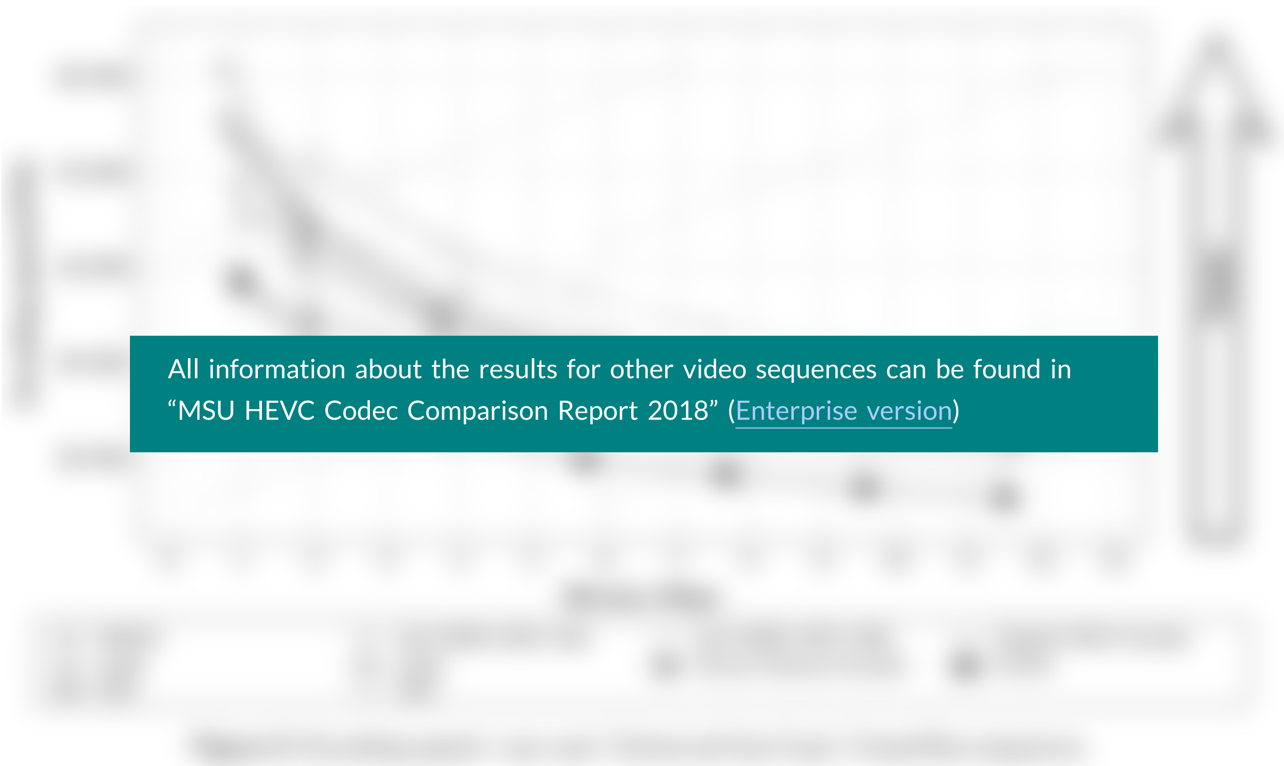


Figure 4: Encoding speed—use case "High quality use case," *Fire* sequence.



All information about the results for other video sequences can be found in "MSU HEVC Codec Comparison Report 2018" (Enterprise version)

7. SPEED/QUALITY TRADE-OFF

On average Pareto optimal presets ("Pareto optimal" encoder/preset means there is no encoder faster and better than it in this test.) are: **AV1**, **VP9**, **x264**, and **SIF Encoder**. But there are some differences depending on test sequences: at *Bay Time-Lapse* sequence there are only three Pareto optimal encoders: **SIF**, **x264**, and **x265**; at *Quiz Trailer* and *Talking Pair* sequences **x265** is also included in Pareto optimal list, etc.

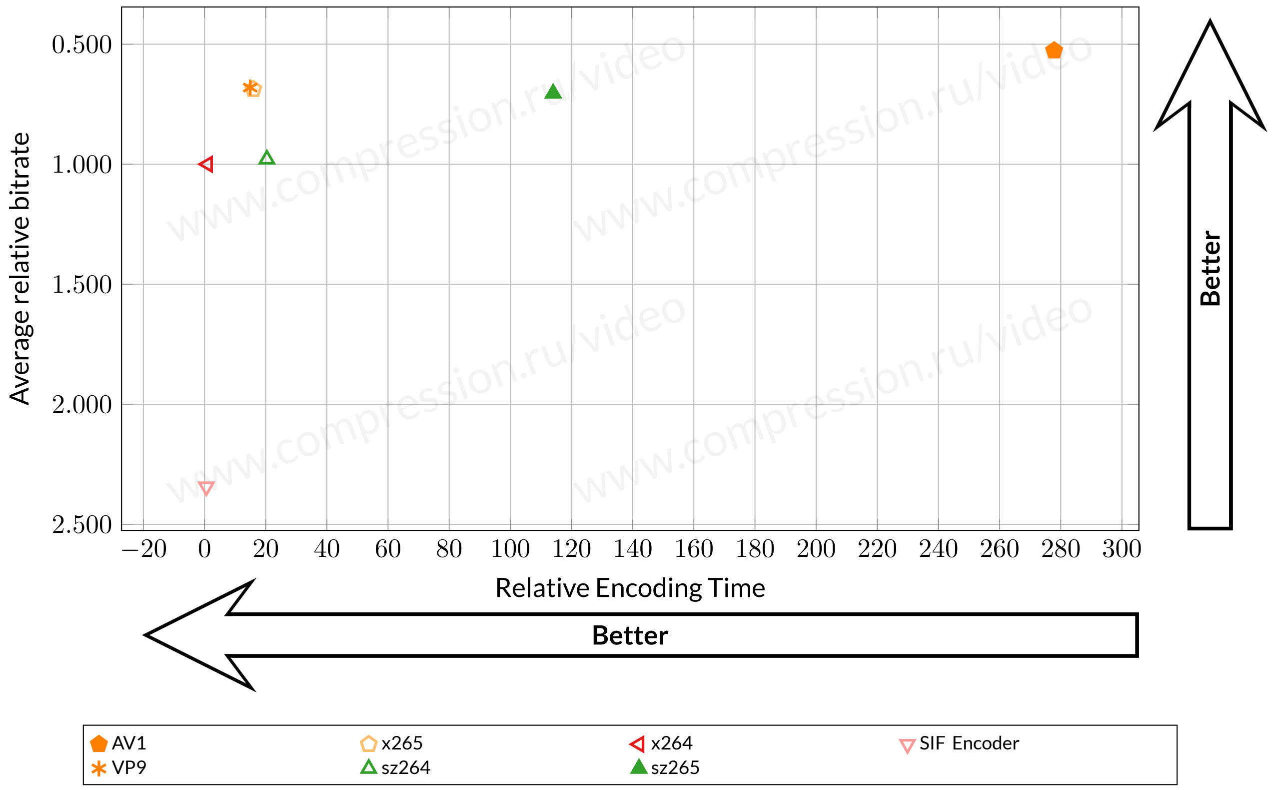


Figure 5: Speed/Quality Trade-Off—use case "High quality use case," all sequences, YUV-SSIM metric.

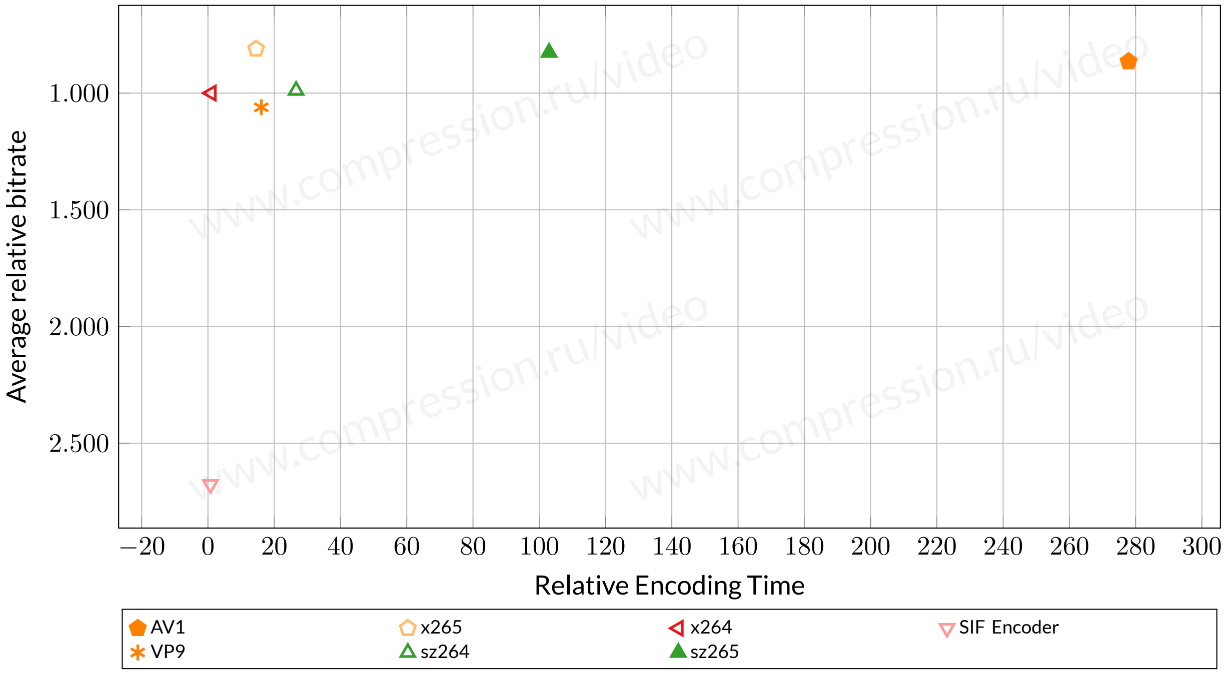


Figure 6: Speed/Quality Trade-Off—use case “High quality use case,” *Bay Time-Lapse* sequence, YUV-SSIM metric.

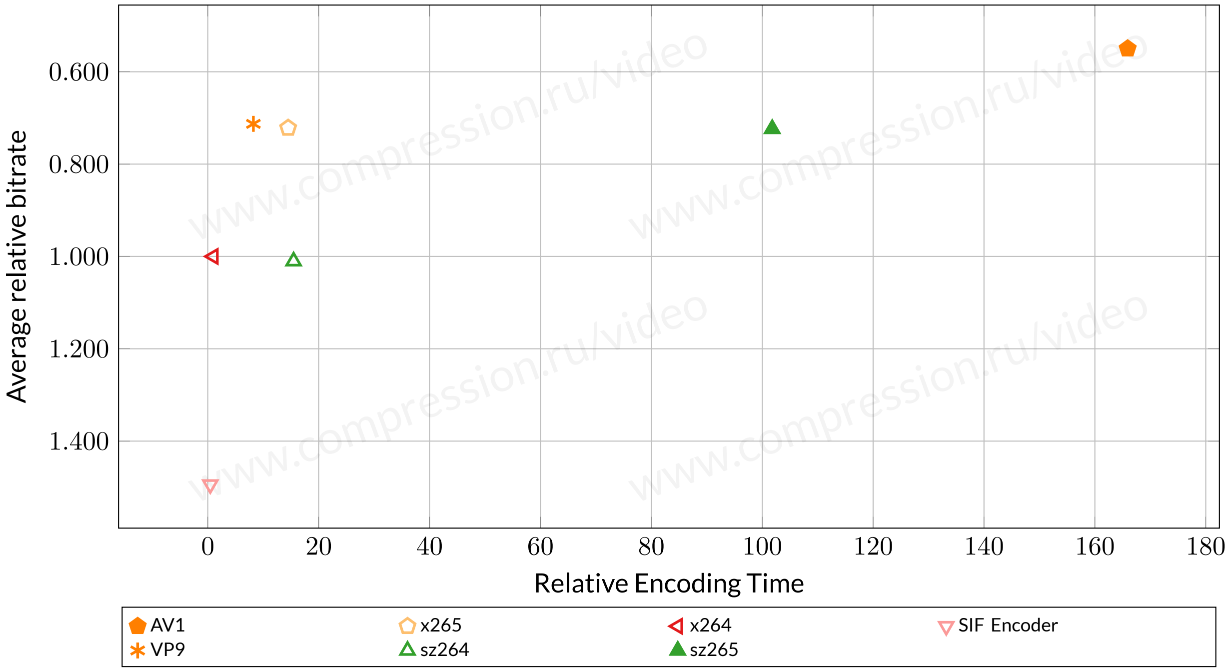
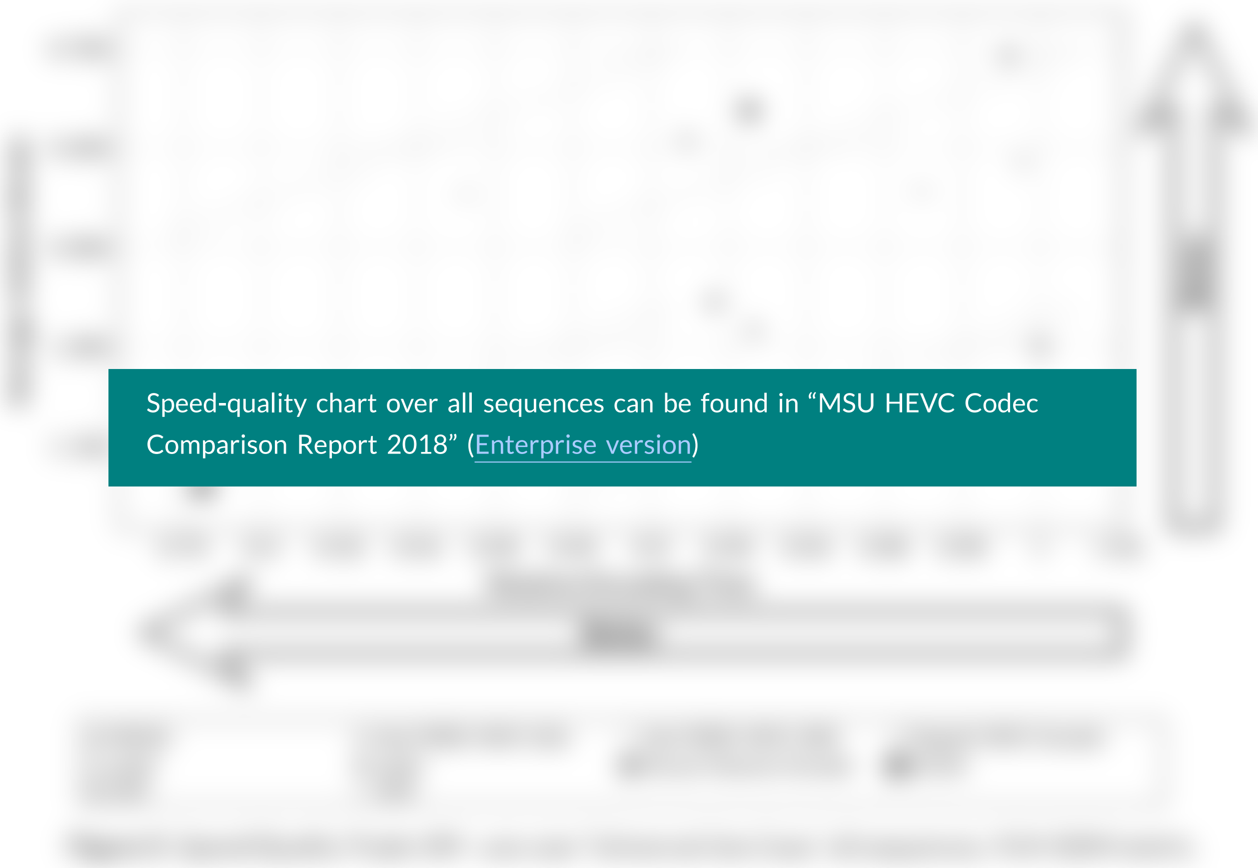


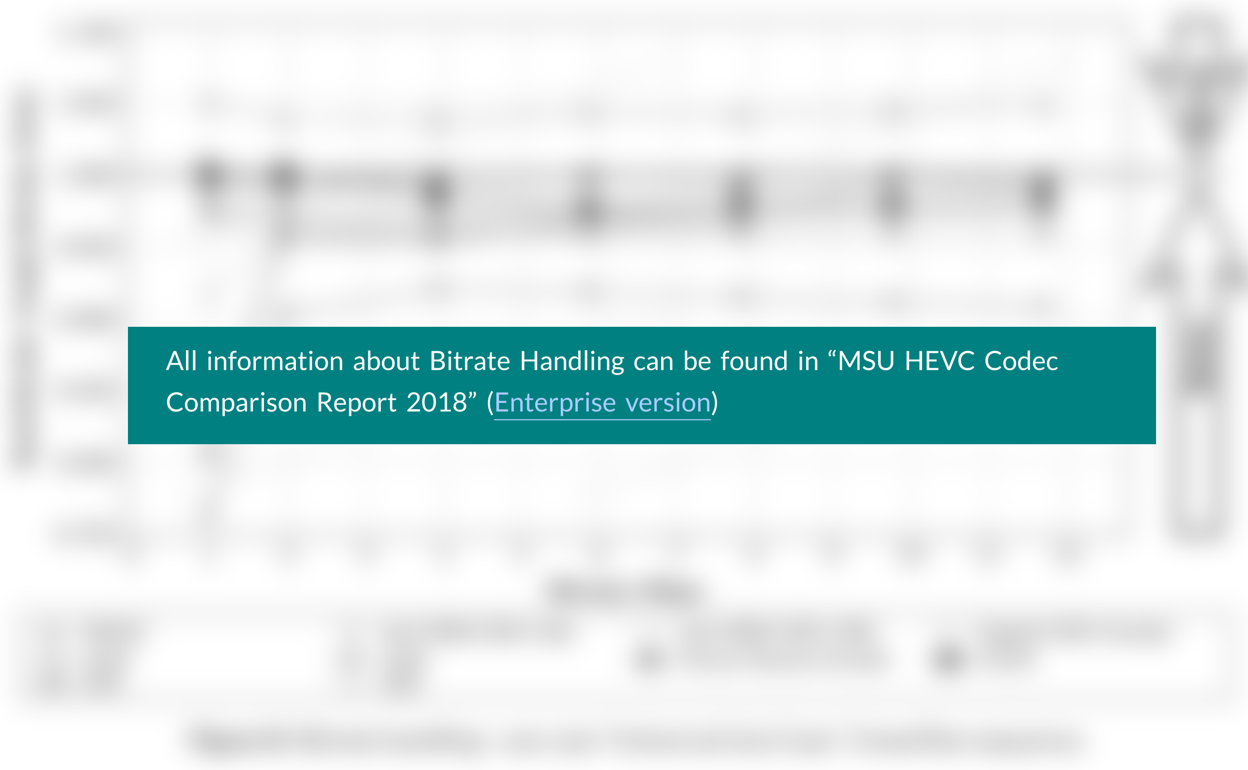
Figure 7: Speed/Quality Trade-Off—use case “High quality use case,” *Fire* sequence, YUV-SSIM metric.



Speed-quality chart over all sequences can be found in “MSU HEVC Codec Comparison Report 2018” (Enterprise version)

8. BITRATE HANDLING

The plots below show how accurately encoded stream's real bitrate matches bitrate requested by a user. **AV1** is absent in this chapter because it was used in CQ (constant quality) mode. Other encoders have some issues in some sequences: **sz265** – at *Bay Time-Lapse* sequence, **SIF** – at *Fire* sequence, *Quiz Trailer* sequence was complex task for bitrate handling mechanism for most of encoders, **VP9** has strong issues at *Talking Pair* sequence.



All information about Bitrate Handling can be found in "MSU HEVC Codec Comparison Report 2018" (Enterprise version)

9. RELATIVE QUALITY ANALYSIS

Note that each number in the tables below corresponds to some range of bitrates (see Appendix D.5). Unfortunately, these ranges can differ significantly because of differences in the quality of compared encoders. This situation can lead to some inadequate results when three or more codecs are compared.

All the information about Relative Quality Analysis could be found in "MSU HEVC Codec Comparison Report 2018" (Enterprise version)

10. CONCLUSION

This ranks are based on quality results ONLY (not considering encoding speed). If we will perform complex analysis – the results will be the next:

- **AV1** has extremely high encoding quality in approachable to other encoders and very low encoding speed due to lack of speed optimization.
- **VP9** and **x265** encoder have very close results (in speed and quality) and have a very good quality/speed balance for high-quality encoding
- **sz265** has good quality results with comparative low encoding speed

The plot below shows overall quality scores for the encoders in our comparison (see Section D for a description of the integral-score computation method). First place in the quality competition goes to **AV1**, second place goes to **VP9**, **x265**, and **sz265**, and third place to **sz264** and **x264**.

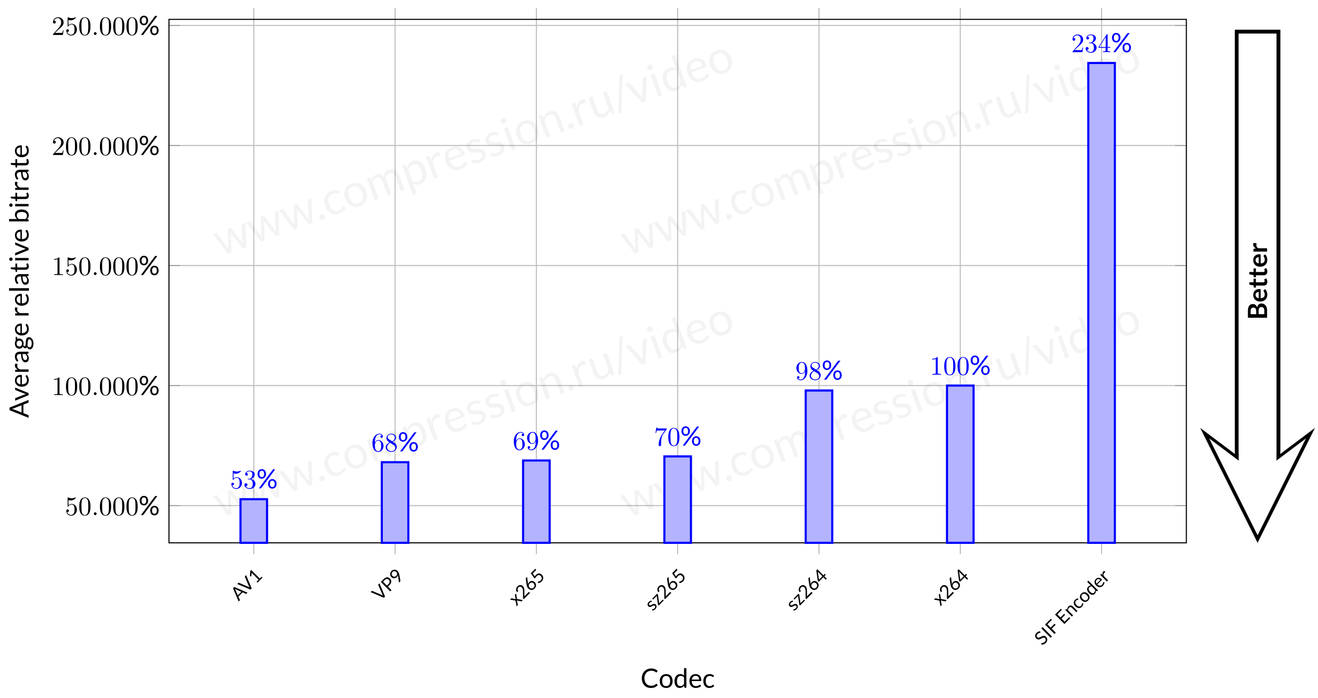


Figure 8: Average bitrate ratio for a fixed quality—all sequences, YUV-SSIM metric.

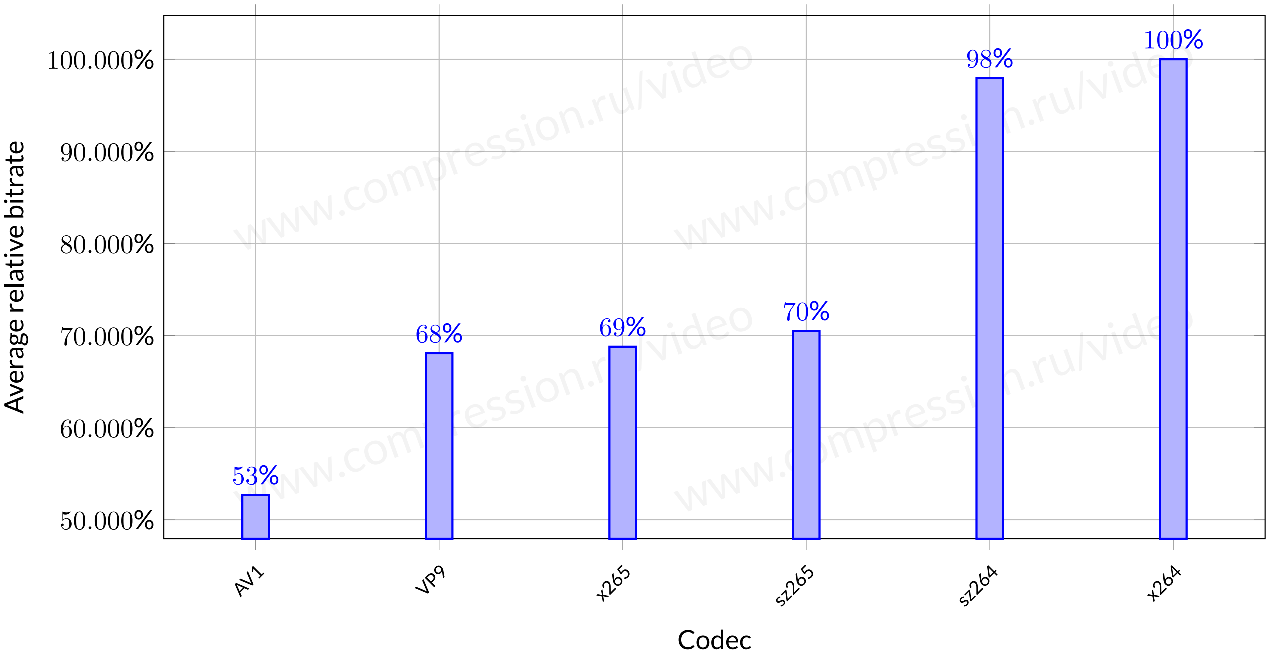


Figure 9: Average bitrate ratio for a fixed quality—all sequences, YUV-SSIM metric, without SIF Encoder.

A. SEQUENCES

Direct download links to video sequences used in this comparison can be found in "MSU HEVC Codec Comparison Report 2018" (Enterprise version)

A.1. Bay Time-Lapse

Sequence title	Bay Time-Lapse
Resolution	1920×1080
Number of frames	489
Color space	YV12
Frames per second	25
Source resolution	FullHD
Bitrate	245.545

Time-lapse view of a sunny bay with grass, moving ships and clouds.

Figure 10: Bay Time-Lapse sequence, frame 416

A.2. Fire

Sequence title	Fire
Resolution	1920×1080
Number of frames	601
Color space	YV12
Frames per second	25
Source resolution	FullHD
Bitrate	622.080

Recording of a bonfire. Initially static camera starts to shake.

Figure 11: Fire sequence, frame 25

A.3. Quiz Trailer

Sequence title	Quiz Trailer
Resolution	1920×1080
Number of frames	600
Color space	YV12
Frames per second	30
Source resolution	FullHD
Bitrate	143.259

A quiz-show trailer with numerous CG effects. A camera is zooming in on people.

Figure 12: Quiz Trailer sequence, frame 350

A.4. Talking Pair

Sequence title	Talking Pair
Resolution	1920×1080
Number of frames	782
Color space	YV12
Frames per second	24
Source resolution	FullHD
Bitrate	113.309

A man and woman talk in a room. Their faces are captured close up.

Figure 13: Talking Pair sequence, frame 260

A.5. Tractor

Sequence title	Tractor
Resolution	1920×1080
Number of frames	690
Color space	YV12
Frames per second	25
Source resolution	FullHD
Bitrate	622.080

A tractor clears the field; scene captured with zoom in.

Figure 14: Tractor sequence, frame 490

B. CODECS

All use encoders presets description could be found in "MSU HEVC Codec Comparison Report 2018" (Enterprise version)

C. SEQUENCE SELECTION

In "MSU Video Codecs Comparison 2016" we introduced a new technique for selecting test sequences. This technique create a data set containing representative sequences that encoders face in everyday situations. For this report we use the same method, but we updated the video database from which we sample videos.

We analyzed 539,765 videos at Vimeo, looking for 4K and FullHD examples with high bitrates (we chose 50 Mbps as our minimum). Doing so enabled us to find and download 942 new 4K videos and 2346 new FullHD videos. Figure 15 shows the bitrate distributions for last year's data set and for the updated data set.

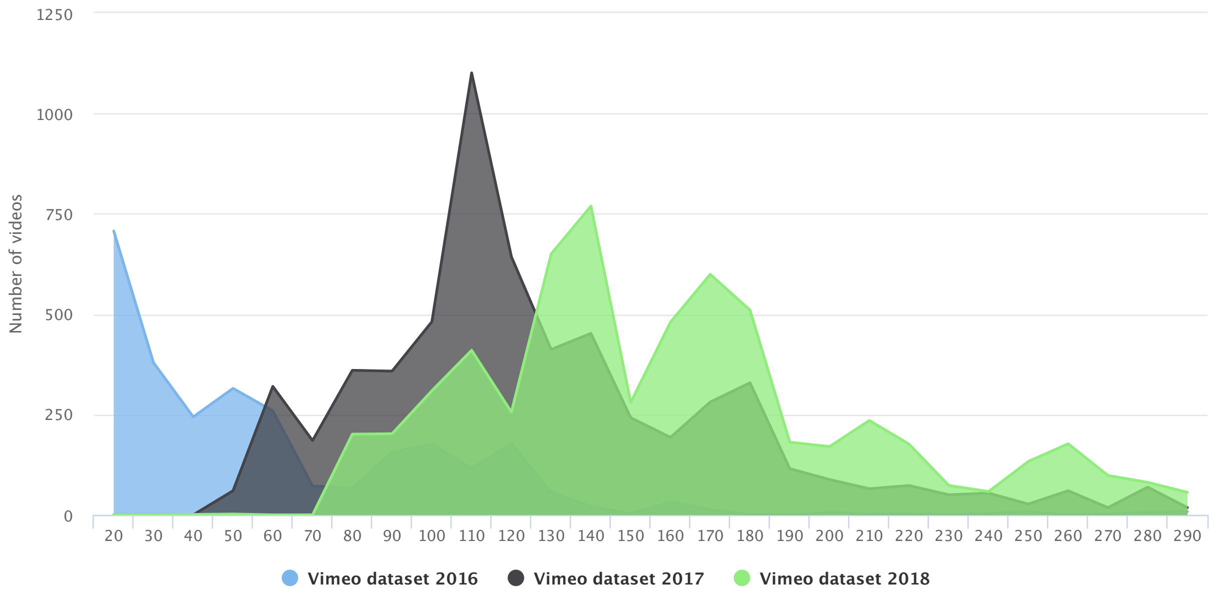


Figure 15: Bitrate distributions for video data set.

We resized and cropped the 4K videos to FullHD resolution in order to avoid compression artifacts, and at scene changes we cut all videos to samples, using an approximate length of 1,000 frames. Besides 6,534 samples from 2,748 newly downloaded videos, we also used 6,390 samples from "MSU Video Codecs Comparison 2017" and 2,909 samples from "MSU Video Codecs Comparison 2016". Thus, our sample database for this year consisted of 15,833 items.

To evaluate spatial and temporal complexity, we encoded all samples using x264 with a constant quantization parameter (QP). We calculated the temporal and spatial complexity for each scene, defining spatial complexity as the average size of the I-frame normalized to the sample's uncompressed frame size. Temporal complexity in our definition is the average size of the P-frame divided by average size of I-frame. [1]

In this year we slightly changed the temporal and spatial complexity calculation process by adding an additional preprocessing step. We use sourse videos from Vimeo, that was uploaded by users, so they all have different chroma subsampling which affects the results of videos evaluated complexity. Therefore to unificate the spatial and temporal complexity results of analysed videos, they all were converted to YUV 4:2:0 chroma subsample. Distribution of obtained samples compared to samples from previous codec comparisons is shown in Figure 16.

[1] C. Chen et. al., "A Subjective Study for the Design of Multi-resolution ABR Video Streams with the VP9 Codec," 2016.

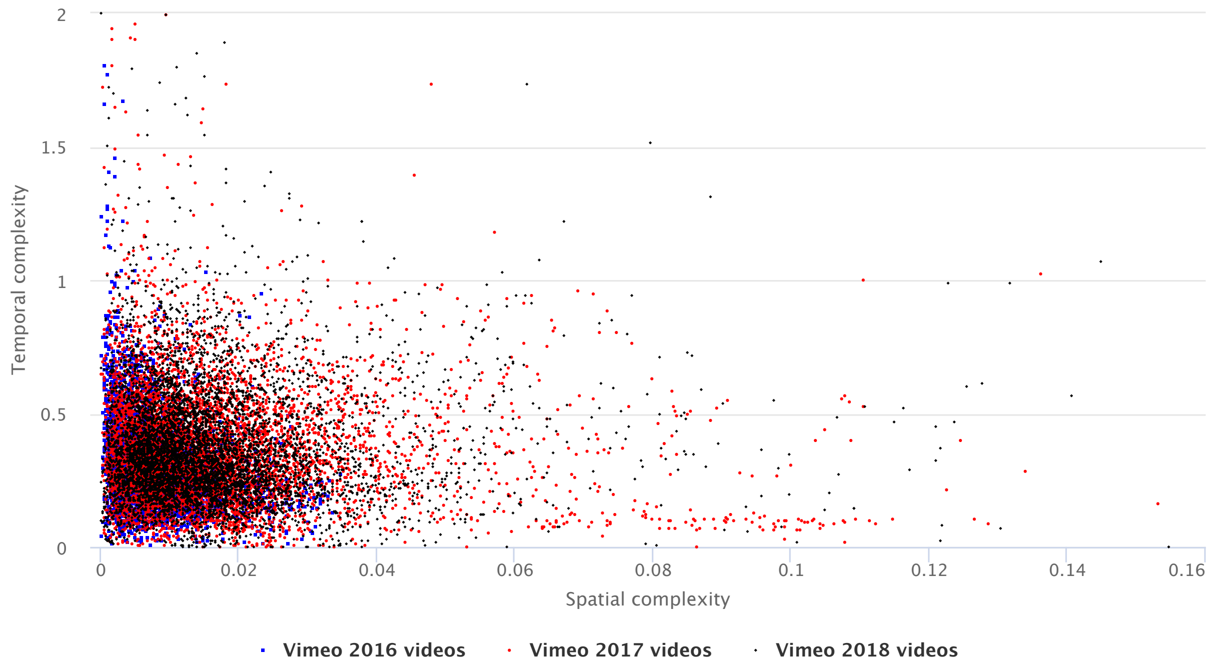


Figure 16: Distribution of obtained samples.

Figure 16 reveals that the new samples have a distribution similar to that of samples from "MSU Video Codecs Comparison 2017". We used the following process to prepare the data set.

We divided the video database into 28 clusters. To avoid completely changing the sequence list, we gave sequences from last year's FullHD data set 10 times greater weight than other sequences. For each cluster we selected the video sequence that's closest to the cluster's center and that has a license enabling derivatives and commercial use. Figure 17 shows the cluster boundaries and constituent sequences.

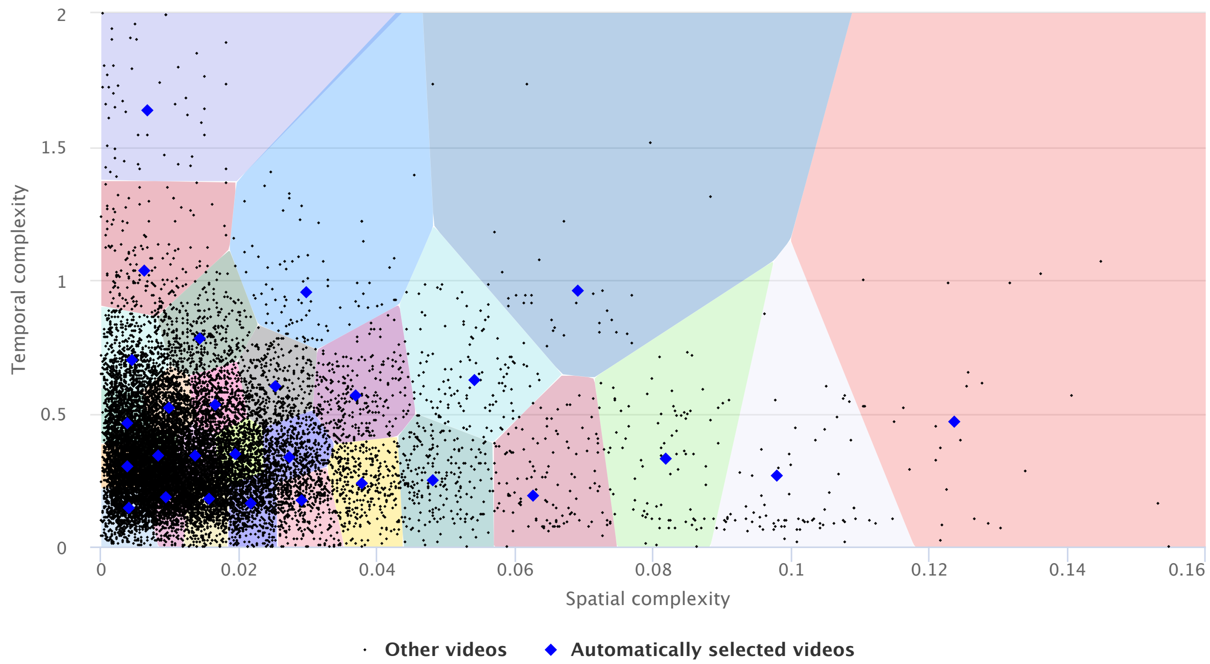


Figure 17: Segmentation of samples.

Figure 18 shows the correspondence of sequences from the previous data set to the newly selected ones. As the figure demonstrates, after adding a preprocessing step for video sequences some clusters don't include videos from old data set.

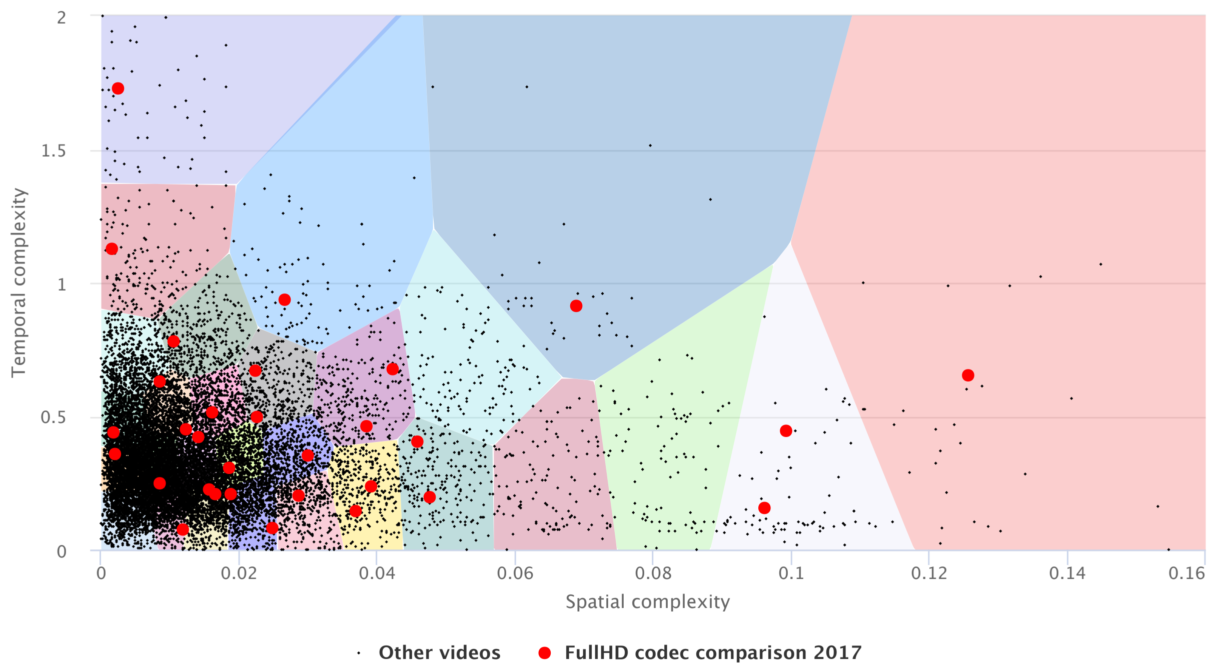


Figure 18: Segmentation of samples relative to old data set.

Some automatically chosen samples contain company names or have other copyright issues, so we removed them from their respective clusters and replaced them with other samples having a suitable license. Figure 19 illustrates these adjustments.

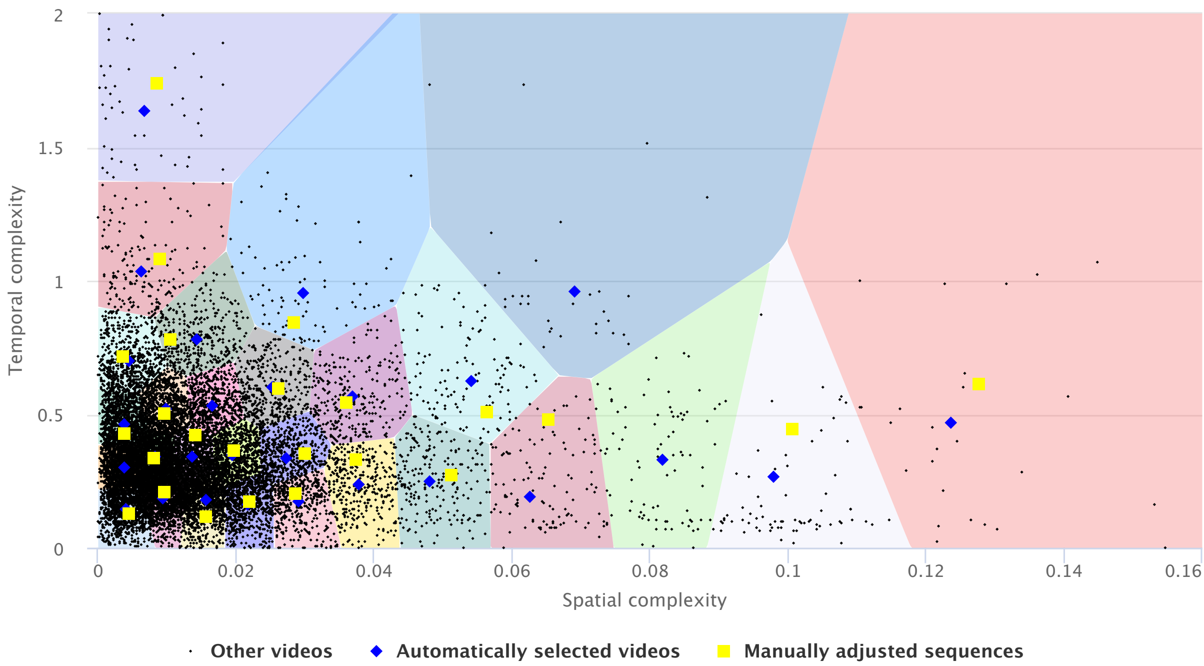


Figure 19: Adjustments to test data set.

Figure 20 shows the final distribution of sequences in the data set.

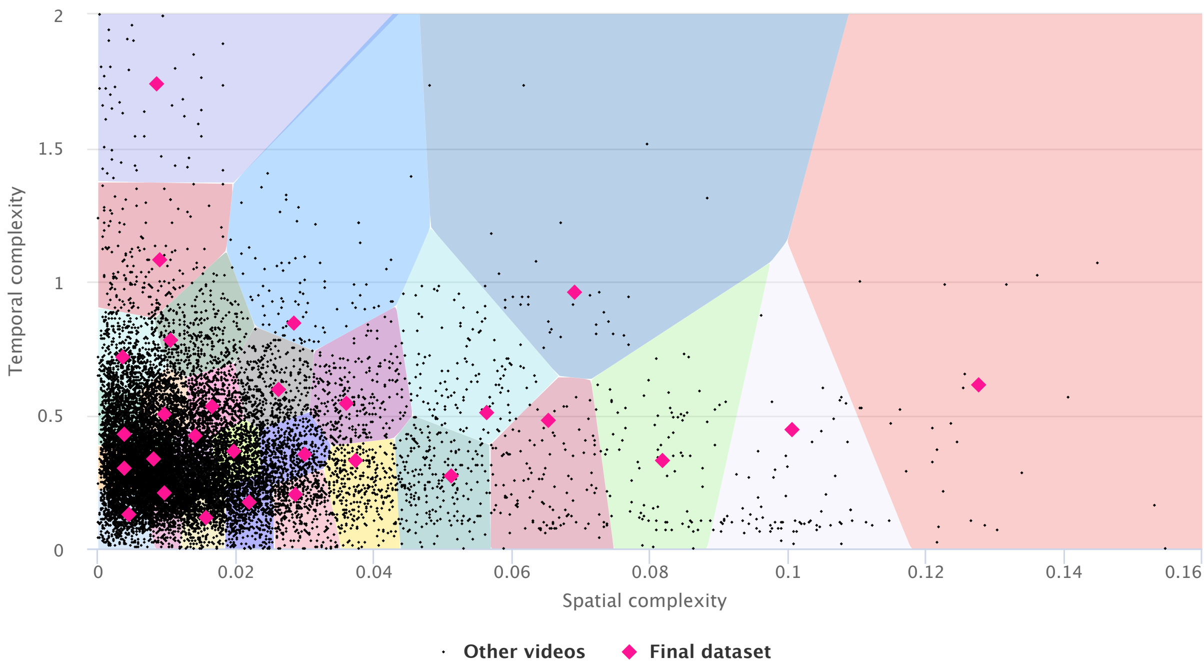


Figure 20: Distribution of sequences in final data set.

The new data set consists of 28 sequences: 5 from the old data set, 16 new ones from Vimeo and 7 from xiph.org. 25 sequences from the old data set were excluded. The average bitrate for all sequences in the final set is 449.72 Mbps, median – 192.02 Mbps. "Hera" (90 Mbps), "Television studio" (92 Mbps) and "Foggy beach" (93 Mbps) sequences have minimal bitrates. The complete list of sequences for new data set appears in Appendix A.

We also compared the distribution of videos from xiph.org with clusters obtained from our data set from Vimeo. The result is presented on Figure 21. It shows that most of the videos from xiph.org database have high spatial and temporal complexity with which codecs rarely face in everyday life.

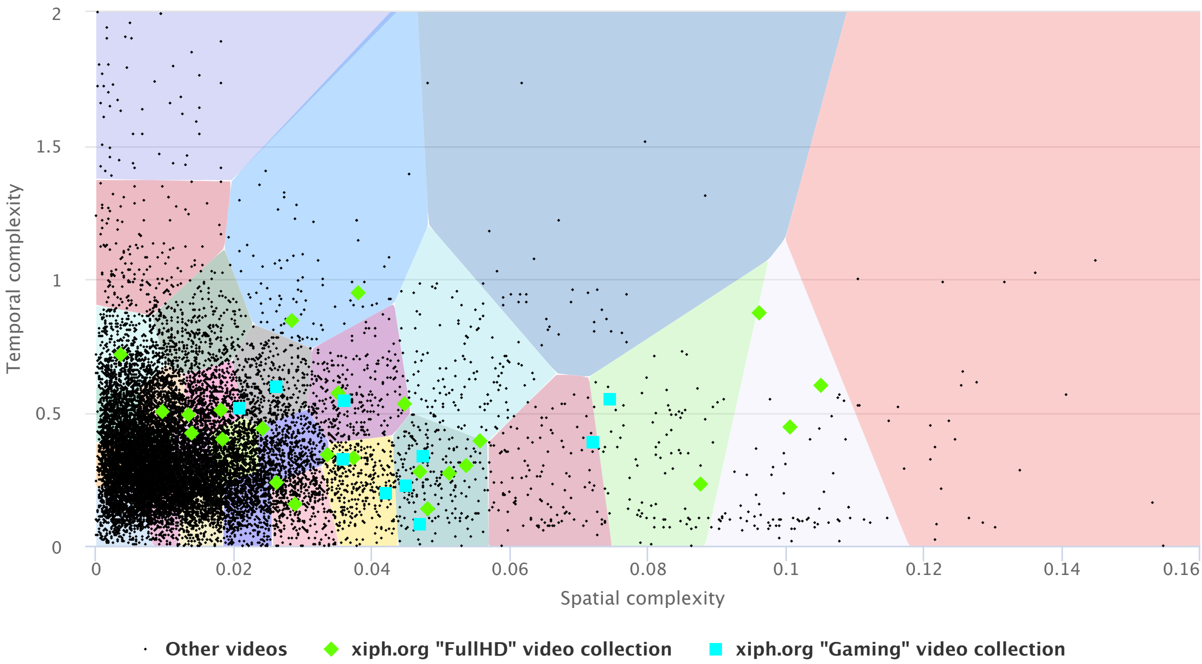


Figure 21: Comparison with xiph.org

D. FIGURE EXPLANATION

The main charts in this comparison are classic RD curves (quality/bitrate graphs) and relative-bitrate/relative-time charts. Additionally, we also used bitrate-handling charts (the ratio of real to target bitrates) and per-frame quality charts.

D.1. RD Curves

The RD charts show variation in codec quality by bitrate or file size. For this metric, a higher value presumably indicates better quality.

D.2. Relative-Bitrate/Relative-Time Charts

Relative-bitrate/relative-time charts show the average bitrate's dependence on relative encoding time for a fixed-quality output. The y-axis shows the ratio of a codec's bitrate under test to the reference codec's bitrate for a fixed quality. A lower value (that is, a higher the value on the graph) indicates a better-performing codec. For example, a value of 0.7 means the codec can encode the sequence in a file that's 30% smaller what the reference codec produces.

The x-axis shows the relative encoding time. Larger values indicate a slower codec. For example, a value of 2.5 means the codec works 2.5 times slower, on average, than the reference codec.

D.3. Graph Example

Figure 22 shows a situation where these graphs can be useful. In the top-left graph, the "Green" codec clearly produces better quality than the "Black" codec. On the other hand, the top-right graph shows that the "Green" codec is slightly slower. Relative-bitrate/relative-time graphs can be useful in precisely these situations: the bottom graph clearly shows that one codec is slower but yields higher visual quality, whereas the other codec is faster but yields lower visual quality.

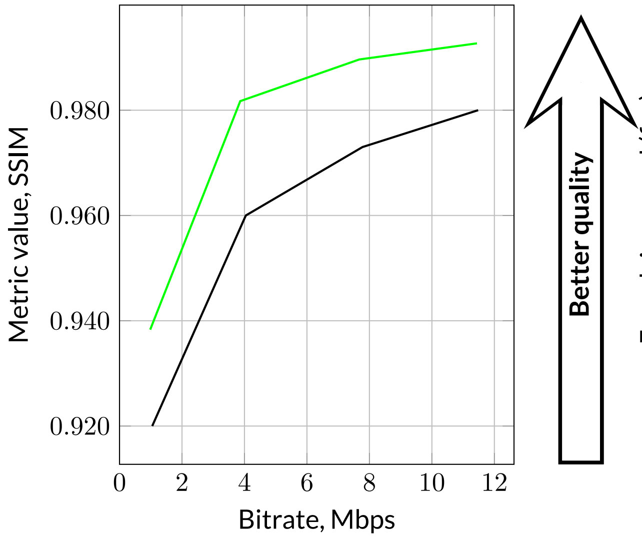
Owing to these advantages, we frequently use relative-bitrate/relative-time graphs in this report because they assist in evaluating the codecs in the test set, especially when the number of codecs is large.

A more detailed description of how we prepared these graphs appears below.

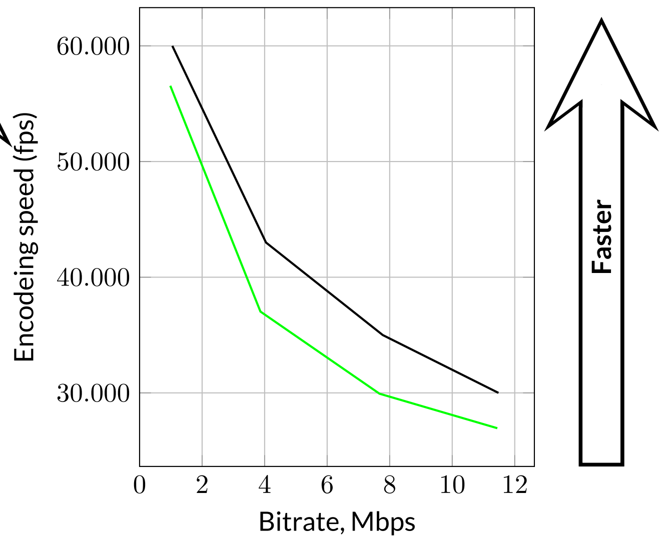
D.4. Bitrate Ratio for the Same Quality

The first step in computing the average bitrate ratio for a fixed quality is to invert the axes of the bitrate/quality graph (see Figure 23b). All further computations use the inverted graph.

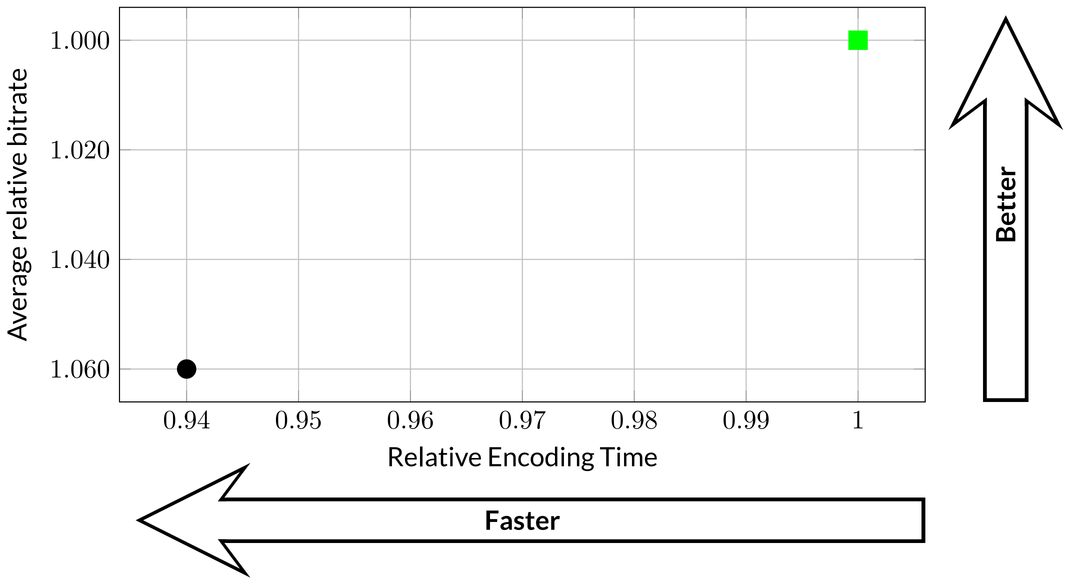
The second step involves averaging the interval over which the quality axis is chosen. The averaging is only over those segments for which both codecs yield results. This limitation is due to the difficulty of developing extrap-



(a) RD curve. "Green" codec is better!

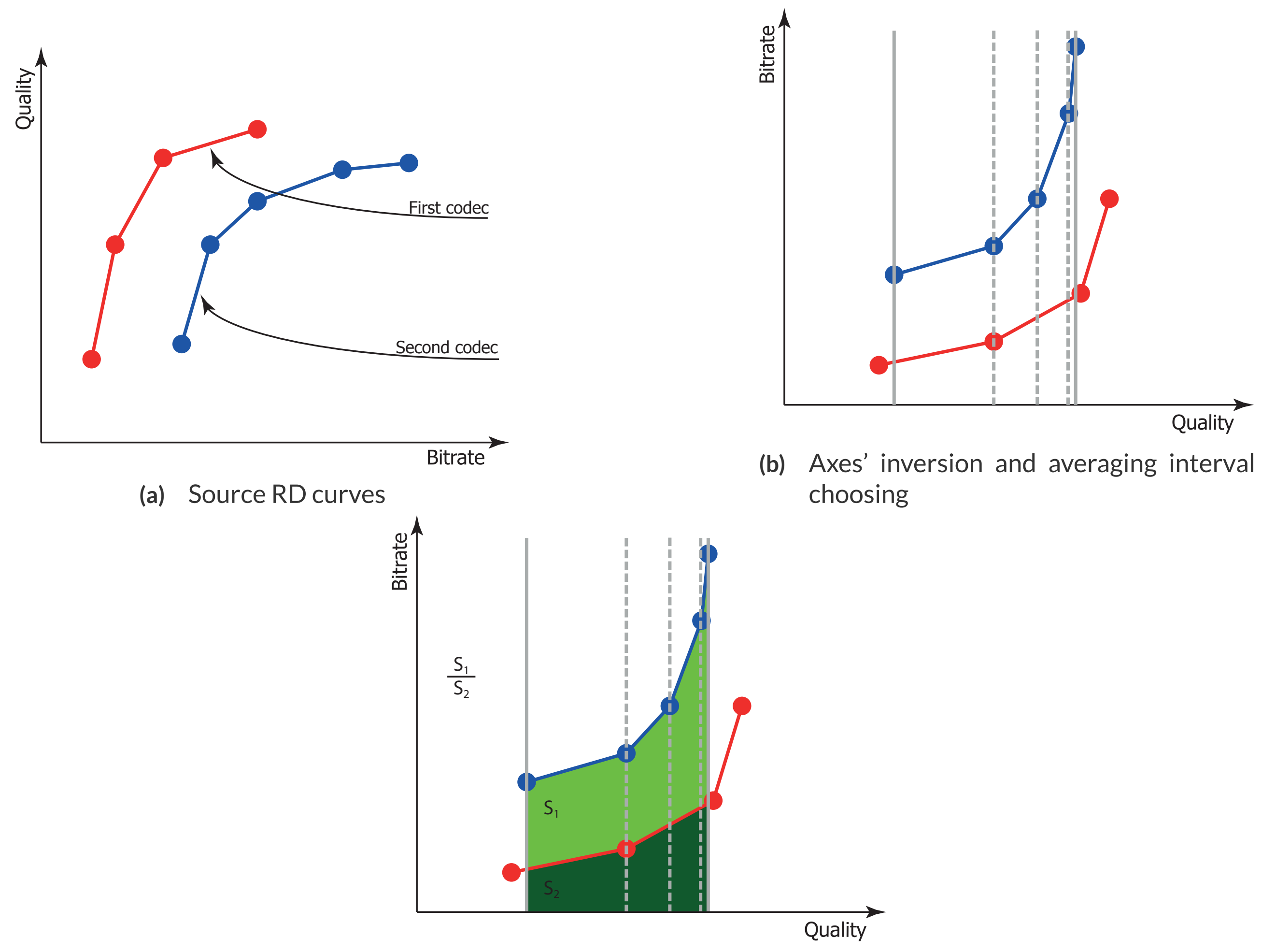


(b) Encoding speed (frames per second). "Green" codec is slower!



(c) Integral situation with codecs. This plot shows the situation more clearly

Figure 22: Speed/Quality trade-off example



(a) Source RD curves

(b) Axes' inversion and averaging interval choosing

(c) Areas under curves ratio

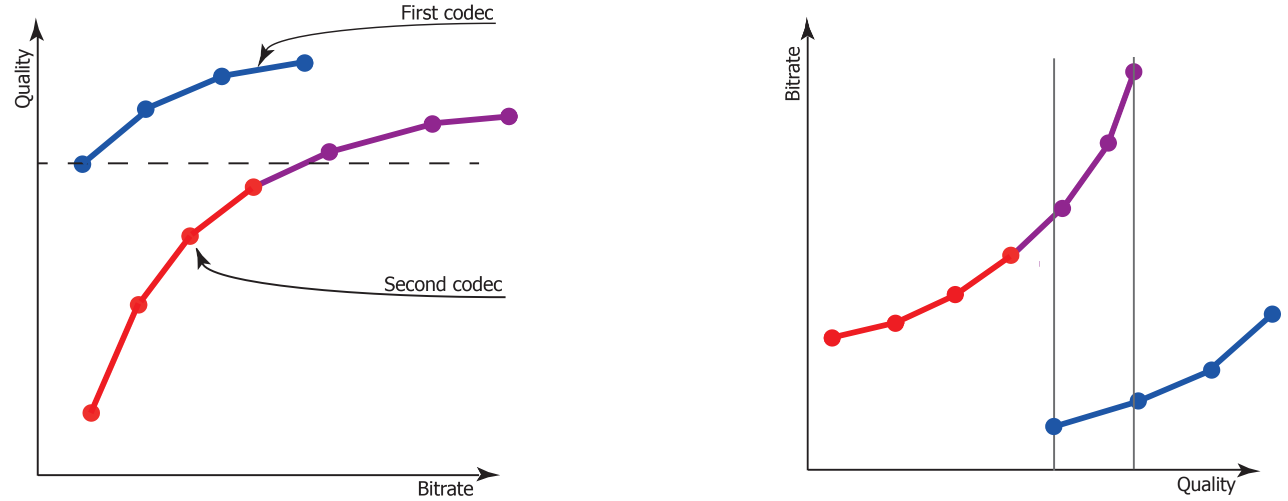
Figure 23: Average bitrate ratio computation

olation methods for classic RD curves; nevertheless, even linear methods are acceptable when interpolating RD curves.

The final step is calculation of the area under the curves in the chosen interpolation segment and determination of their ratio (see Figure 23c). This result is an average bitrate ratio at a fixed quality for the two codecs. When considering more than two codecs, one of is defined as a reference codec, and the quality of the others is compared with that of the reference.

D.4.1. When RD Curves Fail to Cross the Quality Axis

If no segment exists for which two codecs both produce encoding results, we measured the results for additional higher and/or lower bitrates. The schematic example (Figure 24) shows that the results for these extra bitrates (purple) cross with codec two and enable a comparison with codec one.



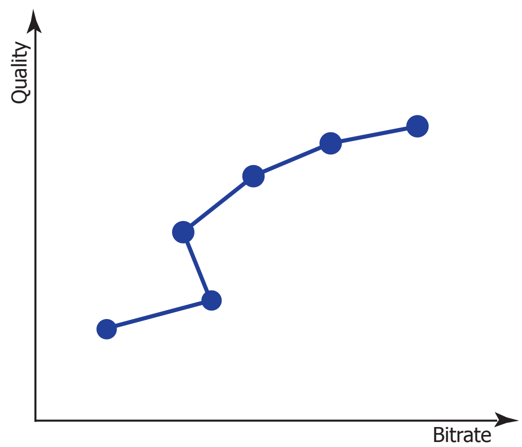
(a) Source RD curves, purple color indicates results for extra bitrates

(b) Axes' inversion and averaging interval choosing

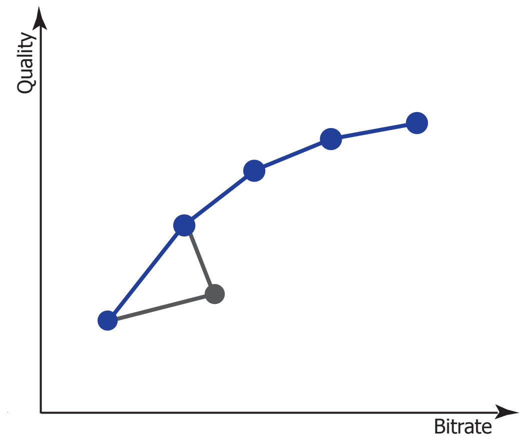
Figure 24: Measuring codec on additional bitrates to make it cross with other codecs over the quality axis.

D.4.2. When RD Curves Are Non-monotonic

Sometimes, especially on complex videos, the encoding results for neighboring bitrates vary greatly owing to the codec's operating characteristics. This situation leads to a non-monotone RD curve, which we process as follows: for each point, use the next point at the target bitrate that has greater or equal quality. This technique yields the reduced monotonic curve, which appears in the example of Figure 25.



(a) Non-monotonic RD-curve.



(b) Points that were used to calculate integral.

Figure 25: Processing non-monotonic RD-curves.

D.5. Relative Quality Analysis

Although most figures in this report provide codec scores relative to a reference encoder (i.e., x264), the "Relative Quality Analysis" sections provide the bitrate ratio at a fixed quality score (see Section D.4) for each codec pair.

This approach may be useful when comparing codec A relative with codec B only.

Below is a simplified example table showing the average bitrate ratio, given a fixed quality, for just two codecs.

	A	B
A	100% ☺	75% ☹
B	134% ☹	100% ☺

Table 3: Example of average bitrate ratio for a fixed quality table

Consider column "B", row "A" of the table, which contains the value 75%. This number should be interpreted in the following way: the average bitrate for Codec B at a fixed quality is 75% less than that for codec A. The icon in the cell depicts the confidence of this estimate. If projections of RD curves on the quality axis (see Figure 23) have large common areas, the cell contains a happy icon. If this overlapping area is small, and thus the bitrate-score calculation is unreliable, the cell contains a sad icon.

Plots of the average bitrate ratio for a fixed quality are visualizations of these tables. Each line in the plot depicts values from one column of the corresponding table.

E. OBJECTIVE-QUALITY METRIC DESCRIPTION

E.1. SSIM (Structural Similarity)

We used the YUV-SSIM objective-quality metric in this report to assess the quality of encoded video sequences. We compute YUV-SSIM as the weighted average of SSIM values for each channel individually (Y-SSIM, U-SSIM and V-SSIM):

$$\text{YUV-SSIM} = \frac{4\,\text{Y-SSIM} + \text{U-SSIM} + \text{V-SSIM}}{6}. \tag{1}$$

Below is a brief description of SSIM computation.

E.1.1. Brief Description

Wang, et al.[2] published the original paper on SSIM. This paper available at http://ieeexplore.ieee.org/iel5/83/28667/01284395.pdf. The SSIM author homepage is http://www.cns.nyu.edu/~lcv/ssim/

The main idea that underlies the structural-similarity (SSIM) index is comparison of the distortion of three image components:

- Luminance
- Contrast
- Structure

The final formula, after combining these comparisons, is

$$\textbf{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x + \mu_y + C_1)(\sigma_x + \sigma_y + C_2)}, \tag{2}$$

where

$$\mu_x = \sum_{i=1}^{N} \omega_i x_i, \tag{3}$$

$$\sigma_x = \sqrt{\sum_{i=1}^{N} \omega_i (x_i - \mu_x)}, \tag{4}$$

$$\sigma_{xy} = \sum_{i=1}^{N} \omega_i (x_i - \mu_x)(y_i - \mu_y). \tag{5}$$

Finally, $C_1 = (K_1 L)^2$ and $C_2 = (K_2 L)^2$, where L is the dynamic range of the pixel values (e.g. 255 for 8-bit greyscale images), and $K1, K2 \ll 1$.

We used $K_1 = 0.01$ and $K_2 = 0.03$ were used for the comparison presented in this report, and we filled the matrix with a value "1" in each position to form a filter for the results map.

[2] Zhou Wang, Alan Conrad Bovik, Hamid Rahim Sheikh and Eero P. Simoncelli, "Image Quality Assessment: From Error Visibility to Structural Similarity," IEEE Transactions on Image Processing, Vol. 13, No. 4, April 2004.

For our implementation, one SSIM value corresponds to two sequences. The value is in the range $[-1, 1]$, with higher values being more desirable (a value of 1 corresponds to identical frames). One advantage of the SSIM metric is that it better represents human visual perception than does PSNR. SSIM is more complex, however, and takes longer to calculate.

E.1.2. Examples

Figure 26 shows an example SSIM result for an original and processed (compressed with lossy compression) image. The value of 0.9 demonstrates that the two images are very similar.

(a) Original (b) Compressed (c) SSIM

Figure 26: SSIM example for compressed image

Figure 27 depicts various distortions applied to the original image, and Figure 28 shows SSIM values for these distortions.

(a) Original image

(b) Image with added noise

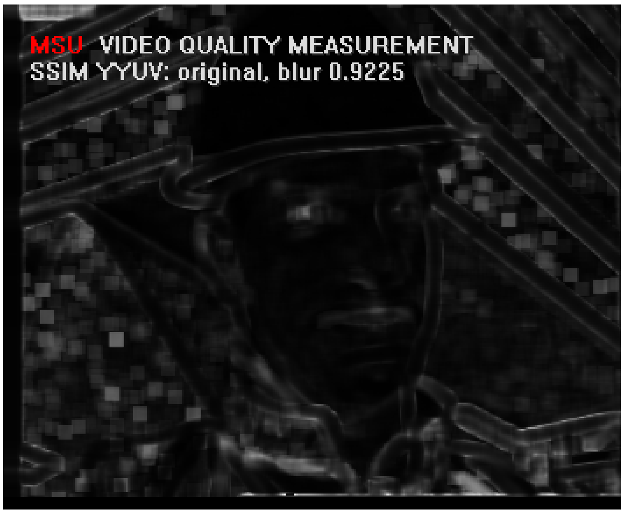
(c) Blurred image

(d) Sharpen image

Figure 27: Examples of processed images

(a) SSIM map for original image, SSIM $= 1$

(b) SSIM map for noisy image, SSIM $= 0.552119$



(c) SSIM map for blurred image, SSIM $= 0.9225$

(d) SSIM map for sharpen image, SSIM $= 0.958917$

Figure 28: SSIM values for original and processed images

E.1.3. Measurement method

We used the MSU Video Quality Measurement Tool (VQMT) to calculate objective metrics for the encoded streams. The tool can be downloaded or purchased at http://compression.ru/video/quality_measure/vqmt_download.html#start.

Run the command

```
vqmt -in "{original_yuv}" IYUV {width}×{height} -in "decoded_yuv" IYUV
{width}×{height} metrics_list -subsampling -json -json_file "{json_filename}" -threads
3
```

where `input_yuv` is the encoded stream name, `width` and `height` are the size of encoded stream in pixels, `metrics_list` is a list of metrics to measure (e.g., "-metr ssim_precise YYUV -metr ssim_precise UYUV -metr ssim_precise VYUV"), and `json_filename` is the name of the output file containing the metric results.

F. ABOUT THE GRAPHICS & MEDIA LAB VIDEO GROUP

Graphics & Media Lab
Video Group

The Graphics & Media Lab Video Group is part of the Computer Science Department of Lomonosov Moscow State University. The Graphics Group began at the end of 1980's, and the Graphics & Media Lab was officially founded in 1998. The main research avenues of the lab include areas of computer graphics, computer vision and media processing (audio, image and video). A number of patents have been acquired based on the lab's research, and other results have been presented in various publications.

The main research avenues of the Graphics & Media Lab Video Group are video processing (pre- and post-, as well as video analysis filters) and video compression (codec testing and tuning, quality metric research and codec development).

The main achievements of the Video Group in the area of video processing include:

- High-quality industrial filters for format conversion, including high-quality deinterlacing, high-quality frame rate conversion, new, fast practical super resolution and other processing tools.
- Methods for modern television sets, such as a large family of up-sampling methods, smart brightness and contrast control, smart sharpening and more.
- Artifact removal methods, including a family of denoising methods, flicking removal, video stabilization with frame edge restoration, and scratch, spot and drop-out removal.
- Application-specific methods such as subtitle removal, construction of panorama images from video, video to high-quality photo conversion, video watermarking, video segmentation and practical fast video deblur.

The main achievements of the Video Group in the area of video compression include:

- Well-known public comparisons of JPEG, JPEG-2000 and MPEG-2 decoders, as well as MPEG-4 and annual H.264 codec testing; codec testing for weak and strong points, along with bug reports and codec tuning recommendations.
- Video quality metric research; the MSU Video Quality Measurement Tool and MSU Perceptual Video Quality Tool are publicly available.
- Internal research and contracts for modern video compression and publication of MSU Lossless Video Codec and MSU Screen Capture Video Codec; these codecs have one of the highest available compression ratios.

The Video Group has also worked for many years with companies like Intel, Samsung and RealNetworks.

In addition, the Video Group is continually seeking collaboration with other companies in the areas of video processing and video compression.

E-mail: video@graphics.cs.msu.ru

MSU Video Quality Measurement Tool

MSU Graphics & Media Lab. Video Group.

3 reasons to use VQMT:

Fastest implementation of VMAF
Fastest SSIM/MS-SSIM speed on 4K/8K video
Professional video analysis with NIQE and artifact metrics

video-measure@compression.ru

1. Widest Range of Metrics & Formats

1.1 20+ Objective Metrics

PSNR several versions	Spatio-Temporal SSIM
MSAD	MSU Blurring Metric
Delta	MSU Brightness Flicking Metric
MSE	MSU Brightness Independent PSNR
VQM	MSU Drop Frame Metric
SSIM	MSU Noise Estimation Metric
MS-SSIM	MSU Scene Change Detector
3-SSIM	MSU Blocking Metric
VMAF	**NIQE (no-reference comparison)**

1.2 HDR support

1.3 Hundreds Video and 30+ Image Formats

All popular video codecs, including **H264** and **HEVC**.

Special support for: RAW, Y4M, AviSynth, PXM.

All popular image formats: PNG, JPEG, TIFF (**with HDR support**), EXR, BMP, PSD, and others

1.4 2k, 4k, 8k support

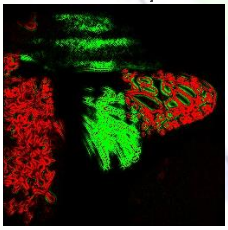
2. Fastest Video Quality Measurement

2.1 Up to 11.7x faster calculation of metrics with GPU (CUDA & OpenGL support)

2.2 Multi-core Processors Support

Visualization Examples

Allows easily detect where codec/filter fails



MSU Blurring Metric

MSU Blocking Metric

VQMT average Speedup

3. Easy Integration

3.1 Linux support

DEB & RPM packages

3.2 Batch Processing with JSON and CSV output

3.3 Plugins SDK

4. Professional Analysis

4.1 Comparative Analysis

4.2 Metric Visualization

MSU VQMT Official Page

Tool was downloaded more than 200 000 times!

Free and Professional versions are available

Big thanks to our contributors:

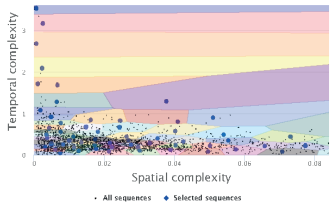
Reduce video file size or encoding speed with optimal codec settings

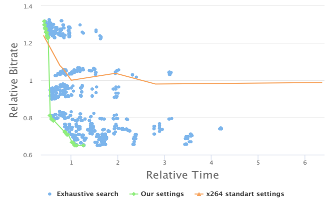
Effective Video Transcoding

For almost 14 years, Lomonosov MSU Graphics&Media Lab's video group has been conducting video codecs comparisons. We know that almost always there is a possibility to find efficient encoding options for every video

We created a representative dataset of **385 videos** chosen from **9000+ FullHD&4K** videos

12 million encoder launches were done on Intel Xeon E3-1125v3

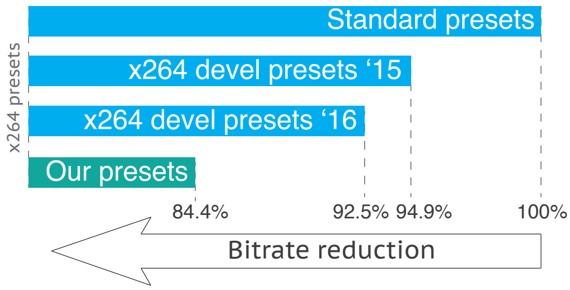




Full-size charts are available on our [project page]

15% bitrate savings in average

Encoding presets determined by our method beats x264 developers' presets with keeping encoding time and encoded video quality



Percentage of file size reduction in average for a representative dataset of 77 videos

We developed a way to find optimal presets for a large number of video classes

Everything is fair! We don't declare an "up-to-x%" bitrate reduction — average file size reduction is 15% higher comparing to standard x264 presets

We find presets that **do not reduce encoding speed and objective quality of encoded video**
You give limitations, and we guarantee the same or higher objective quality and encoding speed

You use standard presets and don't believe that it will work for your videos? Give us a chance — request a demo, for free!

We can find best presets for your videos

1. **Your video** — send us uncompressed video and your preset
2. **Report** — get a report with optimal presets for your video and their gain
3. **Choose and pay** — we offer additional options for better compression and analysis
4. **Get preset** and encode similar videos with it or **Get video** compressed with chosen preset

Subjective comparisons
Receive subjective quality comparison results for your videos

Codec analysis
Find out strong and weak parts of your codec

Saliency-adaptive encoding
Bitrate savings given by adaptive encoding of salient regions

Gaze maps construction
Raw viewers' gaze points on your video

Encoding with extremely low bitrates
Get your video of highest quality for low bitrates

4K and 360-degree encoding
Best presets for high-quality formats encoding

contact **evt@compression.ru** to get them!

Our project page [compression.ru/video/video_codec_optimization/]

In cooperation with Lomonosov MSU Graphics&Media Lab

GRAPHICS & MEDIA LAB VIDEO GROUP