MSU Video Codec Comparison 2017

Part I: FullHD Content, Objective Evaluation

Minory revised on February 25, 2018

Graphics & Media Lab
Video Group

Video group head: Dr. Dmitriy Vatolin

Project head: Dr. Dmitriy Kulikov

Measurements & analysis: Dr. Mikhail Erofeev, Stanislav Dolganov, Sergey Zvezdakov

Codecs:

H.265

- Kingsoft HEVC Encoder
- nj265
- x265

Non H.265

- uAVS2
- nj264
- SIF Encoder
- x264

CS MSU Graphics & Media Lab, Video Group

August 23, 2017

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

`videocodec-testing@graphics.cs.msu.ru`

Contents

1 ACKNOWLEDGMENTS

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

- SIF developer team
- AVS2 developer team
- Nanjing Yunyan
- Kingsoft

The Video Group would also like to thank these companies for their help and technical support during the tests.

2 OVERVIEW

2.1 Sequences

	Sequence	Number of frames	Frame rate	Resolution
1.	Animation	833	24	1920×1080
2.	Apple Tree	338	30	1920×1080
3.	Behind Expedition	1047	30	1920×1080
4.	Cemetry	999	25	1920×1080
5.	Christmas Cats	1500	25	1920×1080
6.	Chronicle	1113	30	1920×1080
7.	City Crowd	763	30	1920×1080
8.	Coffee Beans	1005	24	1920×1080
9.	Color Tune	1049	25	1920×1080
10.	Crowd Run	500	50	1920×1080
11.	Disneyland	317	24	1920×1080
12.	Fire	601	25	1920×1080
13.	Forest Dog	976	25	1920×1080
14.	Fountain	516	25	1920×1080
15.	Gilmour	957	30	1920×1080
16.	Housing Group	1007	24	1920×1080
17.	Infinit	258	25	1920×1080
18.	Innershaq	1569	24	1920×1080
19.	Italian History	989	24	1920×1080
20.	Mountain Bike	1063	24	1920×1080
21.	Real Voters	997	24	1920×1080
22.	Road Runner	999	24	1920×1080
23.	Roseman Bridge	2549	30	1920×1080
24.	Sea Lions	1293	24	1920×1080
25.	Shakewalk	805	25	1920×1080
26.	Sita	1000	25	1920×1080
27.	Skiers	1370	60	1920×1080
28.	Steadicam	979	24	1920×1080

29.	Twin Strangers	1026	25	1920×1080
30.	Wedding	948	24	1920×1080
31.	Ziguinchor	994	25	1920×1080

TABLE 1: Summary of video sequences

Brief descriptions of the sequences used in our comparison are given in Table 1. More detailed descriptions of these sequences can be found in Appendix B.

2.2 Codecs

	Codec	Developer	Version
1.	uAVS2	Digital Media R&D Center, Peking University, Shenzhen Graduate School	V1.0
2.	Kingsoft HEVC Encoder	Kingsoft	V2.5.2
3.	nj264	Nanjing Yunyan Email: jtwen@tsinghua.edu.cn	V1.0
4.	nj265	Nanjing Yunyan Email: jtwen@tsinghua.edu.cn	V1.0
5.	SIF Encoder	SIF Encoder Team Email: info@sifcodec.com	1.43.0
6.	x264	x264 Developer Team	r2833 df79067
7.	x265	MulticoreWare, Inc.	2.3+23-97435a0870befe35

TABLE 2: Short codecs' descriptions

Brief descriptions of the codecs used in our comparison are given in Table 2. x264 was used as a good quality AVC reference codec for comparison purposes. Detailed descriptions of all codecs used in our comparison can be found in Appendix D.

3 OBJECTIVES AND TESTING RULES

In this report we compare encoding quality of recently emerged HEVC encoders and encoders of other standards using objective assessment methods. 31 video sequences with 1080p resolution were used to evaluate performance of codecs under comparison. To choose these sequences we analyzed 512,000 video sequences and selected representative set (the detailed description of selection process is given in Appendix C).

Our comparison consists of three parts corresponding to various encoders' use cases: Fast encoding, Universal encoding, Ripping encoding. For each use case encoder developers had an option to provide encoding parameters to be used in our tests. If no parameters were provided, we either used the same parameters as were used in prior study or, if no prior parameters were available, did our best effort to choose good parameters ourselves. Nevertheless, the chosen parameters had to satisfy minimum speed requirements of the use case:

- Fast encoding—60 FPS
- Universal encoding—25 FPS
- Ripping encoding—1 FPS; for Ripping use case an extra requirement was imposed: encoder with selected parameters had to outperform x264 with "versyslow" preset according to SSIM quality scores.

Computer with the following configuration was used to run codecs under comparison: Core i7 6700K (Skylake) @ 4Ghz, RAM 8 GB, Windows 8.1. For objective quality measurements we used YUV-SSIM quality metric (see Appendix F.1).

4 FAST ENCODING

4.1 RD curves

Next figures show RD curves for video sequences on fast transcoding use case. KingSoft encoder has the best mean quality score, nevertheless it isn't absolute leader for each and every sequence: for example, AVS2 is in first place at Coffee Beans sequence, x265—at Fire, nj265—at Steadicam. Moreover, the one can notice that encoding quality drastically depends on video sequence, while the top SSIM score achieved at Coffee Beans sequence at *lowest* bitrate is 0.96, the top score achieved at Christmas Cats sequence at *highest* bitrate is only 0.82.

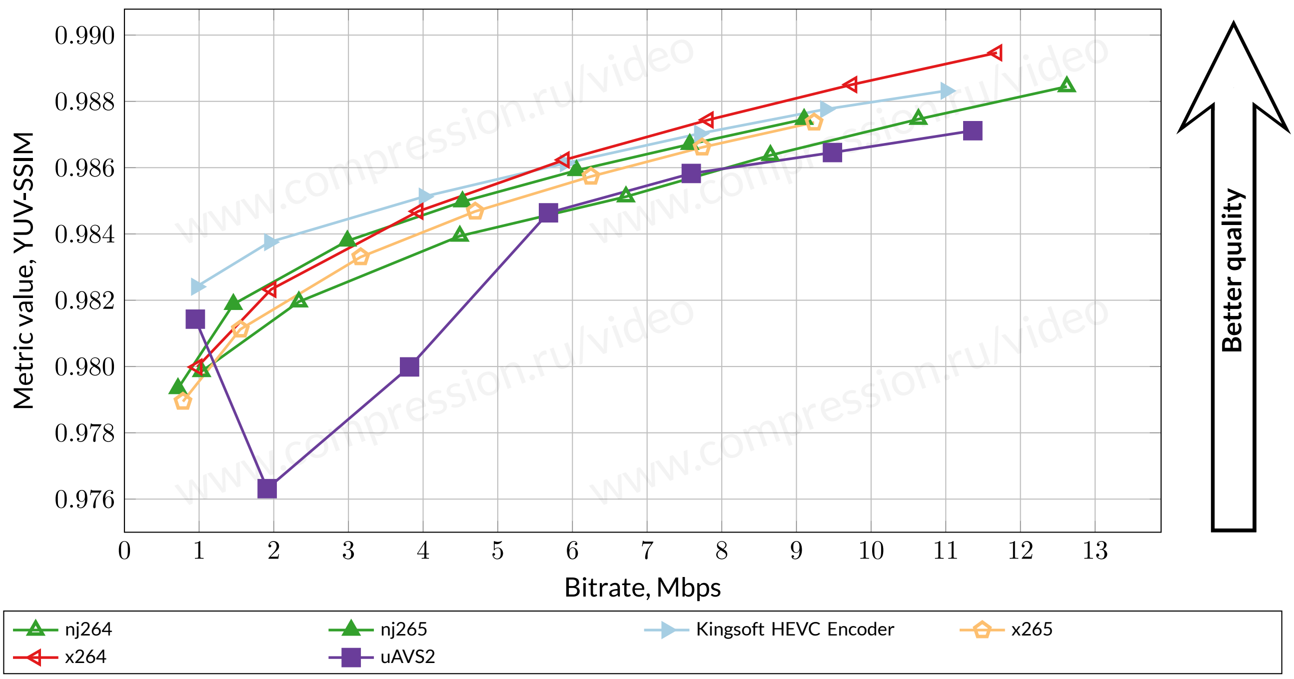


FIGURE 1: Bitrate/quality—use case "Fast Encoding," Color Tune sequence, YUV-SSIM metric

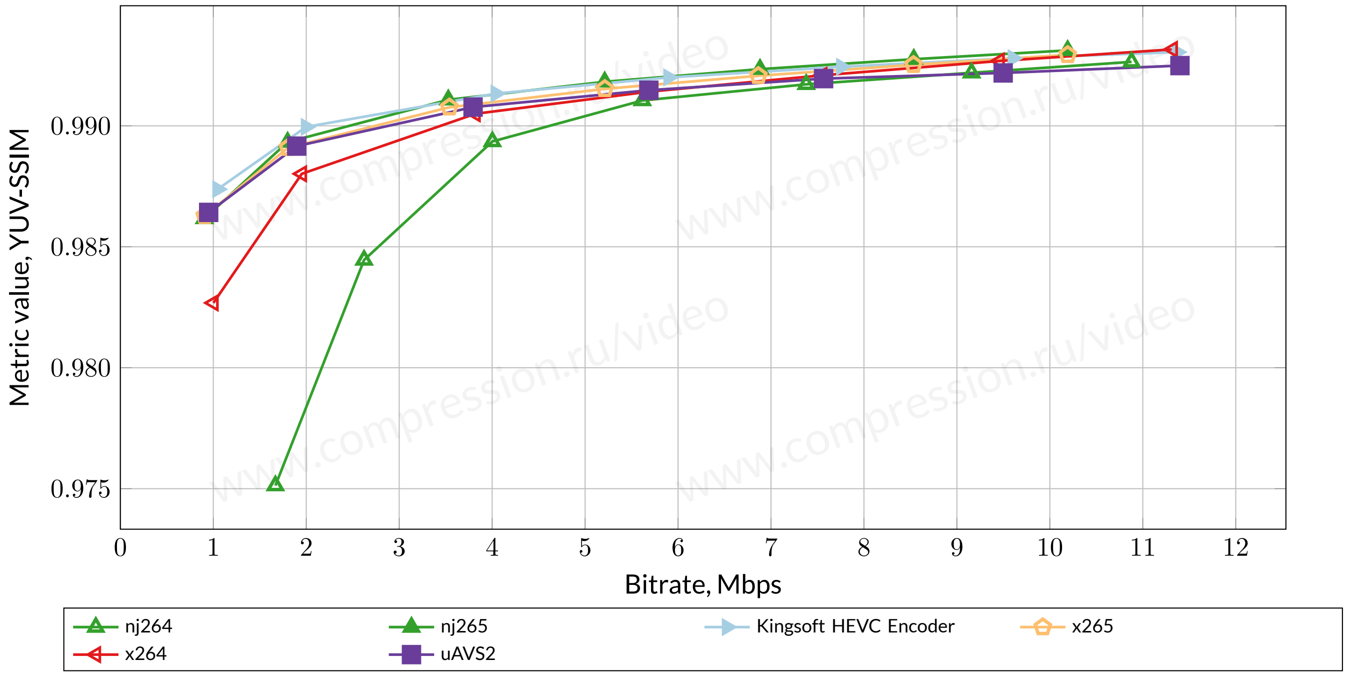


FIGURE 2: Bitrate/quality—use case "Fast Encoding," Steadicam sequence, YUV-SSIM metric

4.2 Encoding Speed

Figures below show difference in encoding speed among participating codecs. In these figures (as in Section 4.1) different codecs take the first place at different sequences. Therefore, we can identify the leader according to mean speed scores only. In this nomination the first place goes to KingSoft. Nevertheless, KingSoft isn't the absolute winner: for example, nj264 is 20% faster than KingSoft at Christmas Cats and Crowd Run sequences, AVS2 is 12% faster at Fire sequence etc.

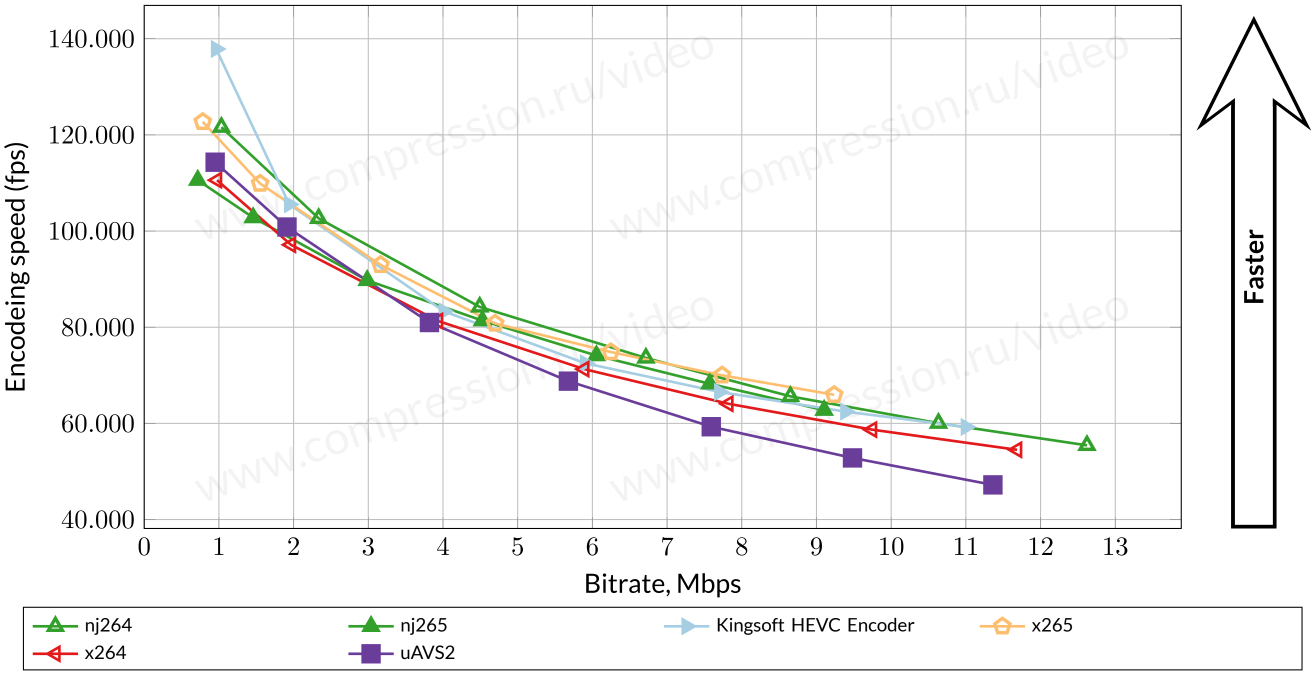


FIGURE 3: Encoding speed—use case "Fast Encoding," Color Tune sequence

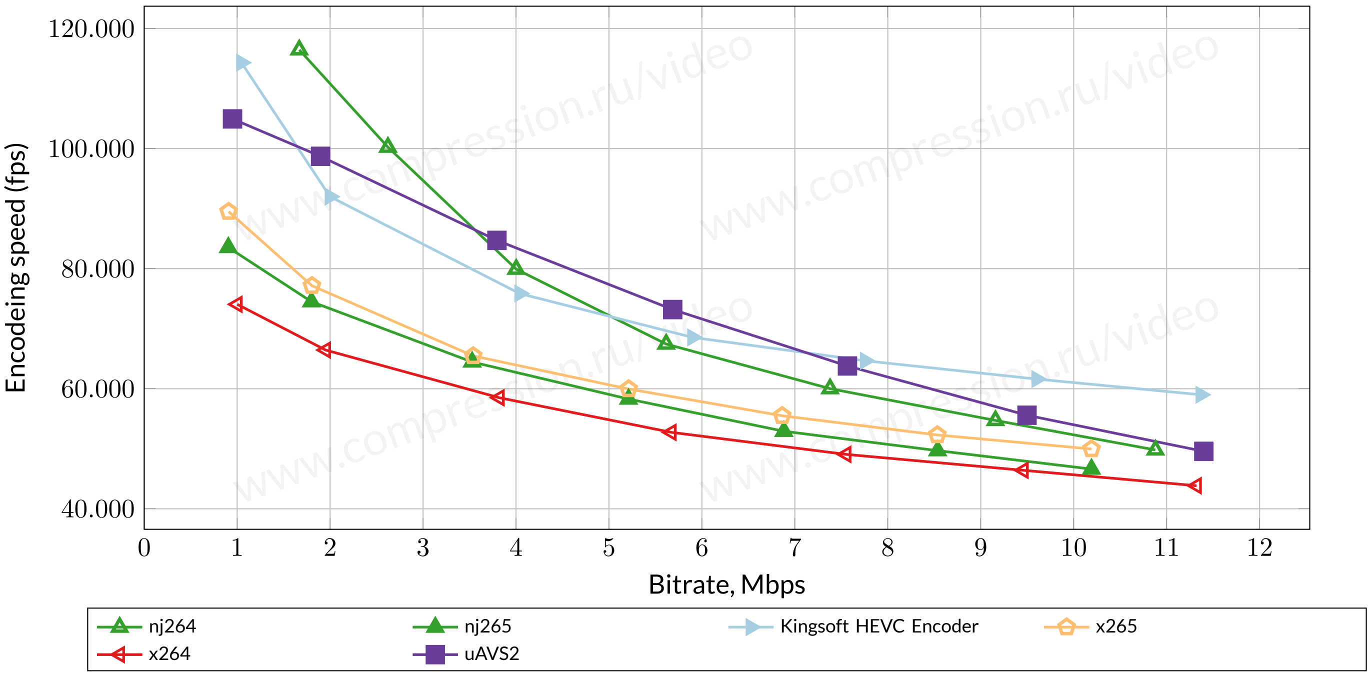


FIGURE 4: Encoding speed—use case "Fast Encoding," Steadicam sequence

4.3 Speed/Quality Trade-Off

Detailed descriptions of the speed/quality trade-off graphs can be found in Appendix E. Sometimes, codec results are not present in the particular graph owing to the codec's extremely poor performance (i.e. the codec's RD curve has no intersection with the reference's RD curve).
The speed/quality trade-off graphs show both relative quality and speed scores of encoders under comparison. Since x264 was chosen as reference codec in our comparison, we normalized all scores using x264 scores.

For fast encoding use case there is only one Pareto optimal encoder in terms of mean speed and quality scores—Kingsoft HEVC encoder. "Pareto optimal" encoder means there is no encoder faster and better than it in this test. Notably there is no encoder at second place according to speed/quality trade-off: if we exclude Kingsoft HEVC encoder from this plot, any codec left would be outperformed by its competitor by either speed or quality (e.g. AVS2 has higher mean quality, but is slower than nj264). There are slight differences for particular sequences: for example, nj264 is the only Pareto optimal encoder for Christmas Cats sequence, AVS2 is the best encoder for Coffee Beans sequence and Infinit sequence has 3 Pareto optimal encoders: KingSoft, x265 and AVS2.

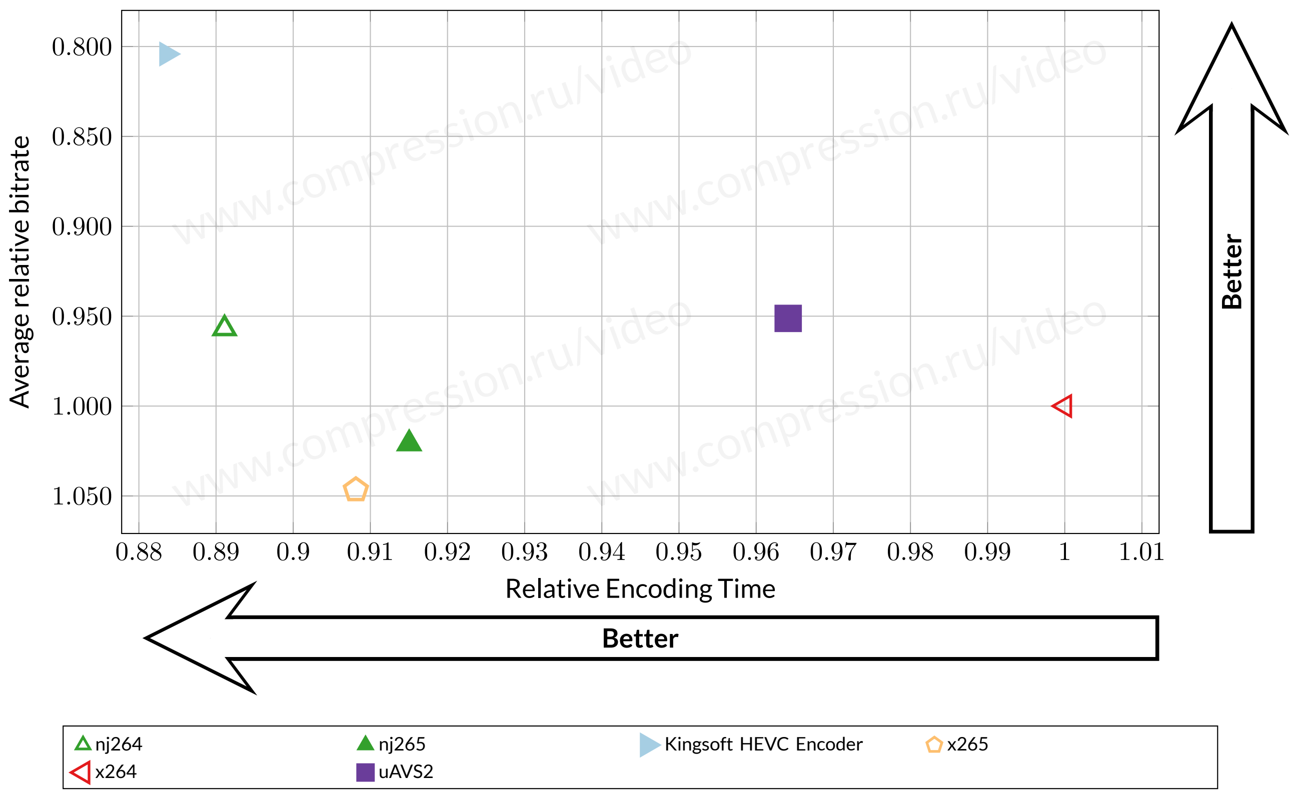


FIGURE 5: Speed/quality trade-off—use case "Fast Encoding," all sequences, YUV-SSIM metric

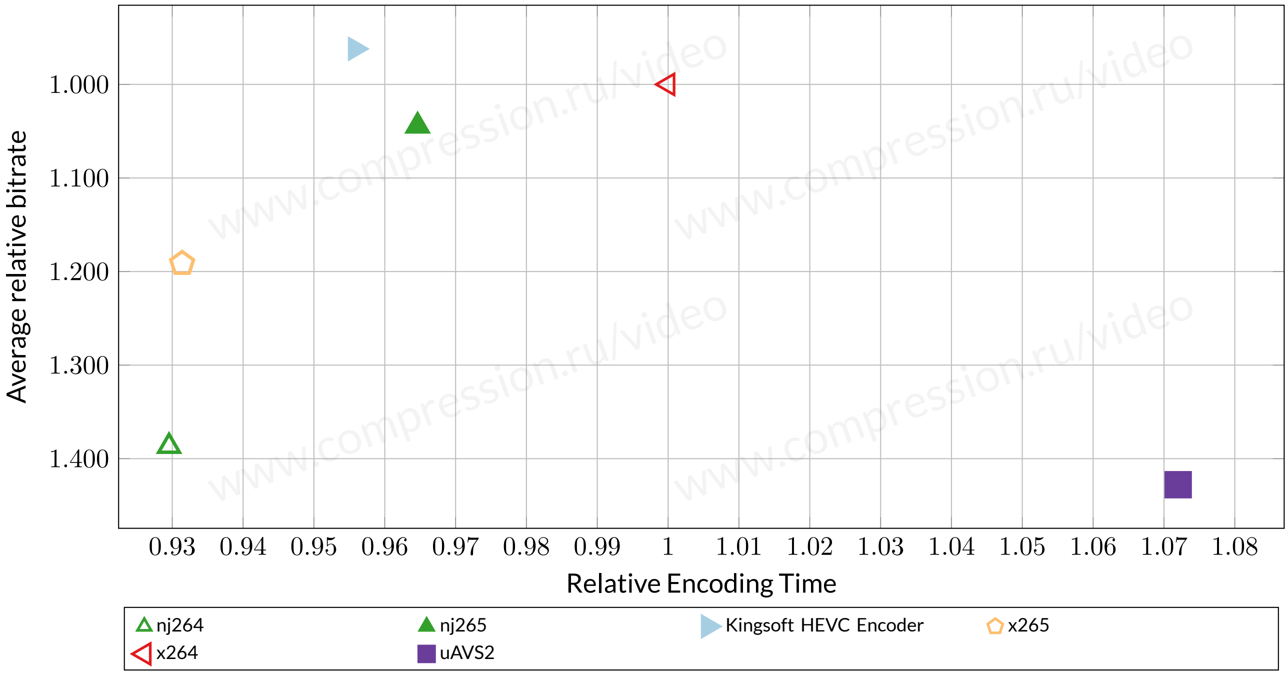


FIGURE 6: Speed/quality trade-off—use case "Fast Encoding," Color Tune sequence, YUV-SSIM metric

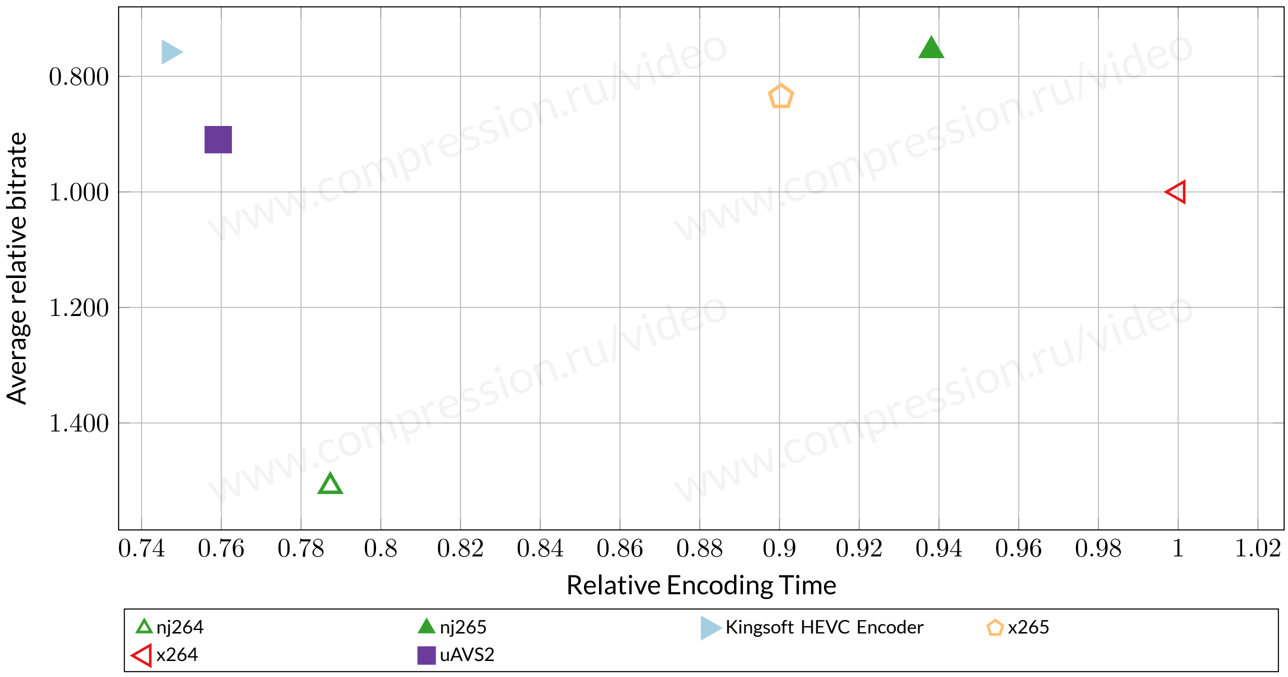


FIGURE 7: Speed/quality trade-off—use case "Fast Encoding," Steadicam sequence, YUV-SSIM metric

4.4 Bitrate Handling

The plots below show how accurately encoded stream's real bitrate matches bitrate requested by user. Almost all encoders handle bitrate well, but there are issues for some encoders at some sequences, e.g. nj264 slightly undershoots target bitrate, AVS2 has some issues at Coffee Beans sequence, x265 lowers target bitrate at some sequences etc.

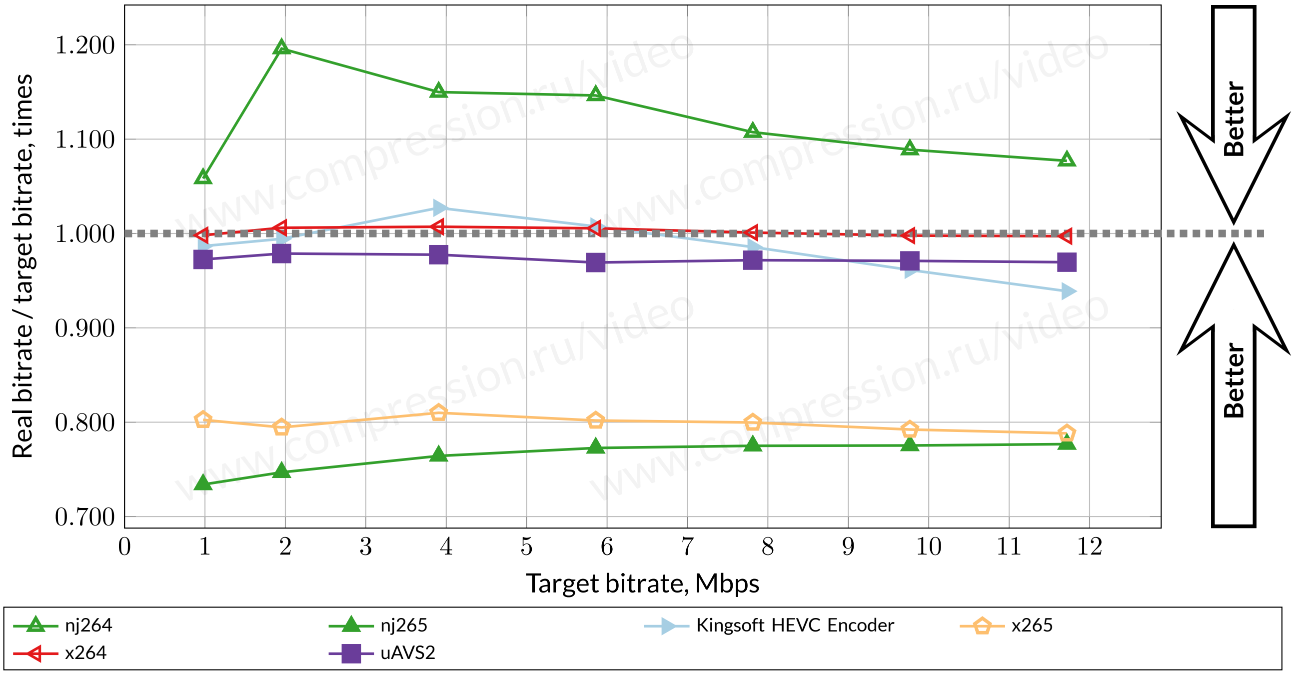


FIGURE 8: Bitrate handling—use case “Fast Encoding,” Color Tune sequence

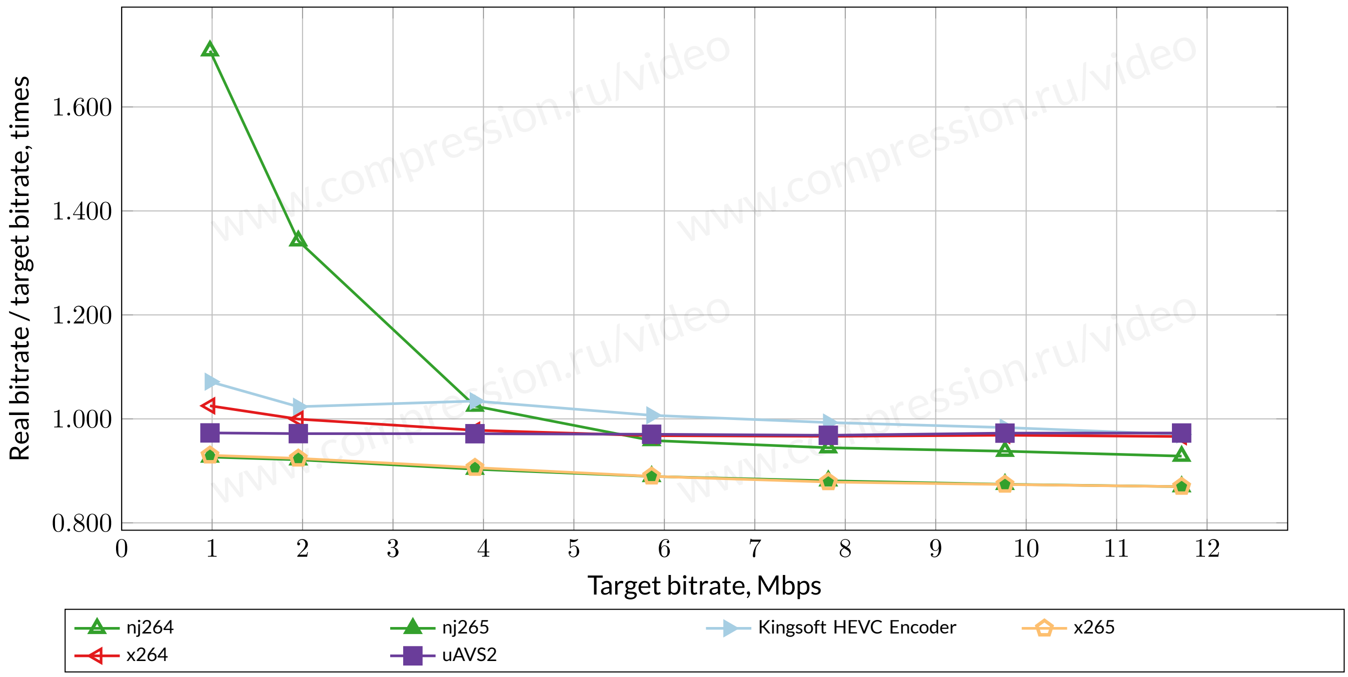


FIGURE 9: Bitrate handling—use case “Fast Encoding,” Steadicam sequence

4.5 Relative Quality Analysis

Note that each number in the tables below corresponds to some range of bitrates (see Appendix E). 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

	nj264	nj265	Kingsoft HEVC Encoder	x265	x264	uAVS2
nj264	100% ☺	112% ☺	83% ☺	118% ☺	105% ☺	103% ☺
nj265	98% ☺	100% ☺	77% ☺	104% ☺	98% ☺	94% ☺
Kingsoft HEVC Encoder	125% ☺	136% ☺	100% ☺	144% ☺	124% ☺	127% ☺
x265	96% ☺	98% ☺	75% ☺	100% ☺	96% ☺	91% ☺
x264	99% ☺	107% ☺	82% ☺	113% ☺	100% ☺	101% ☺
uAVS2	105% ☺	112% ☺	83% ☺	115% ☺	105% ☺	100% ☺

Confidence ☹ 0% — 😐 50% — ☺ 100%

TABLE 3: Average bitrate ratio for a fixed quality—use case "Fast Encoding," YUV-SSIM metric

Figure below depicts the data from the table above. Each line in the figure corresponds to one codec. Values on the vertical axis are the average relative bitrates compared with the codecs along the horizontal axis. A lower bitrate indicates better relative results.

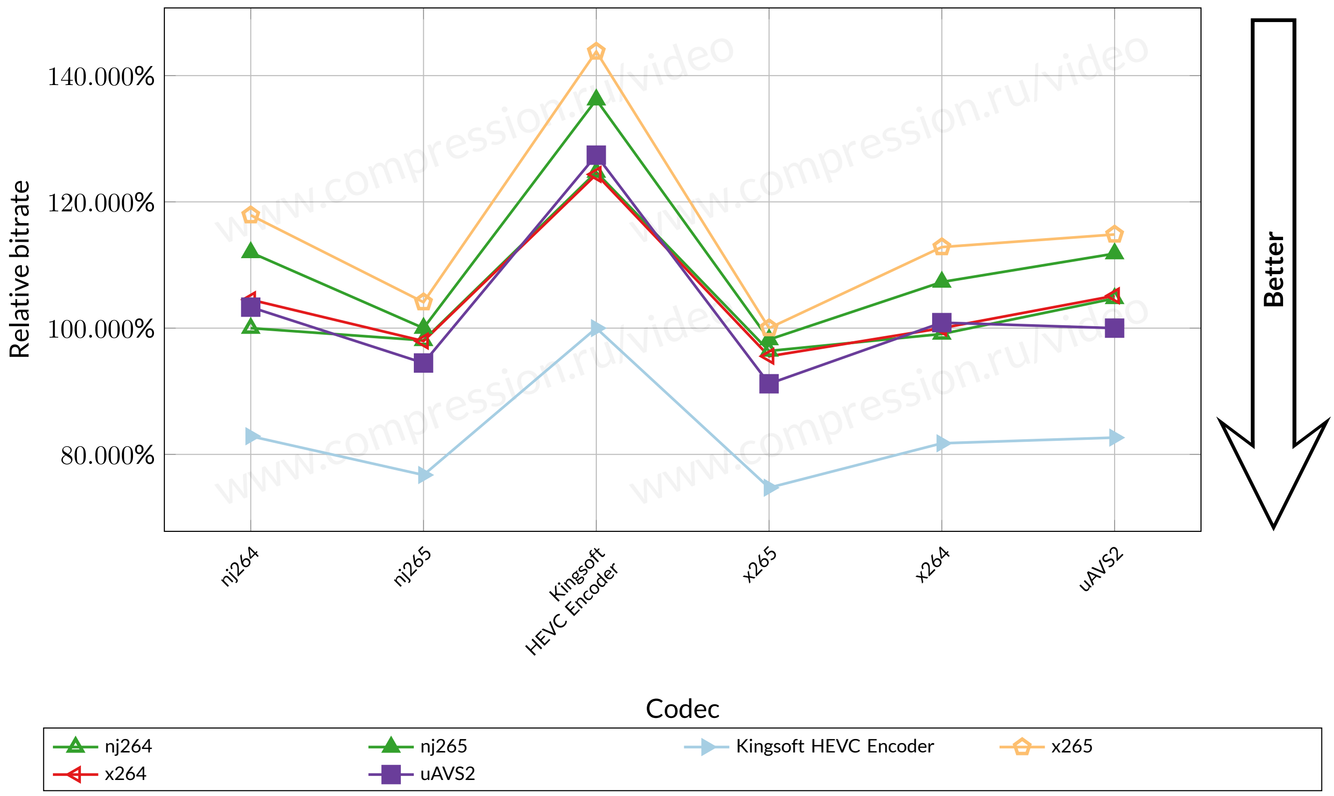


FIGURE 10: Average bitrate ratio for a fixed quality—use case "Fast Encoding," YUV-SSIM metric

5 UNIVERSAL ENCODING

5.1 RD curves

Similarly to fast encoding use case (see Section 4.1), there is no absolute leader for all sequences. KingSoft is the leader according to mean quality score, but for several sequences x265 encoder is the best (e.g. at Cristmas Cats, Infint, Stedicam sequences).

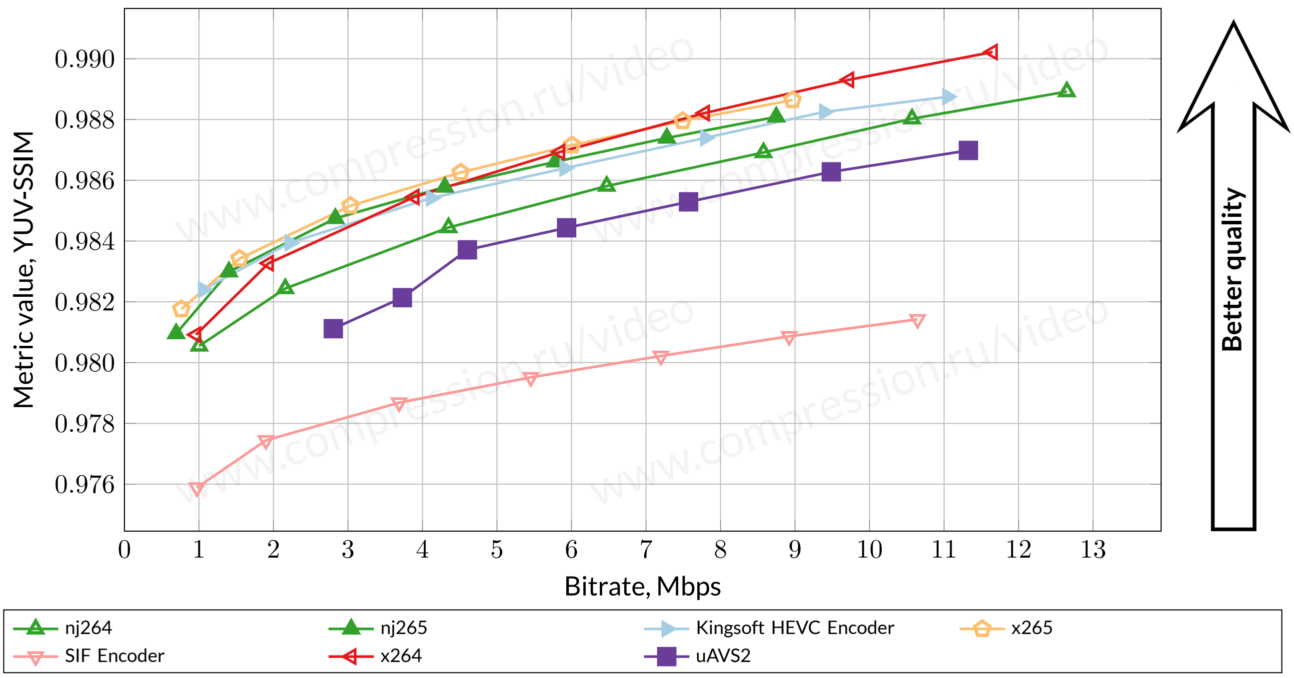


FIGURE 11: Bitrate/quality—use case "Universal Encoding," Color Tune sequence, YUV-SSIM metric

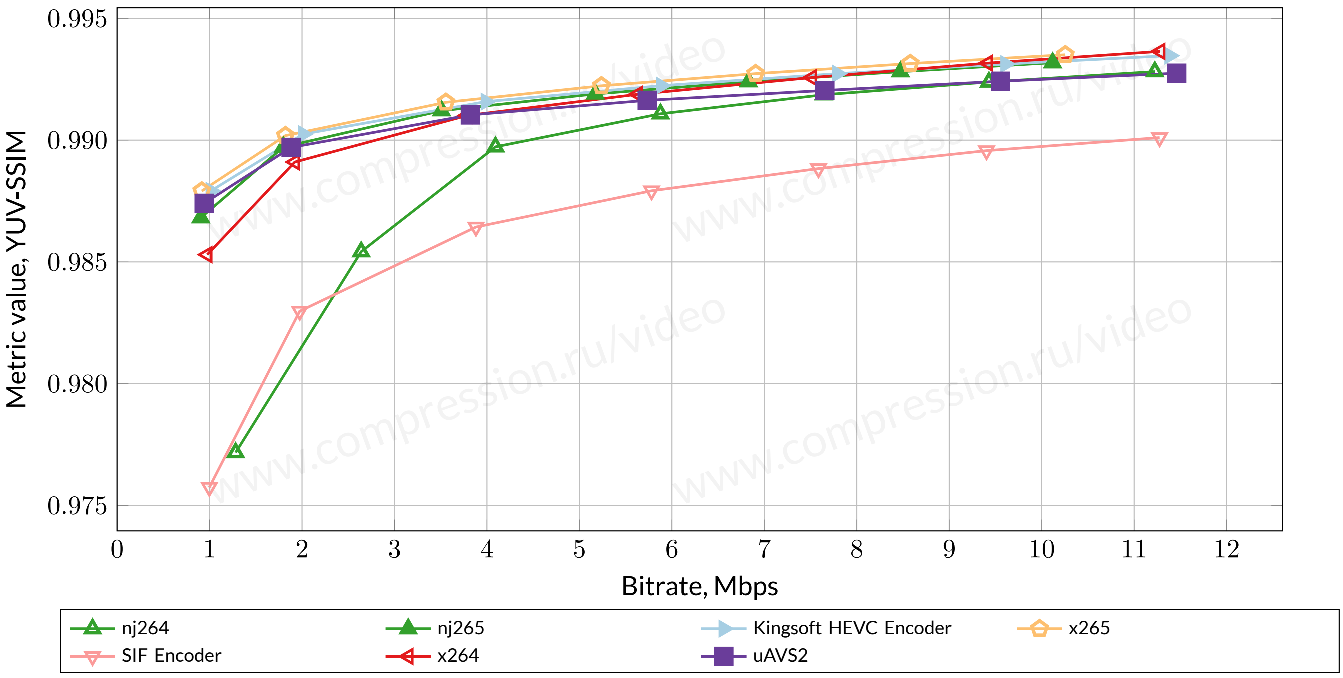


FIGURE 12: Bitrate/quality—use case "Universal Encoding," Steadicam sequence, YUV-SSIM metric

5.2 Encoding Speed

Figures below show difference in encoding speed among participating codecs. Similarly to Section 5.1, no absolute leader by encoding speed can be named. AVS2 has the best mean speed scores, but it isn't the fastest option for all sequences: for example, x265 is 35% faster than AVS2 at Christmas Cats and Crowd Run sequences, nj265 is 43% faster at Infinit sequence etc.

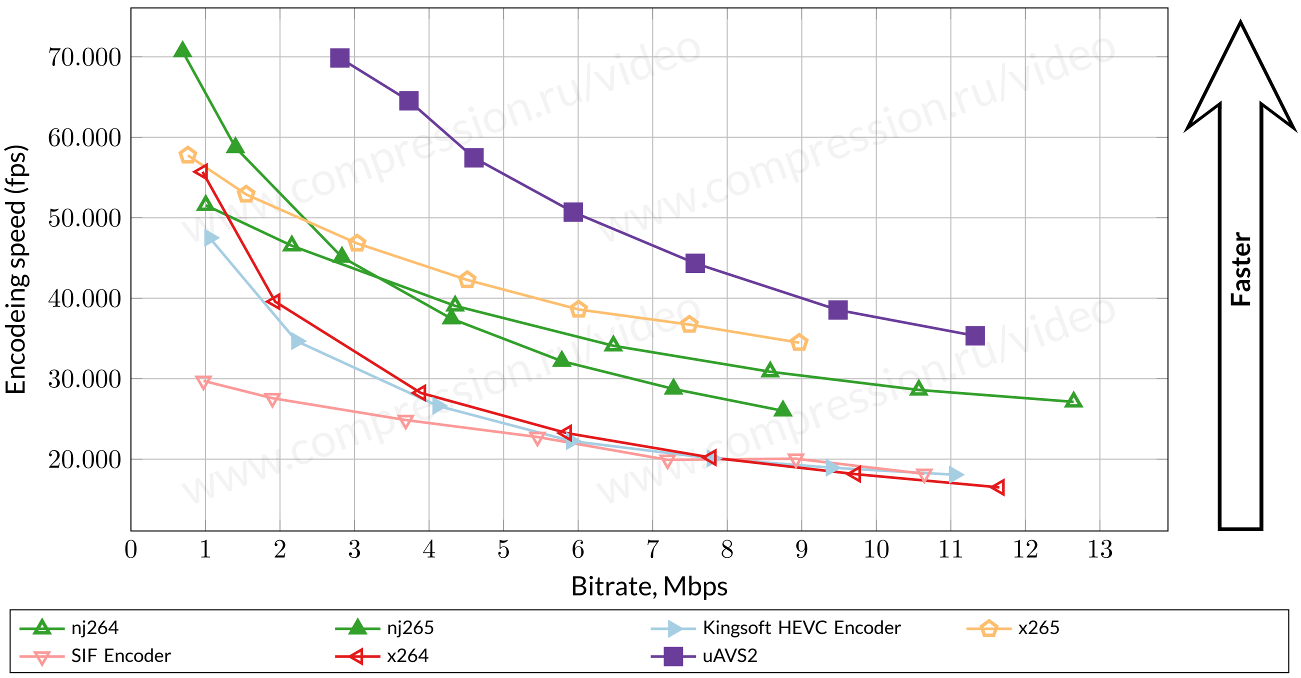


FIGURE 13: Encoding speed—use case "Universal Encoding," Color Tune sequence

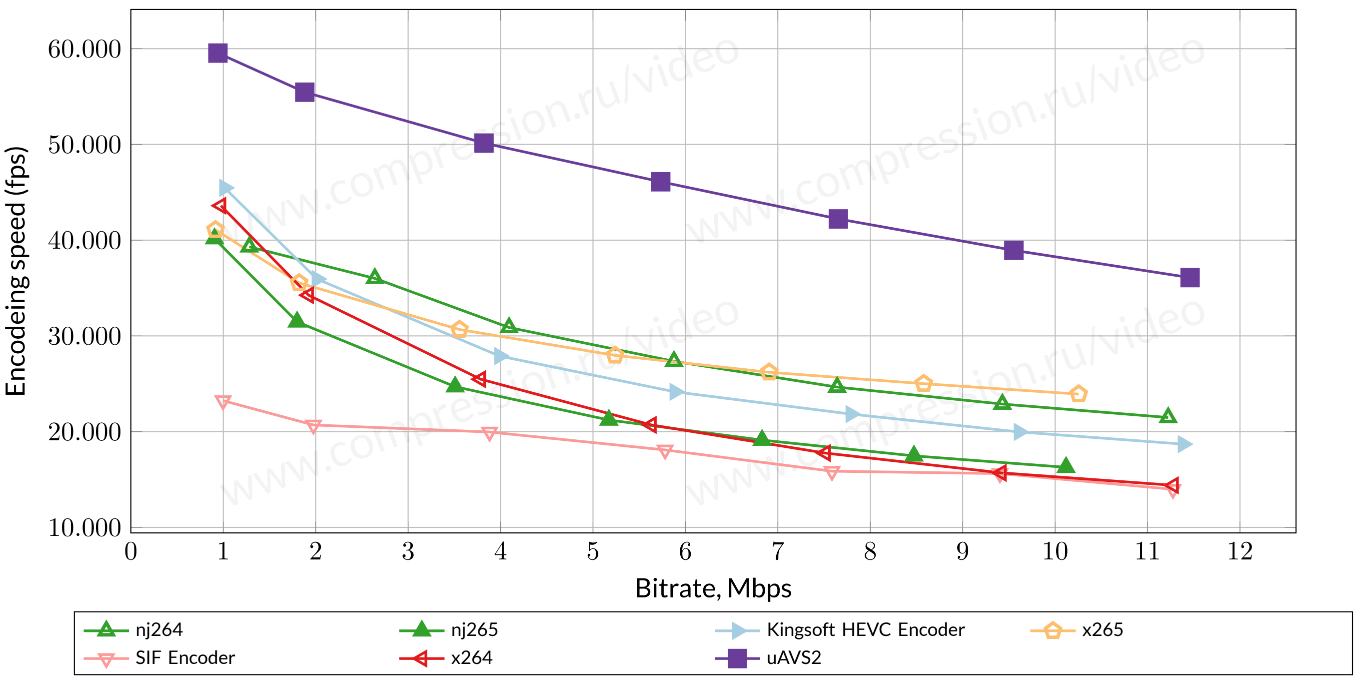


FIGURE 14: Encoding speed—use case "Universal Encoding," Steadicam sequence

5.3 Speed/Quality Trade-Off

Detailed descriptions of the speed/quality trade-off graphs can be found in Appendix E. Sometimes, codec results are not present in the particular graph owing to the codec's extremely poor performance (i.e. the codec's RD curve has no intersection with the reference's RD curve).
The speed/quality trade-off graphs show both relative quality and speed scores of encoders under comparison. Since x264 was chosen as reference codec in our comparison, we normalized all scores using x264 scores.

For universal encoding use case there are three Pareto optimal encoders in terms of mean speed and quality—AVS2, x265 and Kingsoft HEVC encoders. "Pareto optimal" encoder means there is no encoder faster and better than it in this test. Nevertheless, there are slight differences for particular sequences, for example, x265 is the best encoder (i.e. has the best quality and speed scores) for Christmas Cats sequence, Infinit sequence has 2 pareto optimal encoders: nj265 and x265.

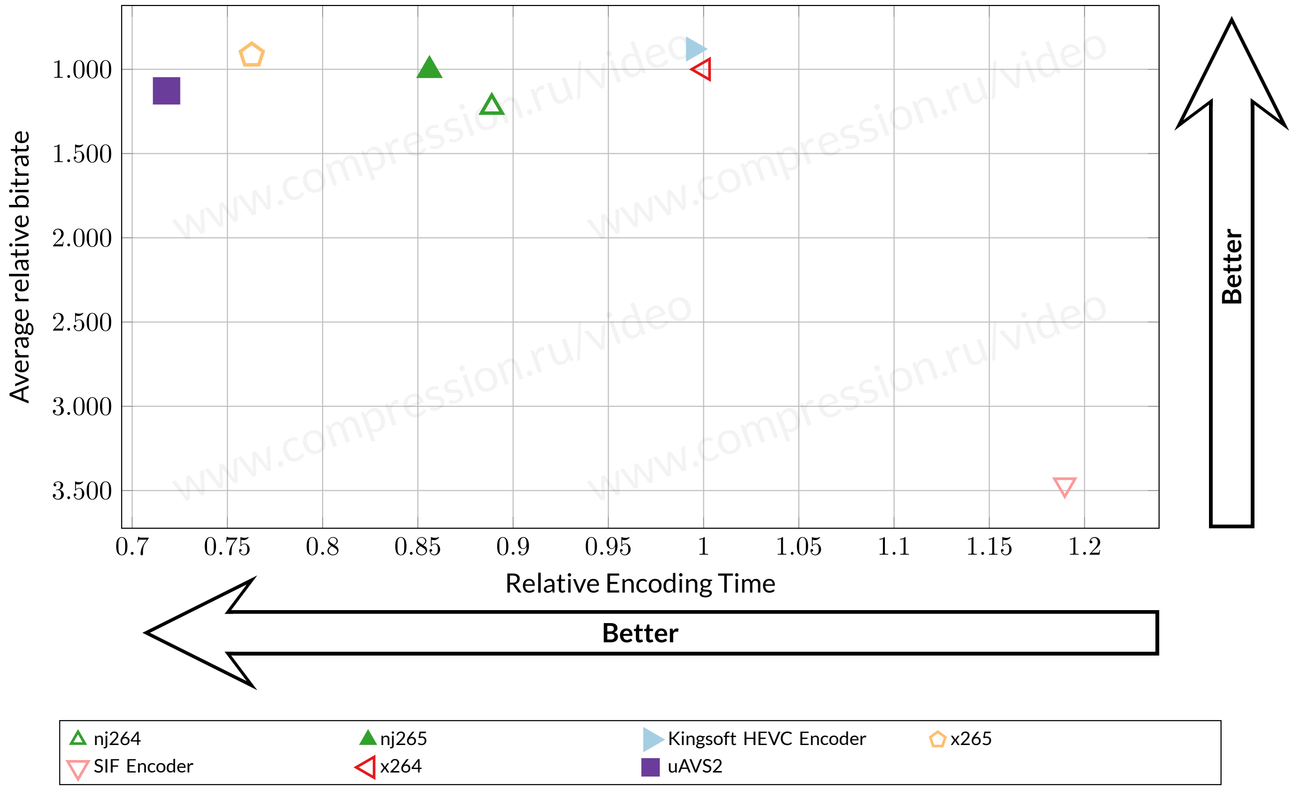


FIGURE 15: Speed/quality trade-off—use case "Universal Encoding," all sequences, YUV-SSIM metric

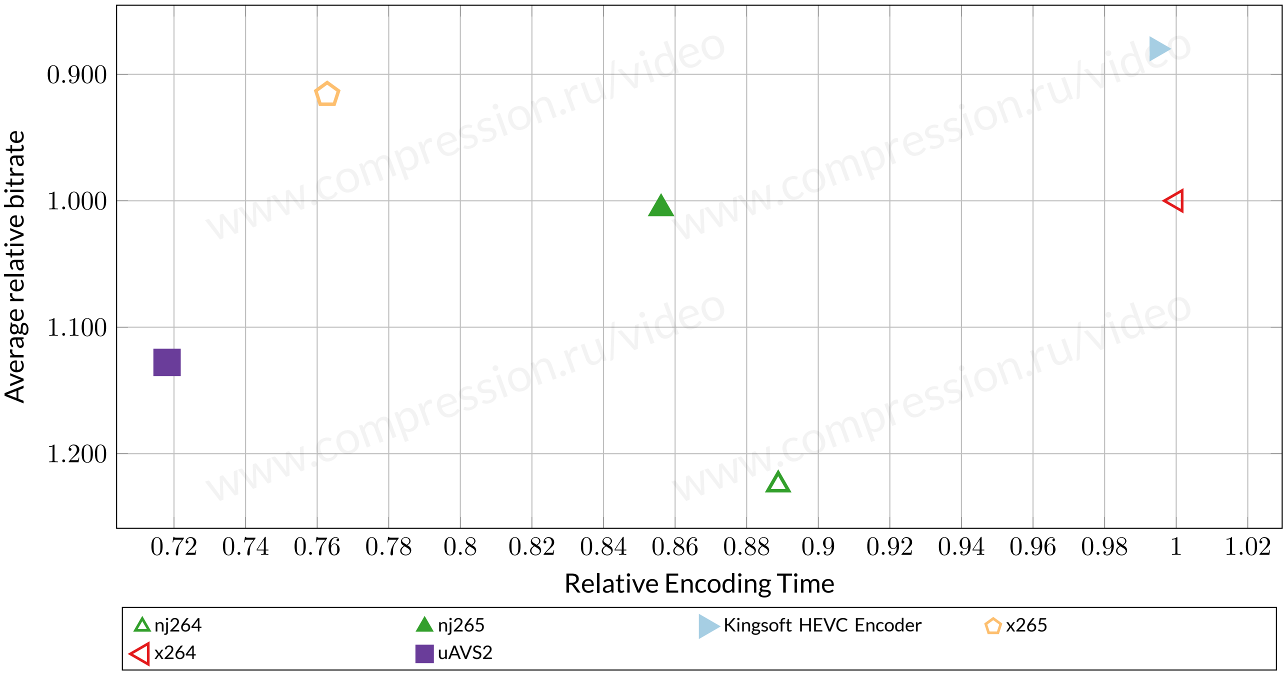


FIGURE 16: Speed/quality trade-off—use case "Universal Encoding," all sequences, YUV-SSIM metric, without SIF Encoder

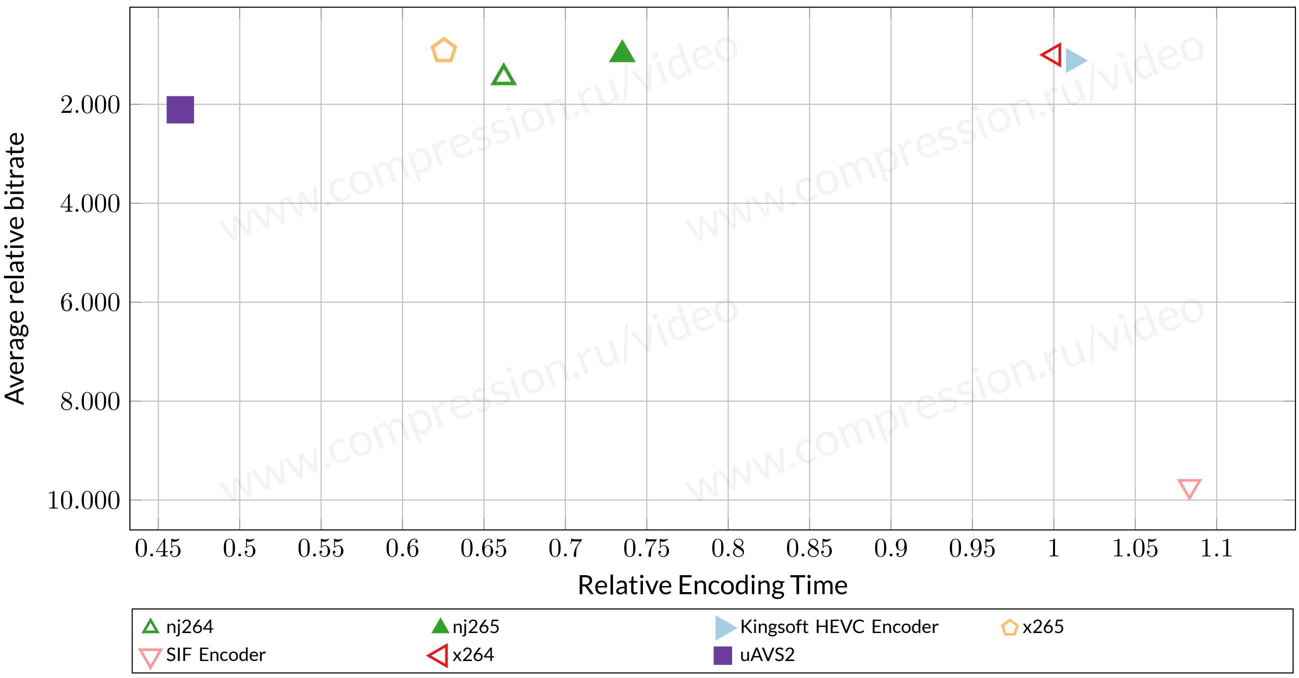


FIGURE 17: Speed/quality trade-off—use case "Universal Encoding," Color Tune sequence, YUV-SSIM metric

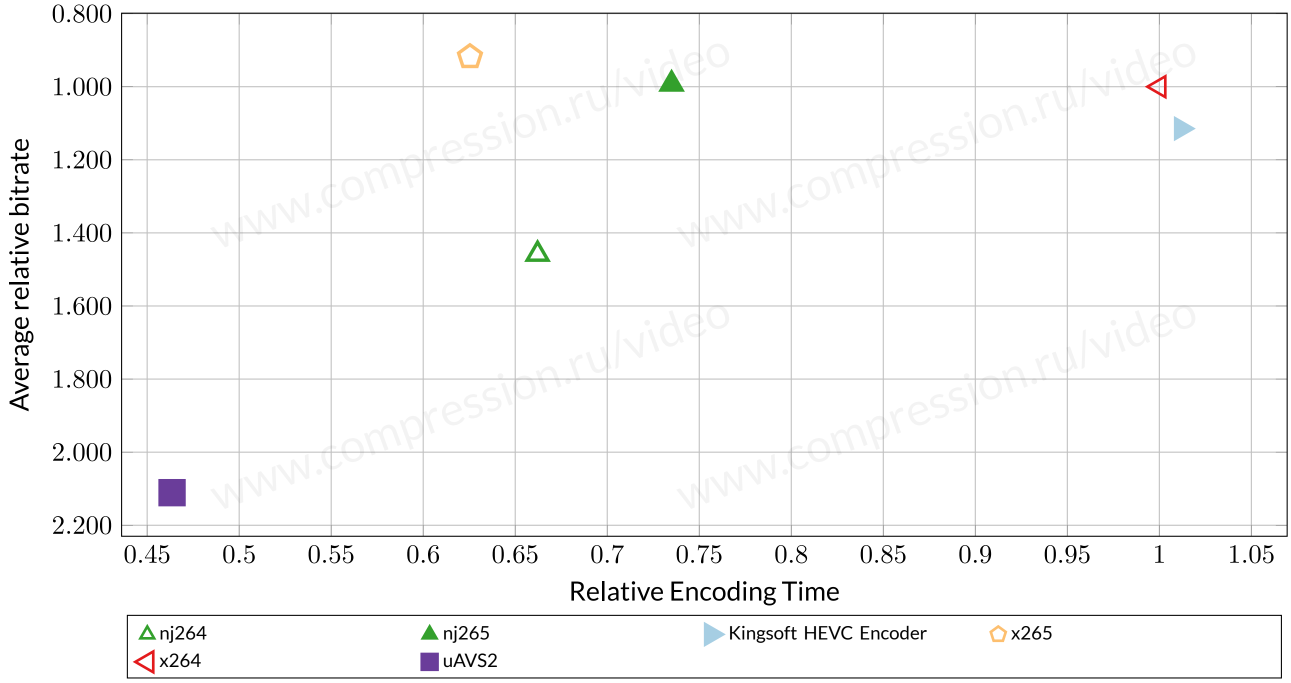


FIGURE 18: Speed/quality trade-off—use case "Universal Encoding," Color Tune sequence, YUV-SSIM metric, without SIF Encoder

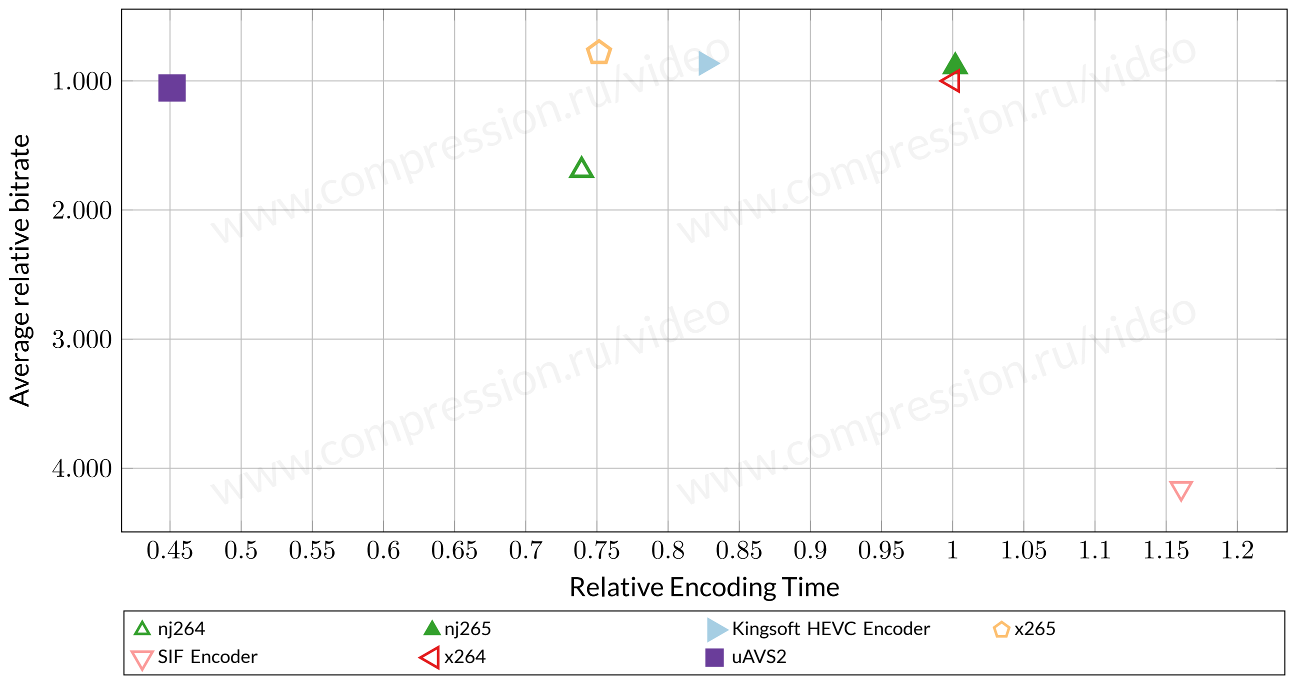


FIGURE 19: Speed/quality trade-off—use case "Universal Encoding," Steadicam sequence, YUV-SSIM metric

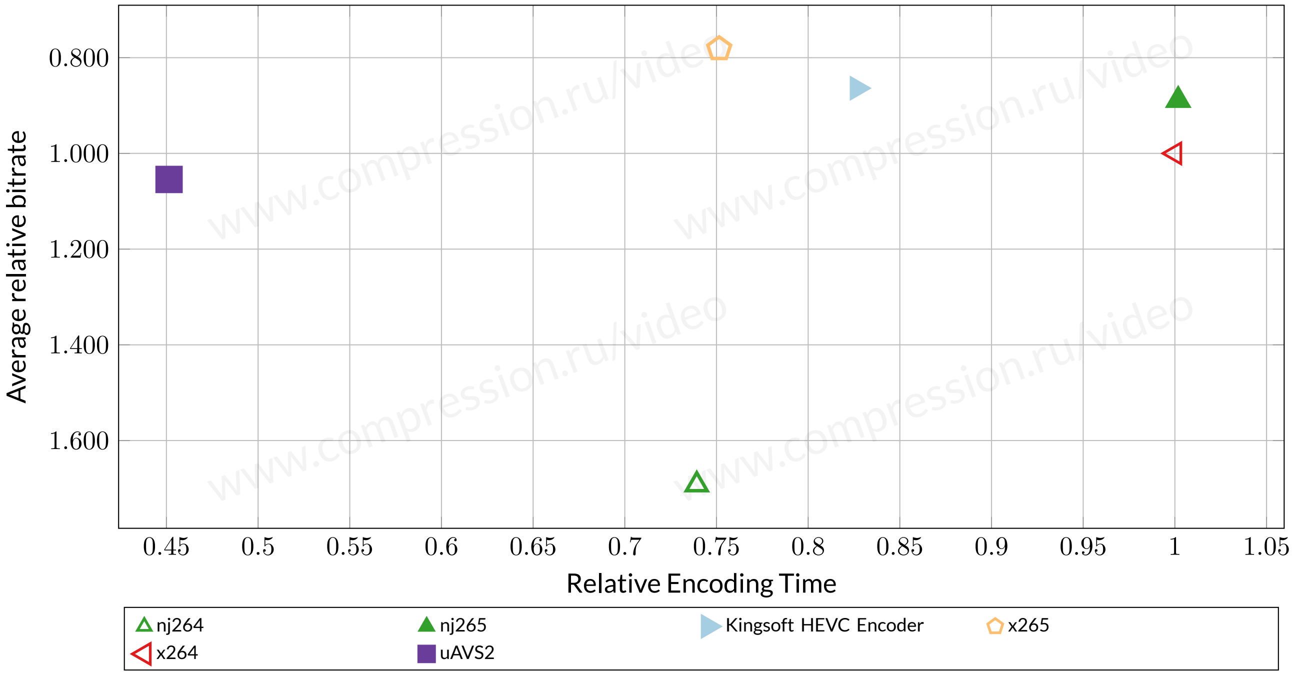


FIGURE 20: Speed/quality trade-off—use case "Universal Encoding," Steadicam sequence, YUV-SSIM metric, without SIF Encoder

5.4 Bitrate Handling

The plots below show how accurately encoded stream's real bitrate matches bitrate requested by user. Almost all encoders handle bitrate well, but there are issues for some encoders at some sequences, for example, it is challenging for most codecs to hit target bitrate for Color Tune sequence (e.g. AVS2 overshoots low target bitrates, x265 and nj265 undershoot all the target bitrates), Infinit sequence shows high bitrate fluctuation for most of the encoders etc.

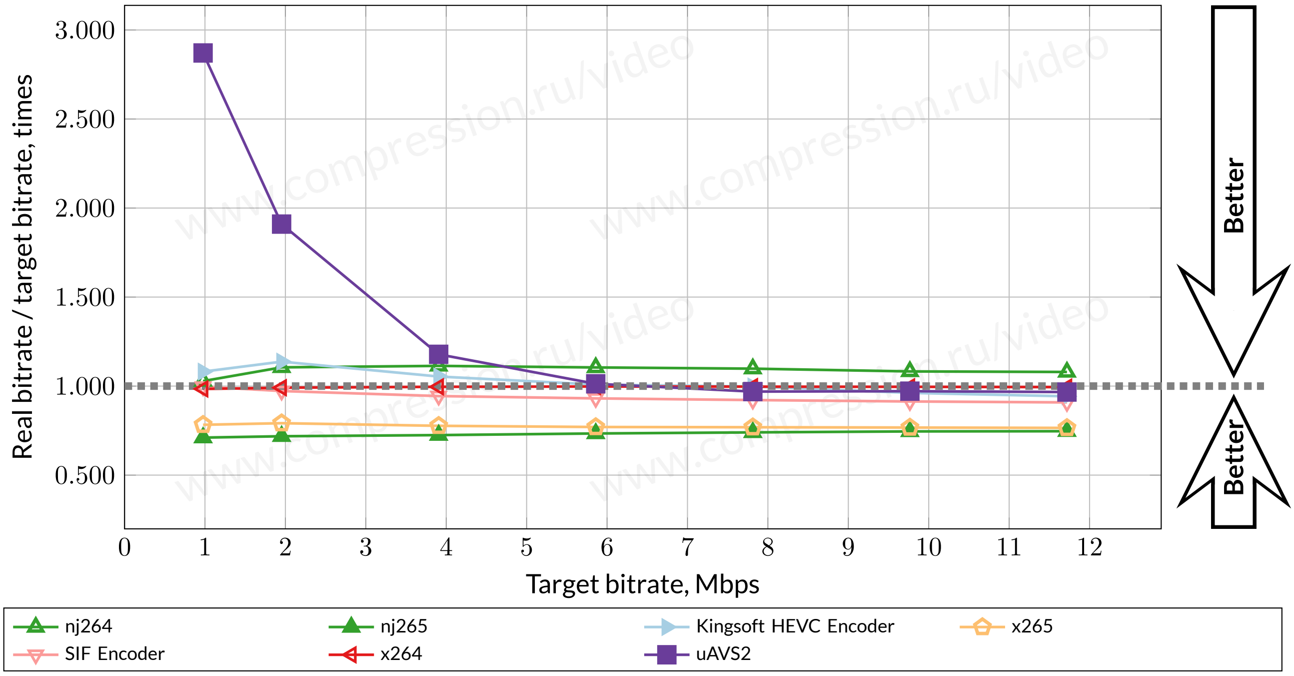


FIGURE 21: Bitrate handling—use case “Universal Encoding,” Color Tune sequence

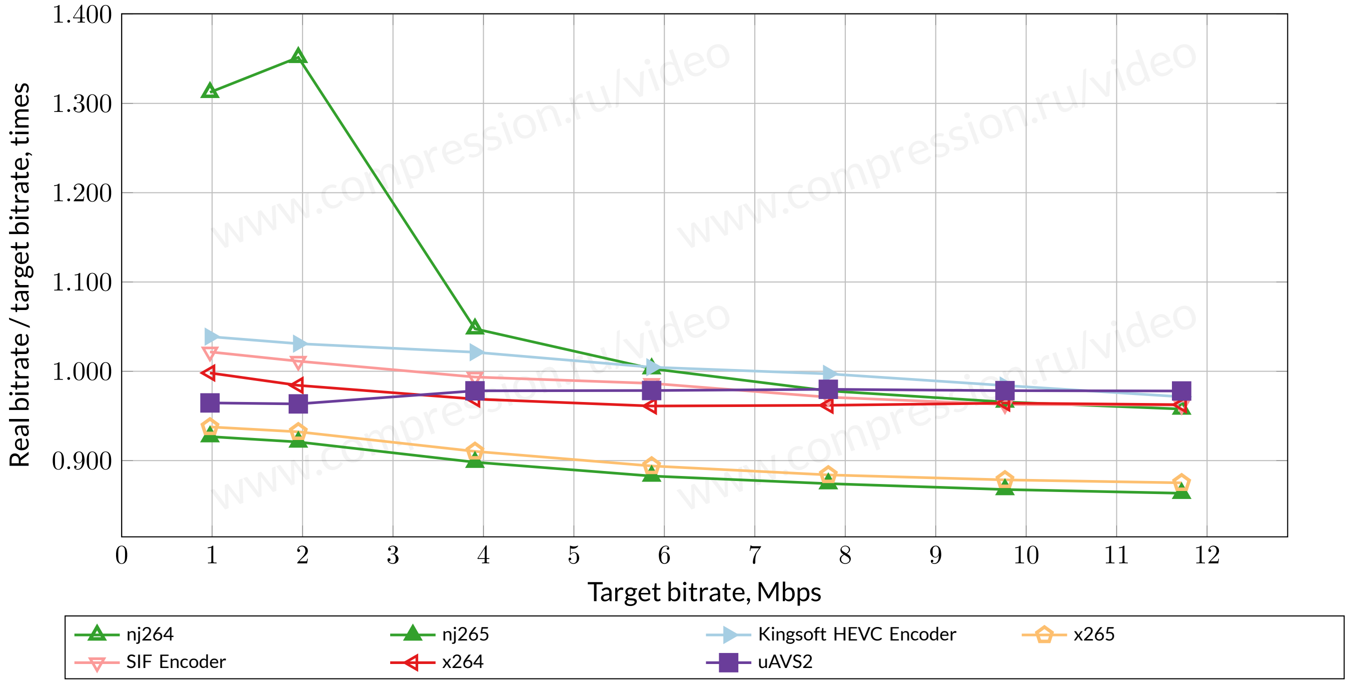


FIGURE 22: Bitrate handling—use case “Universal Encoding,” Steadicam sequence

5.5 Relative Quality Analysis

Note that each number in the tables below corresponds to some range of bitrates (see Appendix E). 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

	nj264	nj265	Kingsoft HEVC Encoder	x265	SIF Encoder	x264	uAVS2
nj264	100% ☺	83% ☺	71% ☺	76% ☺	333% 😐	82% ☺	96% ☺
nj265	125% ☺	100% ☺	87% ☺	91% ☺	445% 😐	99% ☺	119% ☺
Kingsoft HEVC Encoder	144% ☺	118% ☺	100% ☺	106% ☺	N/A ☹	114% ☺	137% ☺
x265	139% ☺	113% ☺	97% ☺	100% ☺	N/A ☹	109% ☺	130% ☺
SIF Encoder	38% ☺	28% 😐	N/A ☹	N/A ☹	100% ☺	29% 😐	30% 😐
x264	125% ☺	102% ☺	89% ☺	93% ☺	418% 😐	100% ☺	119% ☺
uAVS2	111% ☺	91% ☺	77% ☺	81% ☺	392% 😐	89% ☺	100% ☺

Confidence ☹ 0% 😐 50% ☺ 100%

TABLE 4: Average bitrate ratio for a fixed quality—use case "Universal Encoding," YUV-SSIM metric

Figure below depicts the data from the table above. Each line in the figure corresponds to one codec. Values on the vertical axis are the average relative bitrates compared with the codecs along the horizontal axis. A lower bitrate indicates better relative results.

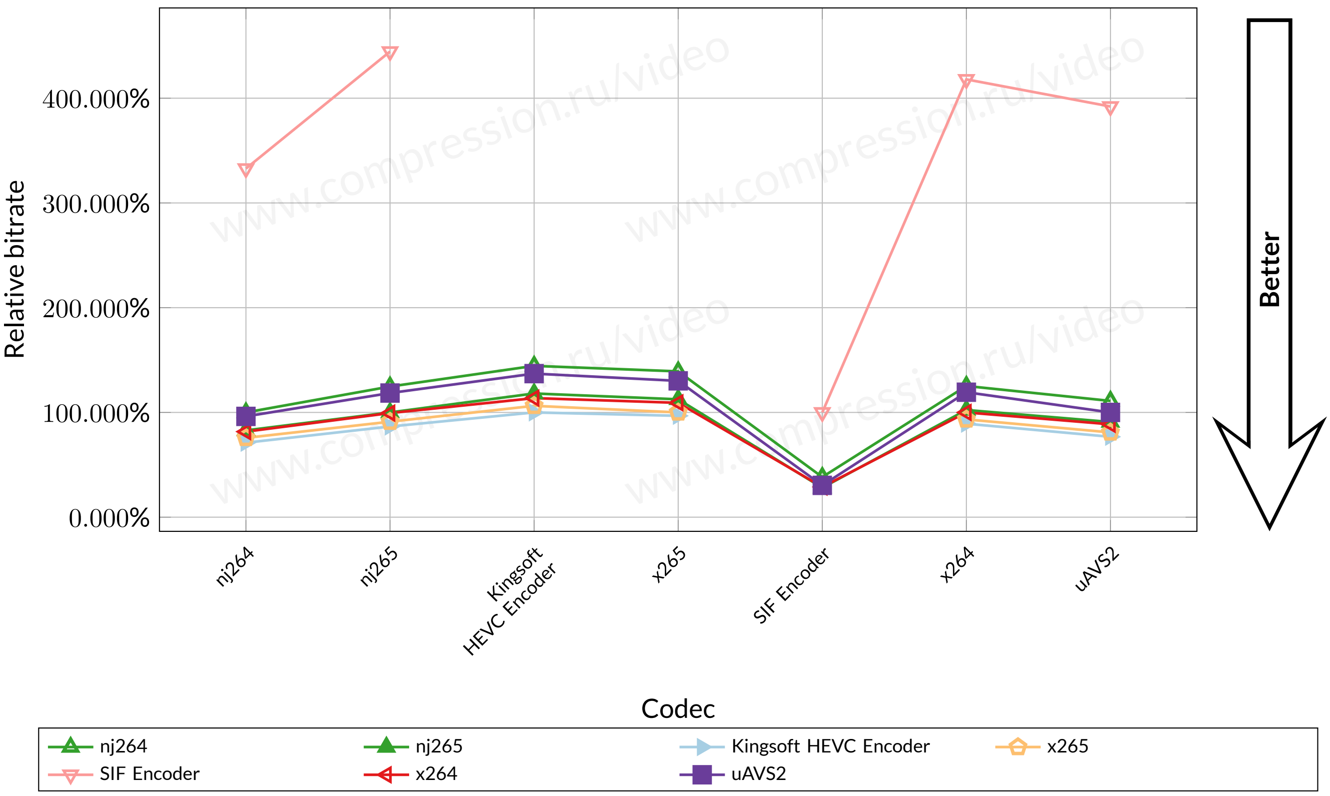


FIGURE 23: Average bitrate ratio for a fixed quality—use case "Universal Encoding," YUV-SSIM metric

6 RIPPING ENCODING

6.1 RD curves

Similarly to Fast and Universal encoding use cases, we can't name the absolute leader. Nevertheless, x265 takes the first place according to mean quality score. For particular sequences, however, the first place is taken by other encoders: for example, nj265 has the best SSIM scores for low bitrates at Christmas Cats sequence and for all bitrates at Crowd Run Sequence.

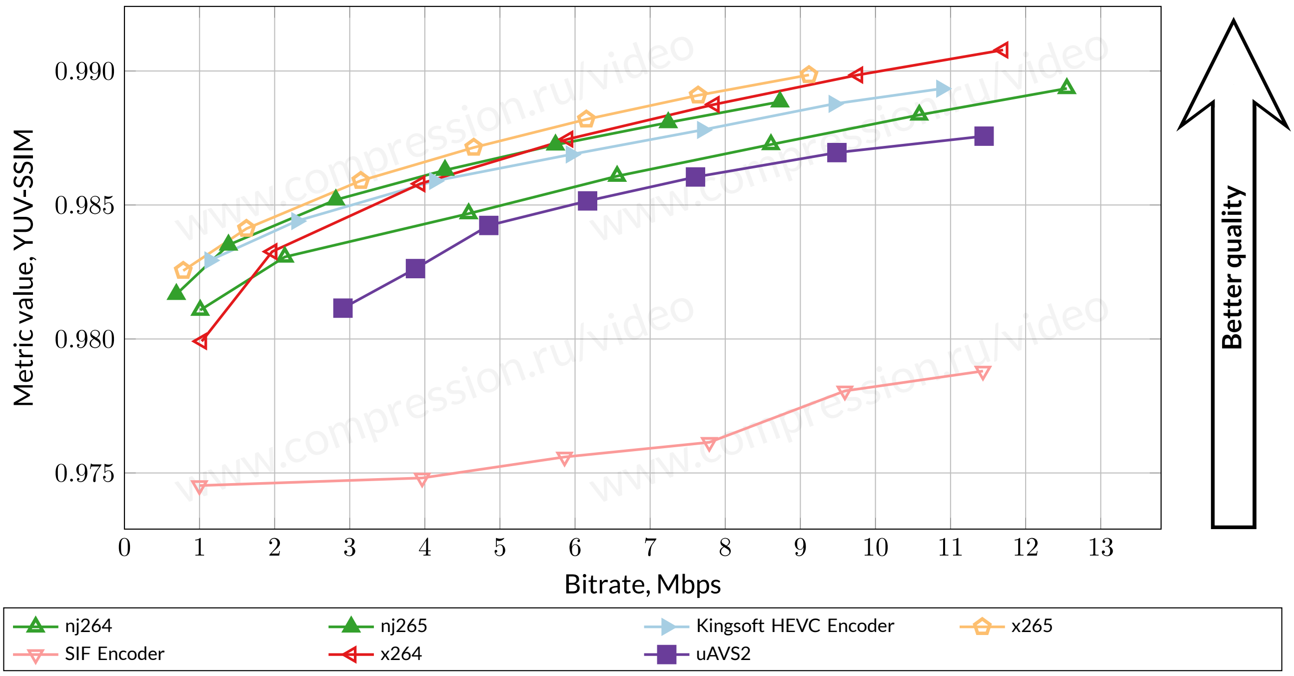


FIGURE 24: Bitrate/quality—use case "Ripping Encoding," Color Tune sequence, YUV-SSIM metric

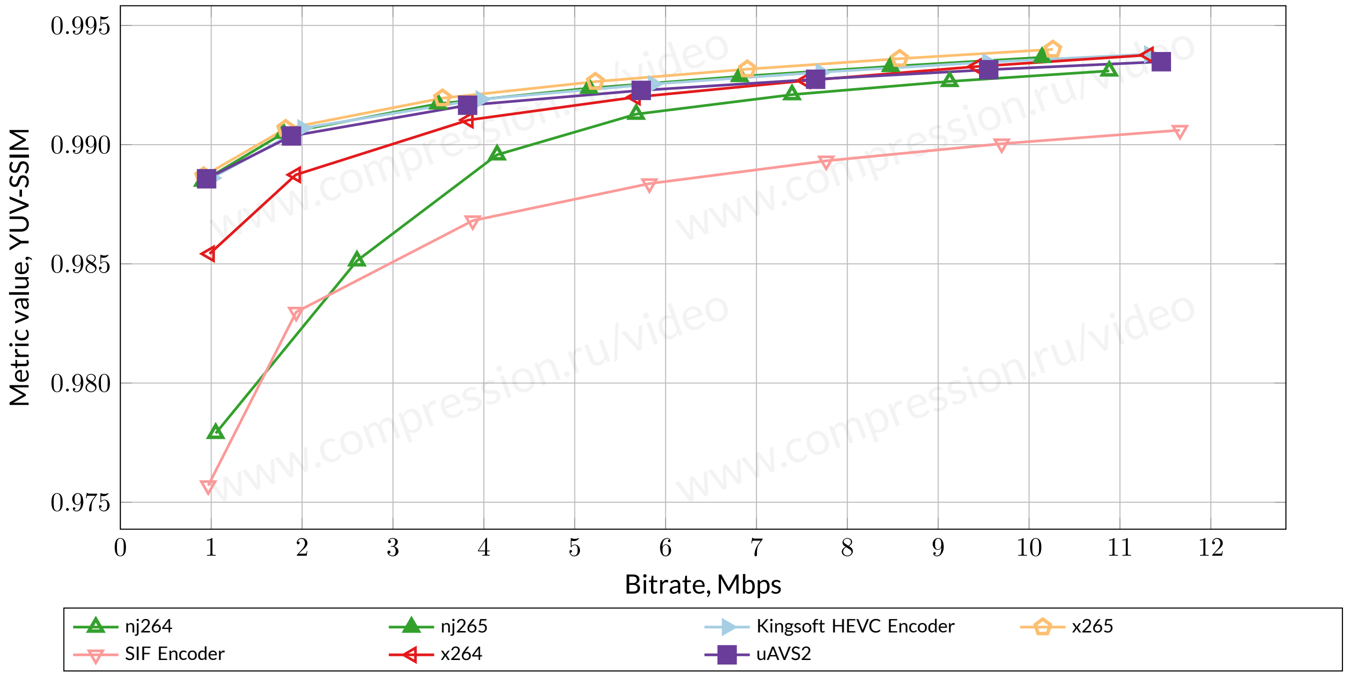


FIGURE 25: Bitrate/quality—use case "Ripping Encoding," Steadicam sequence, YUV-SSIM metric

6.2 Encoding Speed

Figures below show difference in encoding speed among participating codecs. SIF encoder is the fastest option for all sequences. According to mean speed scores, the second place goes to AVS2.

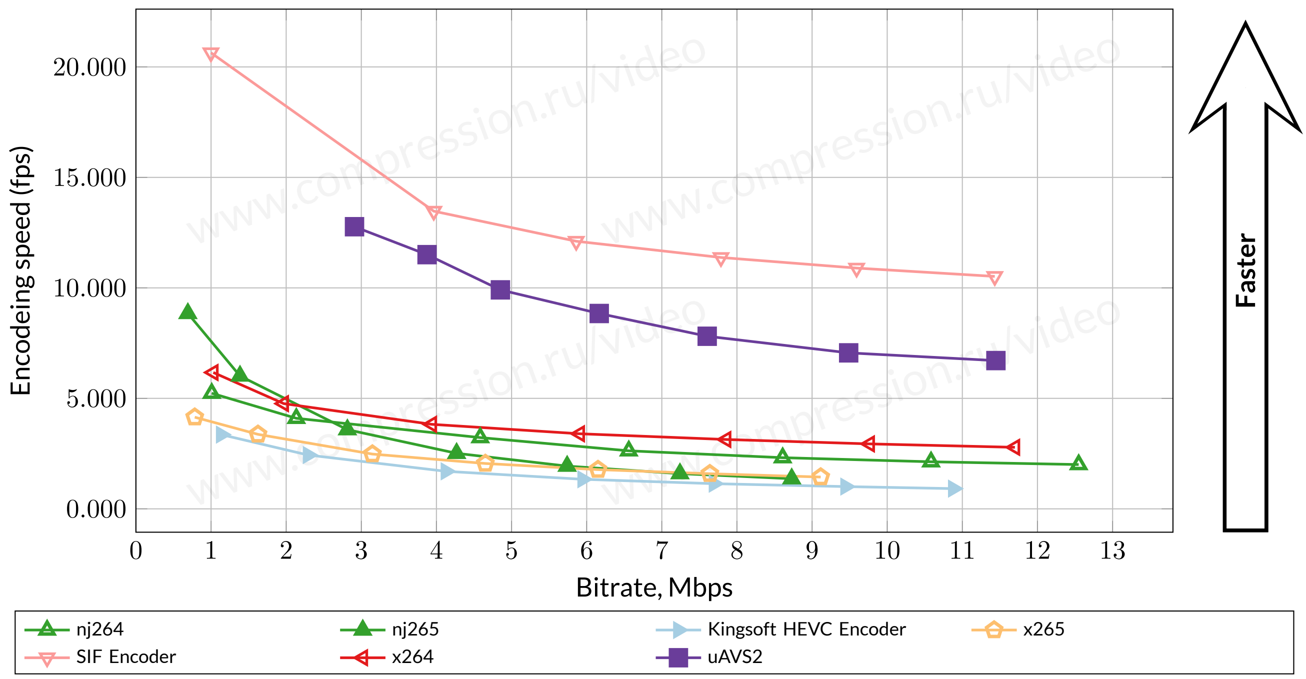


FIGURE 26: Encoding speed—use case "Ripping Encoding," Color Tune sequence

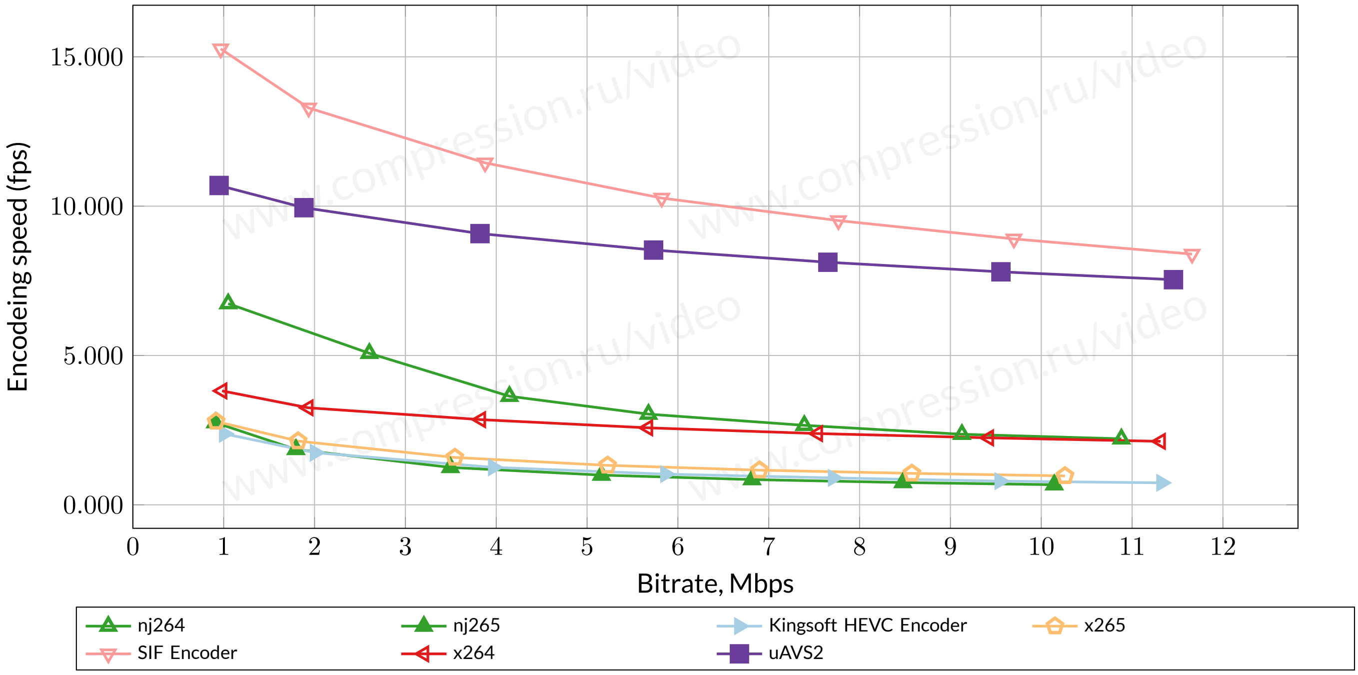


FIGURE 27: Encoding speed—use case "Ripping Encoding," Steadicam sequence

6.3 Speed/Quality Trade-Off

Detailed descriptions of the speed/quality trade-off graphs can be found in Appendix E. Sometimes, codec results are not present in the particular graph owing to the codec's extremely poor performance (i.e. the codec's RD curve has no intersection with the reference's RD curve).
The speed/quality trade-off graphs show both relative quality and speed scores of encoders under comparison. Since x264 was chosen as reference codec in our comparison, we normalized all scores using x264 scores.

For Ripping encoding use case there are three Pareto optimal encoders in terms of mean speed and quality—SIF, AVS2 and x265. "Pareto optimal" encoder means there is no encoder faster and better than it in this test. Nevertheless, there are slight differences for particular sequences, for example, x264 is among the Pareto optimal codecs at Color Tune sequence.

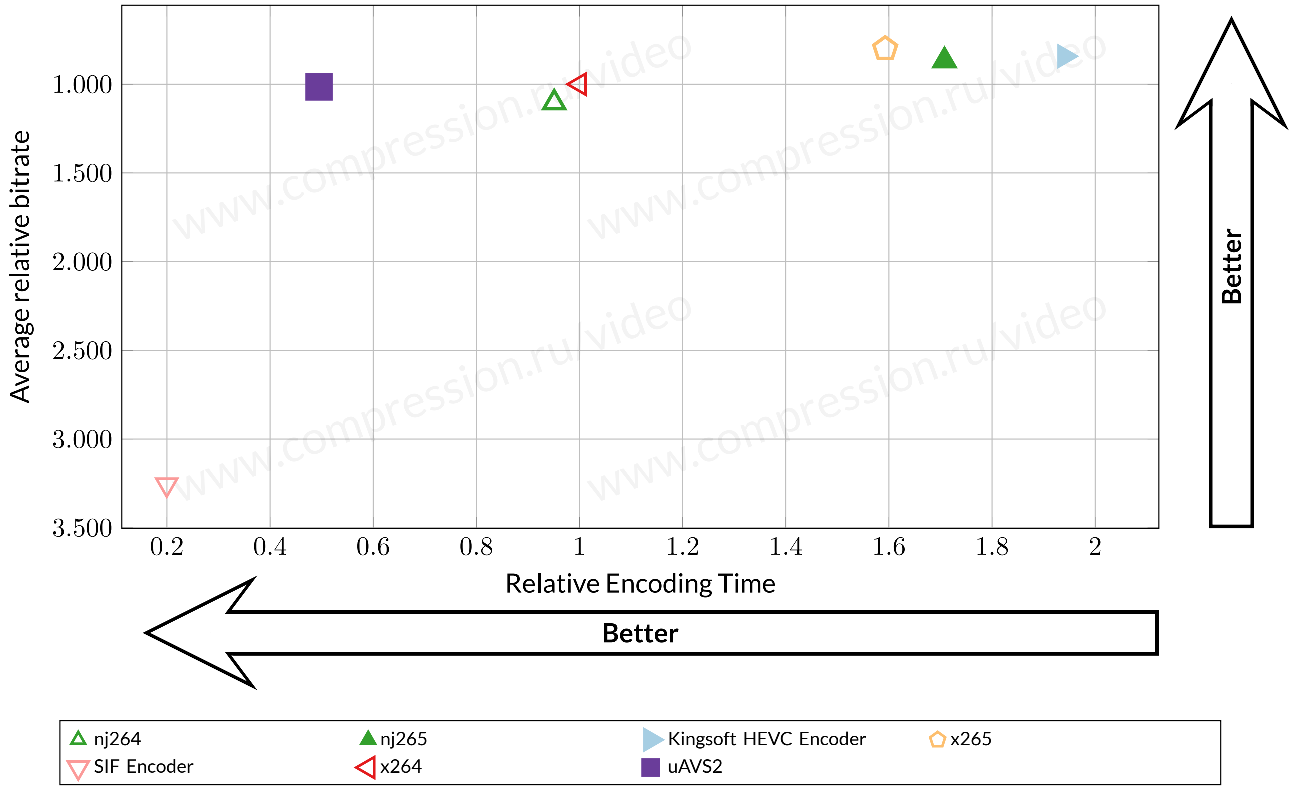


FIGURE 28: Speed/quality trade-off—use case "Ripping Encoding," all sequences, YUV-SSIM metric

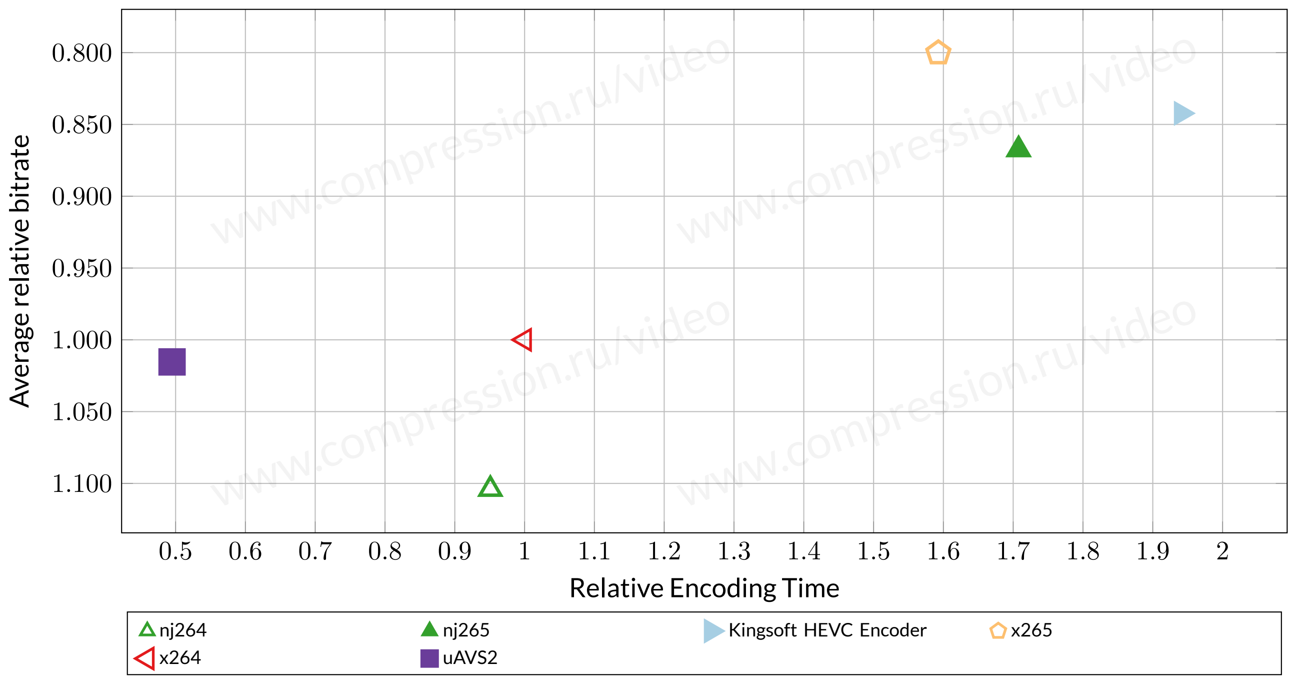


FIGURE 29: Speed/quality trade-off—use case "Ripping Encoding," all sequences, YUV-SSIM metric, without SIF Encoder

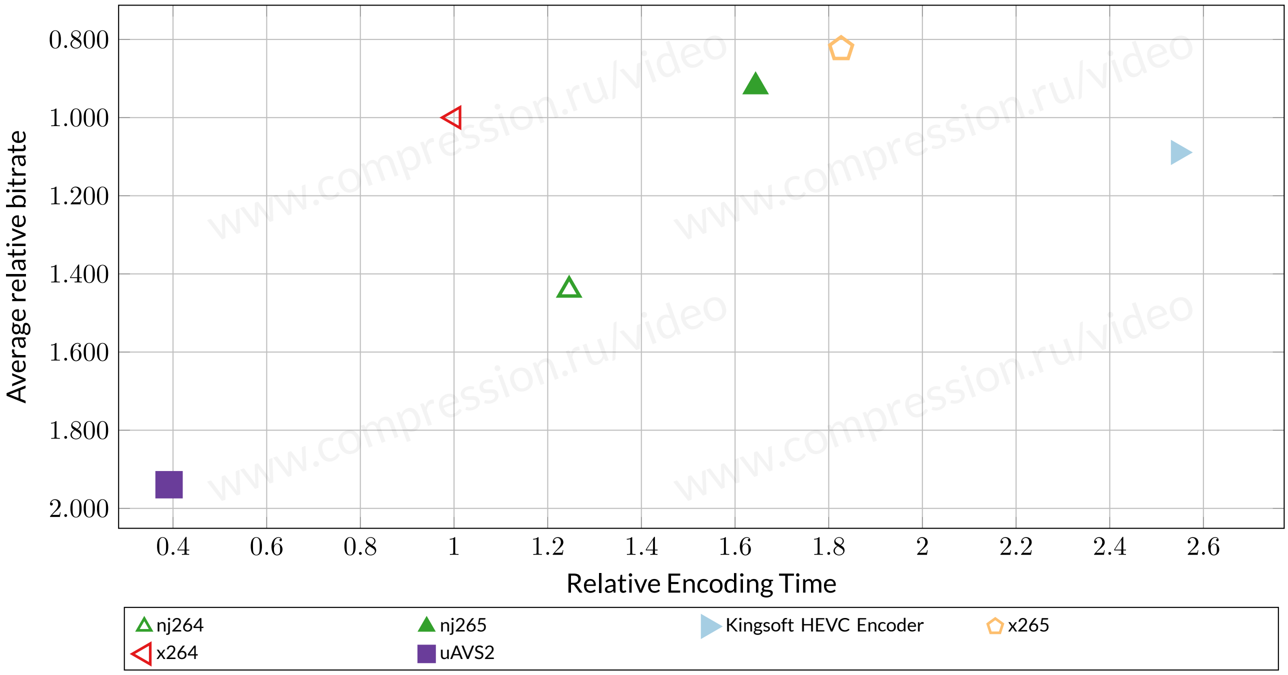


FIGURE 30: Speed/quality trade-off—use case “Ripping Encoding,” Color Tune sequence, YUV-SSIM metric

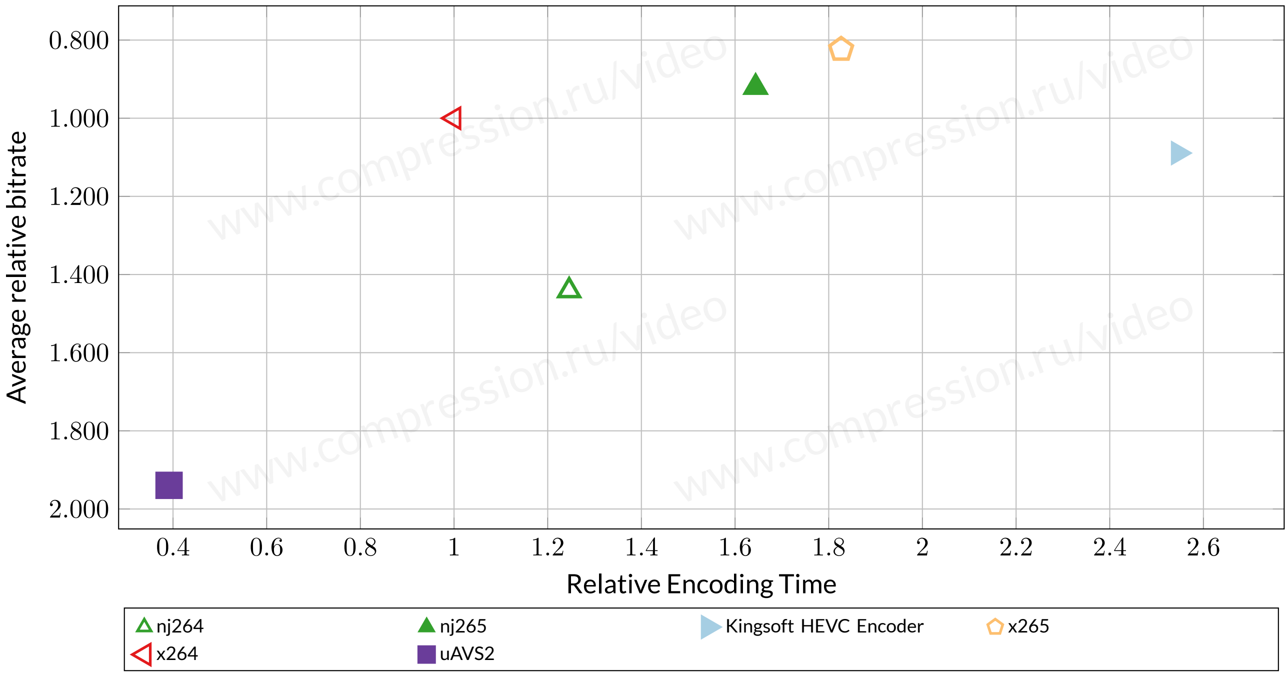


FIGURE 31: Speed/quality trade-off—use case “Ripping Encoding,” Color Tune sequence, YUV-SSIM metric, without SIF Encoder

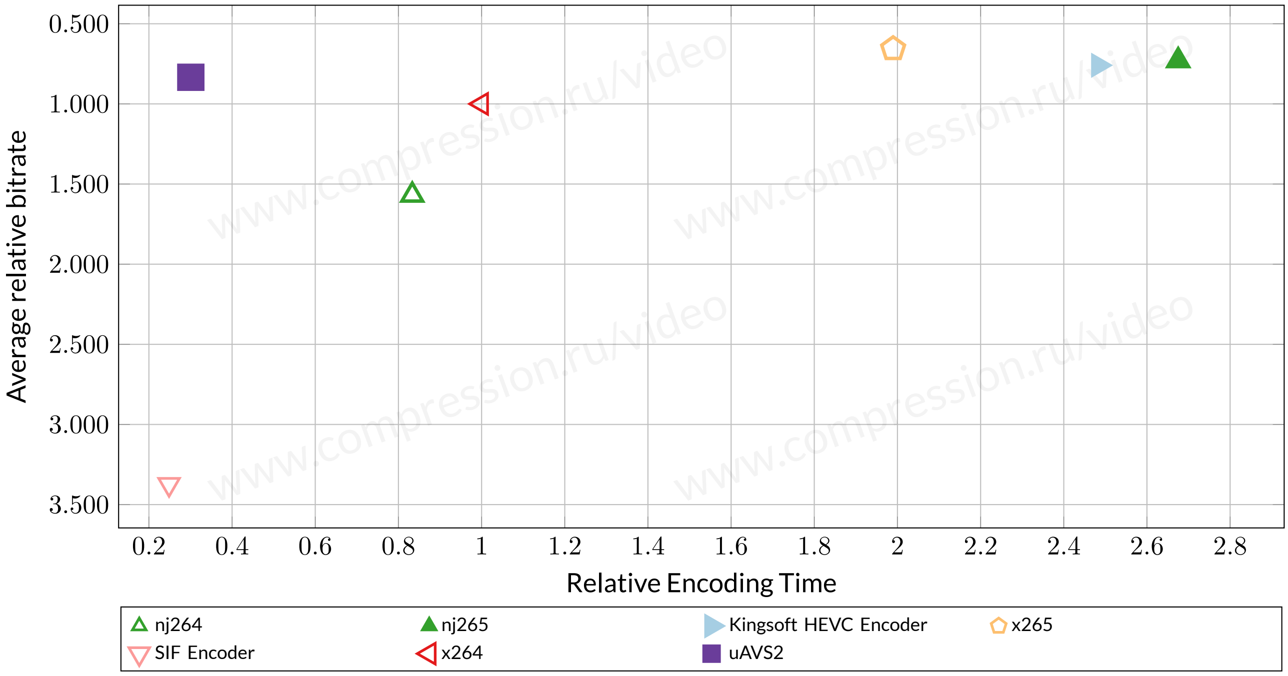


FIGURE 32: Speed/quality trade-off—use case "Ripping Encoding," Steadicam sequence, YUV-SSIM metric

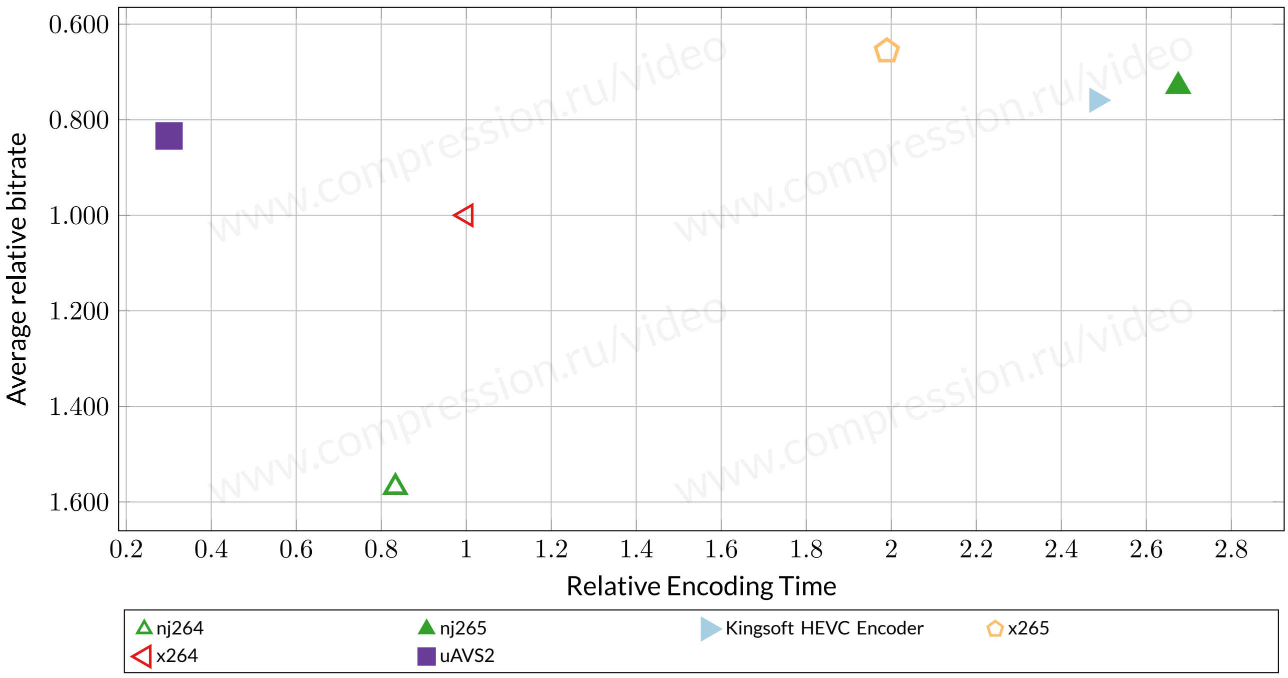


FIGURE 33: Speed/quality trade-off—use case "Ripping Encoding," Steadicam sequence, YUV-SSIM metric, without SIF Encoder

6.4 Bitrate Handling

The plots below show how accurately encoded stream's real bitrate matches bitrate requested by user. Almost all encoders handle bitrate well, but there are issues for some encoders at some sequences, for example, AVS2 overshoots low target bitrates for Color Tune and Crowd Run sequences, most of encoders undershoot low target bitrates at Coffee Beans sequence etc.

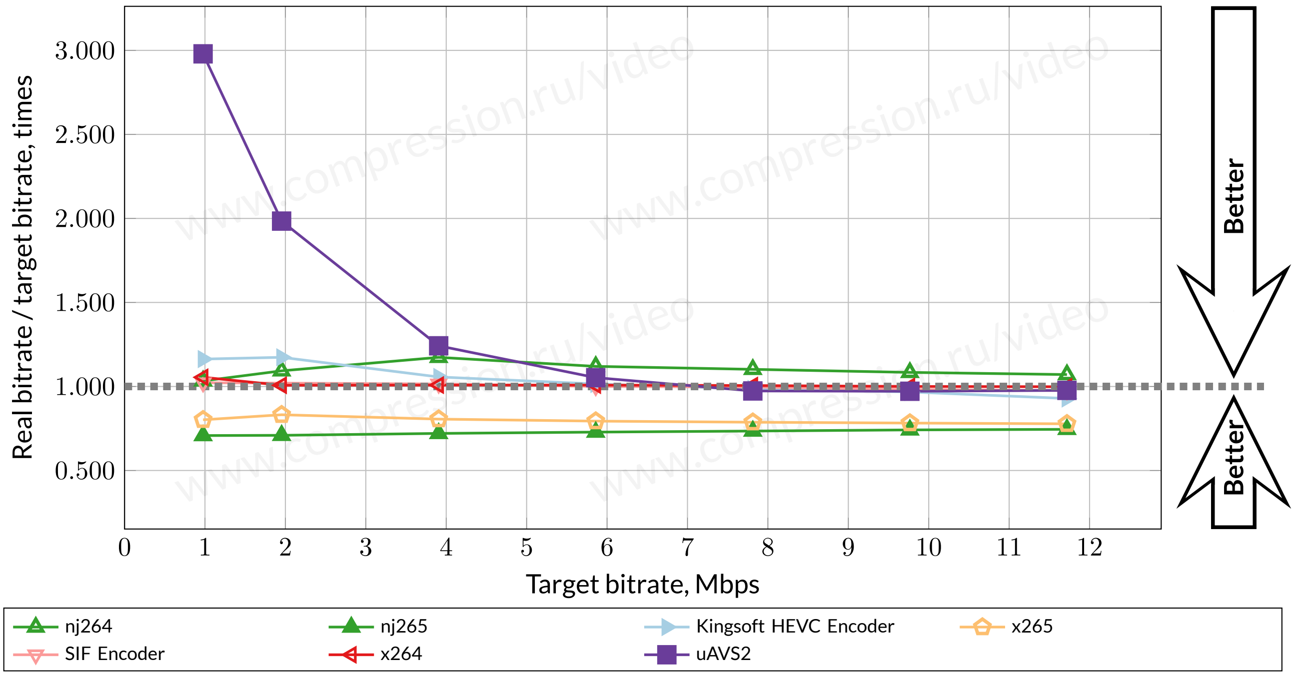


FIGURE 34: Bitrate handling—use case "Ripping Encoding," Color Tune sequence

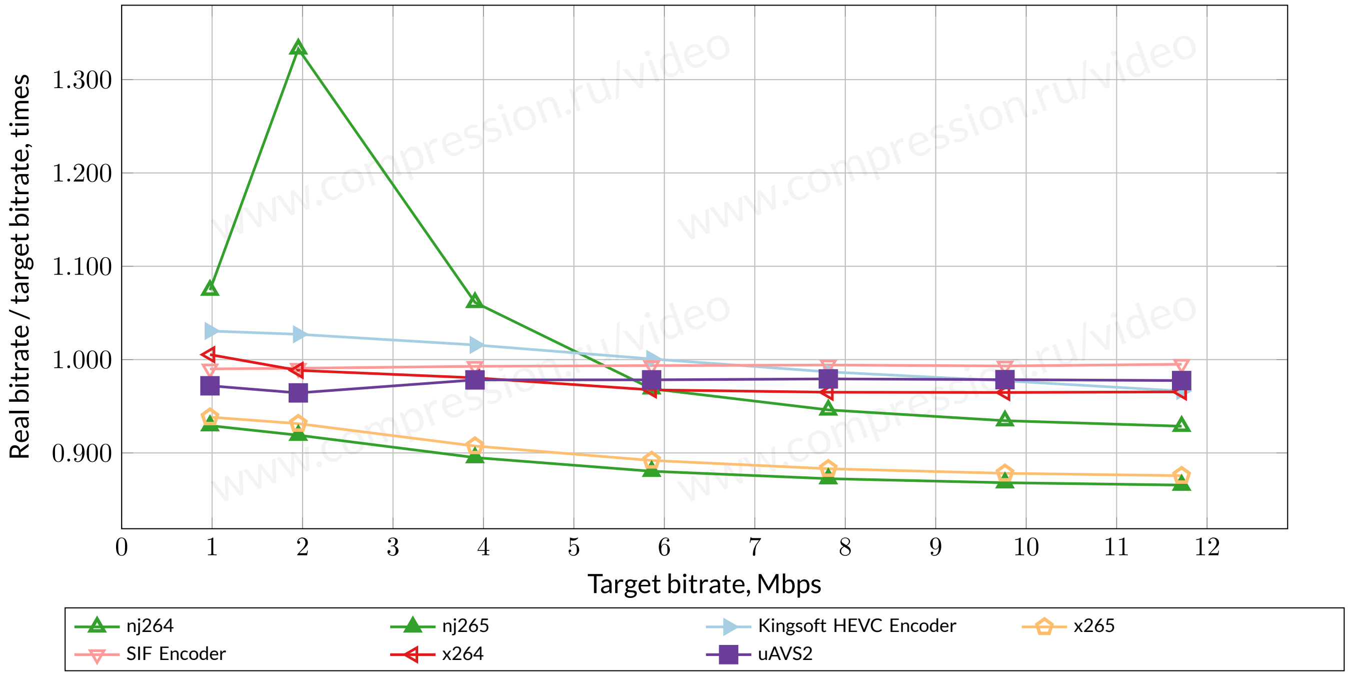


FIGURE 35: Bitrate handling—use case “Ripping Encoding,” Steadicam sequence

6.5 Relative Quality Analysis

Note that each number in the tables below corresponds to some range of bitrates (see Appendix E). 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

	nj264	nj265	Kingsoft HEVC Encoder	x265	SIF Encoder	x264	uAVS2
nj264	100% ☺	79% ☺	75% ☺	73% ☺	N/A ☹	91% ☺	96% ☺
nj265	131% ☺	100% ☺	97% ☺	92% ☺	N/A ☹	115% ☺	128% ☺
Kingsoft HEVC Encoder	137% ☺	105% ☺	100% ☺	95% ☺	N/A ☹	119% ☺	130% ☺
x265	144% ☺	110% ☺	106% ☺	100% ☺	N/A ☹	125% ☺	137% ☺
SIF Encoder	36% ☺	24% 😐	N/A ☹	N/A ☹	100% ☺	31% 😐	N/A ☹
x264	113% ☺	88% ☺	85% ☺	81% ☺	N/A ☹	100% ☺	110% ☺
uAVS2	112% ☺	87% ☺	82% ☺	78% ☺	N/A ☹	98% ☺	100% ☺

Confidence ☹ 😐 ☺

0% 50% 100%

TABLE 5: Average bitrate ratio for a fixed quality—use case “Ripping Encoding,” YUV-SSIM metric

Figure below depicts the data from the table above. Each line in the figure corresponds to one codec. Values on the vertical axis are the average relative bitrates compared with the codecs along the horizontal axis. A lower bitrate indicates better relative results.

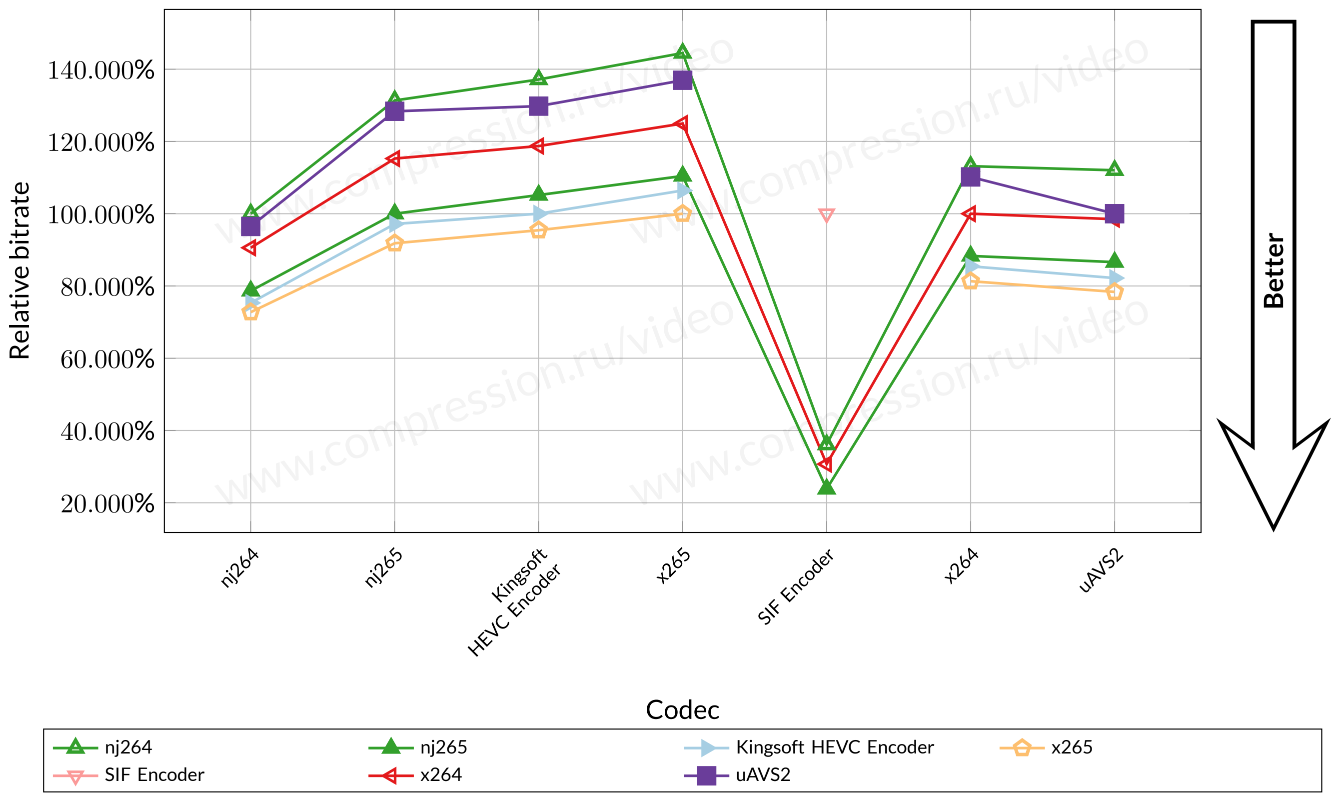


FIGURE 36: Average bitrate ratio for a fixed quality—use case "Ripping Encoding," YUV-SSIM metric

7 CONCLUSION

According to quality scores, the codecs can be ordered in the following way:

- Kingsoft HEVC encoder is in the first place
- x265 is in the second place
- nj265 is in the third place.

Below we show the plot illustrating speed/quality relation of all presets used in our comparison. x264 with "veryslow" preset was chosen as the reference point. Each line on the plot corresponds to encoder and each point on the line corresponds to preset. Along x-axis we put mean speed of encoder's preset on our test dataset. Position along y-axis is determined by preset's bitrate relative to reference (i.e. how much or less bits encoder needs to gain same quality as reference). Detailed description of relative bitrate computation can be found in Appendix E.4.

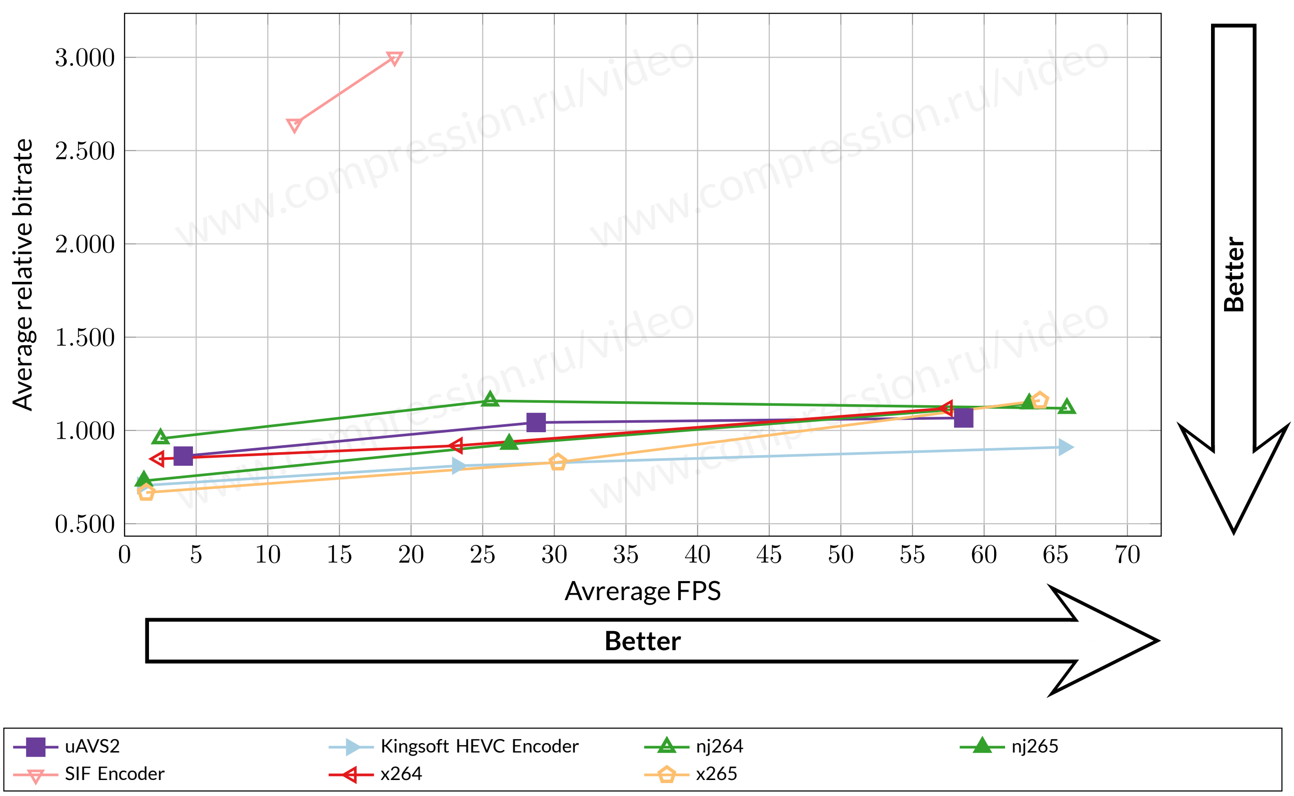


FIGURE 37: Average bitrate ratio and speed for various presets—YUV-SSIM metric.

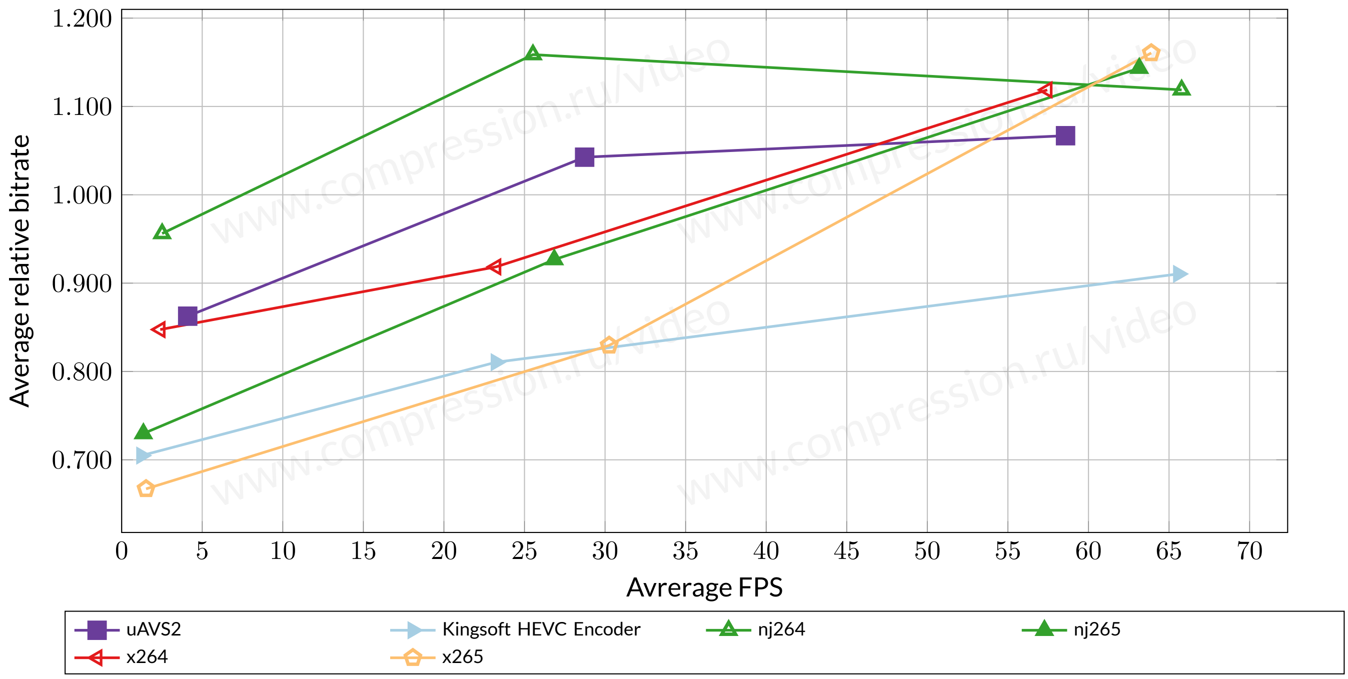


FIGURE 38: Average bitrate ratio and speed for various presets—YUV-SSIM metric, without SIF Encoder.

7.1 Fast Encoding

KingSoft HEVC encoder significantly outperforms all its competitors according to quality scores computed for Fast encoding use case. The rest of codecs gained quality scores in a very narrow range (95-105%).

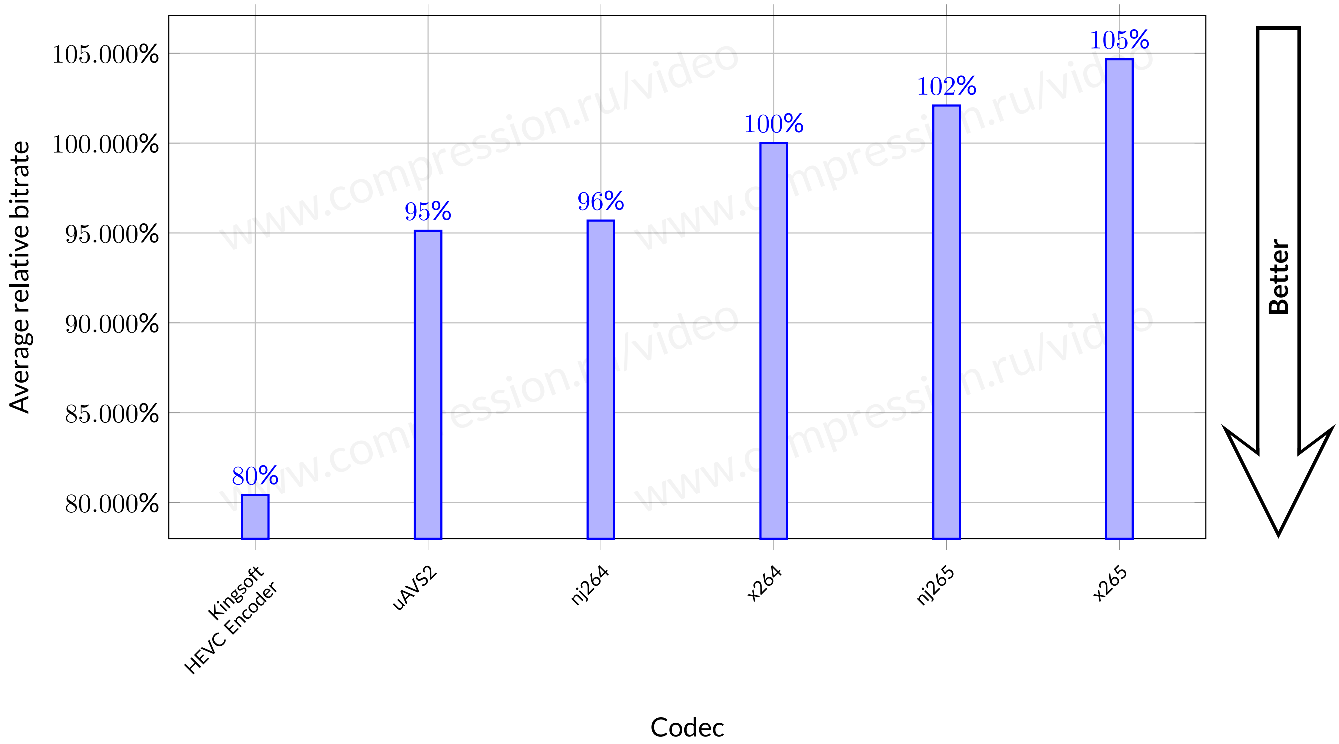


FIGURE 39: Average bitrate ratio for a fixed quality—use case "Fast Encoding," YUV-SSIM metric.

7.2 Universal Encoding

The competition among codecs was harder for Universal use case than for Fast encoding use case: KingSoft HEVC encoder's quality gain is not that significant in comparison with x265 (which is in the second place). The overall range of quality scores is wider as well.

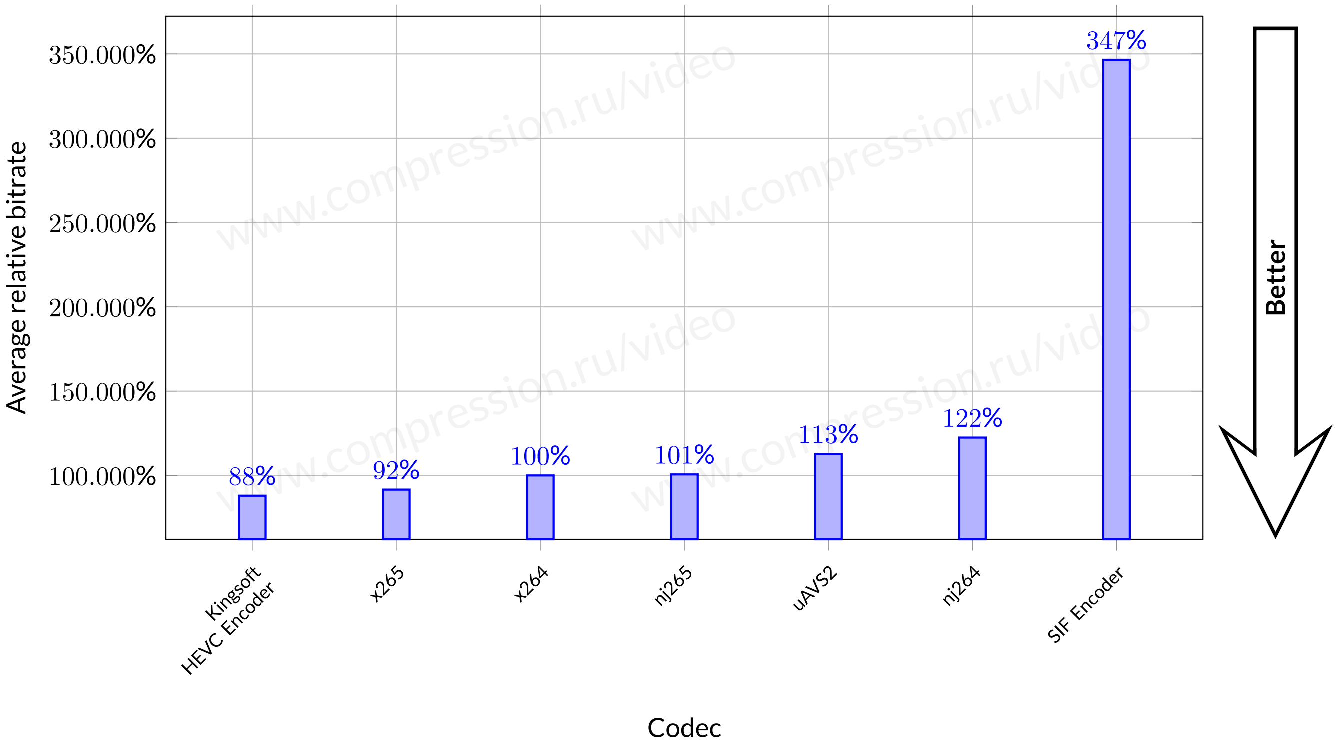


FIGURE 40: Average bitrate ratio for a fixed quality—use case "Universal Encoding," YUV-SSIM metric.

7.3 Ripping Encoding

KingSoft HEVC encoder—the codec with the best quality scores at Fast and Universal use cases—takes the second place at Ripping use case. The first place is taken by x265, the third—by nj265. Notably, at this use case most of HEVC encoders in our comparison gained higher quality scores than H.264 encoders (x264, nj264).

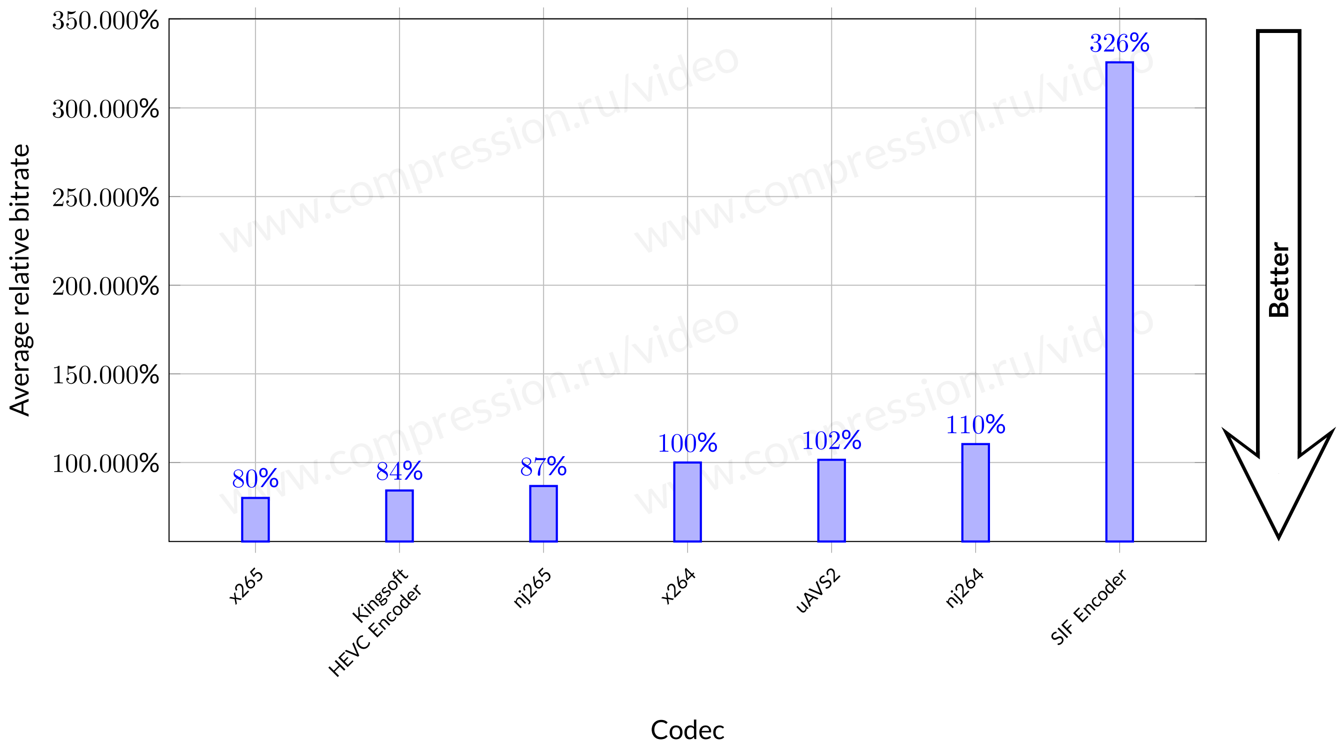


FIGURE 41: Average bitrate ratio for a fixed quality—use case "Ripping Encoding," YUV-SSIM metric.

7.4 Overall

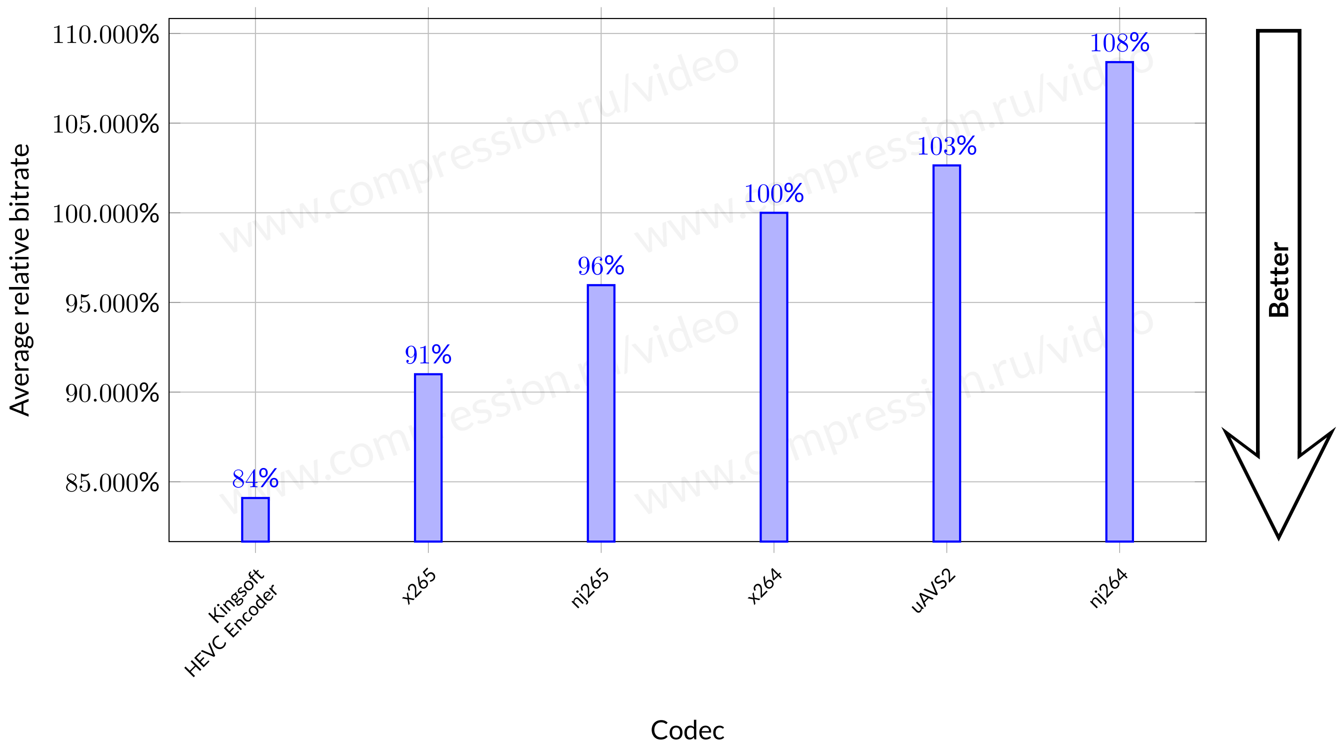


FIGURE 42: Average bitrate ratio for a fixed quality—YUV-SSIM metric.

A PARTICIPANTS' COMMENTS

A.1 Kingsoft

Thanks a lot for MSU's support of this comparison. After a comparison with X265's parameters, it can be figured that a large promotion happens at "-tune ssim" with aqmode=2 compared to "-tune psnr", whereas Kingsoft's encoder does not apply adaptive quantization in this comparision. In next comparison, Kingsoft's encoder will employ the adaptive quantization. What's more, Kingsoft's encoder achieves the best bitrate handling, which will produce some BDRate loss compared to x265.

B SEQUENCES

B.1 "Animation"

Sequence title	Animation
Resolution	1920×1080
Number of frames	833
Color space	YV12
Frames per second	24
Source	https://vimeo.com/173789876#t=0
Source resolution	FullHD
Bitrate	131.76 Mbps

The video illustrates steps of computer graphics creation process.

FIGURE 43: Animation sequence, frame 216

B.2 "Apple Tree"

Sequence title	Apple Tree
Resolution	1920×1080
Number of frames	338
Color space	YV12
Frames per second	30
Source resolution	FullHD
Bitrate	746.496 Mbps

Camera zooms out from an apple tree with an average speed.

FIGURE 44: Apple Tree sequence, frame 30

B.3 "Behind Expedition"

Sequence title	Behind Expedition
Resolution	1920×1080
Number of frames	1047
Color space	YV12
Frames per second	30
Source	https://vimeo.com/204404590#t=0
Source resolution	FullHD
Bitrate	148.727 Mbps

Shipyard with view of large ships. Some scenes contain text overlaid by means of computer graphics.

FIGURE 45: Behind Expedition sequence, frame 25

B.4 "Cemetry"

Sequence title	Cemetry
Resolution	1920×1080
Number of frames	999
Color space	YV12
Frames per second	25
Source	https://vimeo.com/204151442#t=0
Source resolution	4K
Bitrate	112.49 Mbps

A series of person close-up shots. The camera zooms out slowly in the end of the video.

FIGURE 46: Cemetry sequence, frame 25

B.5 “Christmas Cats”

Sequence title	Christmas Cats
Resolution	1920×1080
Number of frames	1500
Color space	YV12
Frames per second	25
Source	https://vimeo.com/192252473#t=0
Source resolution	FullHD
Bitrate	191.087 Mbps

Concert record with superimposed complicated translucent CG effects.

FIGURE 47: Christmas Cats sequence, frame 25

B.6 “Chronicle”

Sequence title	Chronicle
Resolution	1920×1080
Number of frames	1113
Color space	YV12
Frames per second	30
Source	https://vimeo.com/123145218#t=164
Source resolution	FullHD
Bitrate	127.235 Mbps

Compilation of photos and video sequences. Most of the scenes have grain noise.

FIGURE 48: Chronicle sequence, frame 25

B.7 "City Crowd"

Sequence title	City Crowd
Resolution	1920×1080
Number of frames	763
Color space	YV12
Frames per second	30
Source resolution	FullHD
Bitrate	746.496 Mbps

City street with walking people and approaching tram. Static camera.

FIGURE 49: City Crowd sequence, frame 30

B.8 “Coffee Beans”

Sequence title	Coffee Beans
Resolution	1920×1080
Number of frames	1005
Color space	YV12
Frames per second	24
Source	https://vimeo.com/205129846#t=0
Source resolution	FullHD
Bitrate	198.561 Mbps

A walking person is filmed by hand-held camera, then the process of coffee roasting is shown. The camera is moving slowly most of the time. The video contains crossfades.

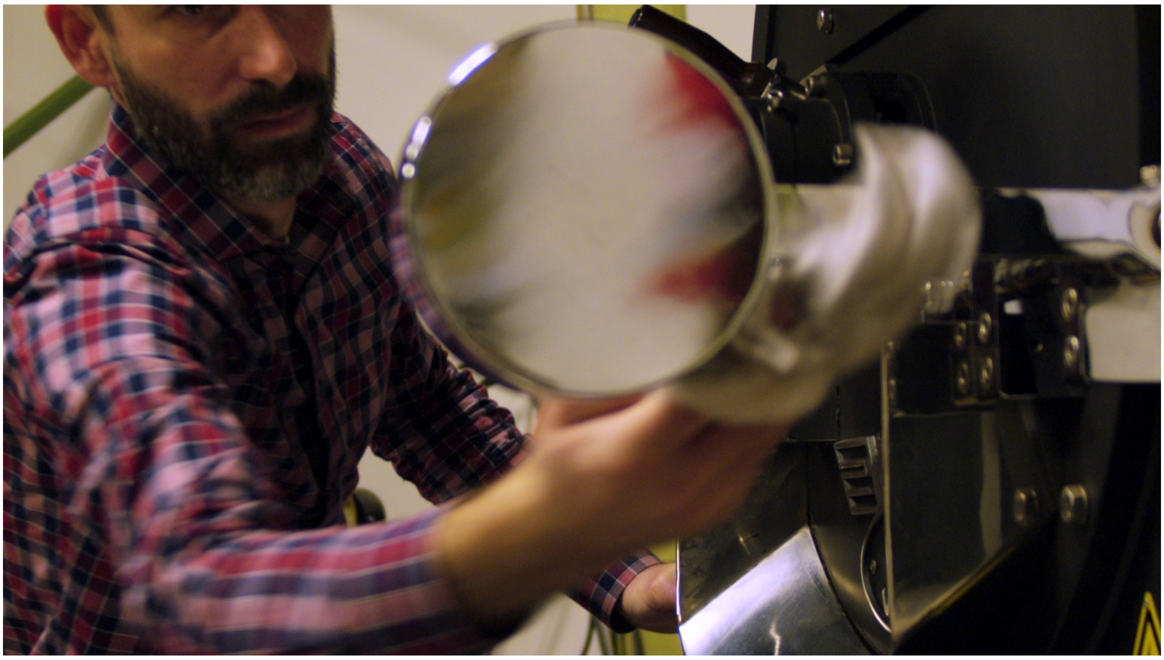


FIGURE 50: Coffee Beans sequence, frame 216

B.9 "Color Tune"

Sequence title	Color Tune
Resolution	1920×1080
Number of frames	1049
Color space	YV12
Frames per second	25
Source	https://vimeo.com/87772228#t=118
Source resolution	FullHD
Bitrate	113.19 Mbps

The video shows the same scene filmed by professional digital camera with different color settings.

FIGURE 51: Color Tune sequence, frame 25

B.10 “Crowd Run”

Sequence title	Crowd Run
Resolution	1920×1080
Number of frames	500
Color space	YV12
Frames per second	50
Source resolution	FullHD
Bitrate	1244.16 Mbps

A crowd of sportsmen runs while the camera slowly moves left and right.

FIGURE 52: Crowd Run sequence, frame 50

B.11 "Disneyland"

Sequence title	Disneyland
Resolution	1920×1080
Number of frames	317
Color space	YV12
Frames per second	24
Source	https://vimeo.com/152119430#t=0
Source resolution	4K
Bitrate	430.225 Mbps

Time lapse of disneyland castle located in a park with people. Camera slowly zooms in.

FIGURE 53: Disneyland sequence, frame 24

B.12 "Fire"

Sequence title	Fire
Resolution	1920×1080
Number of frames	601
Color space	YV12
Frames per second	25
Source resolution	FullHD
Bitrate	622.08 Mbps

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

FIGURE 54: Fire sequence, frame 25

B.13 “Forest Dog”

Sequence title	Forest Dog
Resolution	1920×1080
Number of frames	976
Color space	YV12
Frames per second	25
Source	https://vimeo.com/147443541#t=119
Source resolution	FullHD
Bitrate	200.535 Mbps

Macro shooting, the camera slowly changes focus. Then video shows a forest landscape, people and a dog.

FIGURE 55: Forest Dog sequence, frame 25

B.14 “Fountain”

Sequence title	Fountain
Resolution	1920×1080
Number of frames	516
Color space	YV12
Frames per second	25
Source	https://vimeo.com/92772980#t=0
Source resolution	4K
Bitrate	78.856 Mbps

Static camera captures people passing by in front of a fountain in a city.

FIGURE 56: Fountain sequence, frame 25

B.15 "Gilmour"

Sequence title	Gilmour
Resolution	1920×1080
Number of frames	957
Color space	YV12
Frames per second	30
Source	https://vimeo.com/188317665#t=28
Source resolution	FullHD
Bitrate	130.928 Mbps

Slideshow with various transition effects.

FIGURE 57: Gilmour sequence, frame 25

B.16 "Housing Group"

Sequence title	Housing Group
Resolution	1920×1080
Number of frames	1007
Color space	YV12
Frames per second	24
Source	https://vimeo.com/184904666#t=165
Source resolution	FullHD
Bitrate	62.951 Mbps

Compilation of landscape shots and shots of people talking and walking at both indoor and outdoor locations.

FIGURE 58: Housing Group sequence, frame 25

B.17 "Infinit"

Sequence title	Infinit
Resolution	1920×1080
Number of frames	258
Color space	YV12
Frames per second	25
Source	https://vimeo.com/180708512#t=0
Source resolution	FullHD
Bitrate	275.398 Mbps

The camera flies through CG buildings and statues, then video shows a picture of man with CG title on it.

FIGURE 59: Infinit sequence, frame 25

B.18 "Innershaq"

Sequence title	Innershaq
Resolution	1920×1080
Number of frames	1569
Color space	YV12
Frames per second	24
Source	https://vimeo.com/989385261#t=4
Source resolution	FullHD
Bitrate	56.064 Mbps

Animated cartoon combined with shooted video clips.

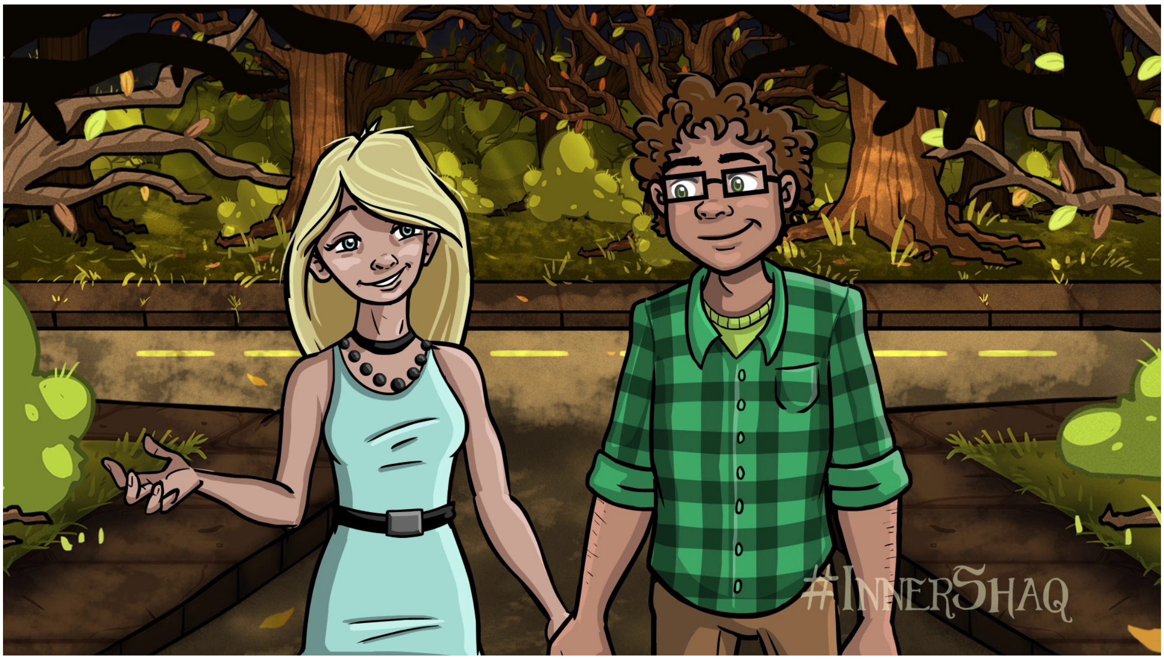


FIGURE 60: Innershaq sequence, frame 25

B.19 "Italian History"

Sequence title	Italian History
Resolution	1920×1080
Number of frames	989
Color space	YV12
Frames per second	24
Source	https://vimeo.com/207945404#t=292
Source resolution	FullHD
Bitrate	175.228 Mbps

City and nature views with an old film stock effects.

FIGURE 61: Italian History sequence, frame 25

B.20 "Mountain Bike"

Sequence title	Mountain Bike
Resolution	1920×1080
Number of frames	1063
Color space	YV12
Frames per second	24
Source	https://vimeo.com/188799676#t=38
Source resolution	FullHD
Bitrate	71.226 Mbps

The sequence films bikers riding in the forest. Consists of quadcopter shooting, slowmotion and close-up shots.

FIGURE 62: Mountain Bike sequence, frame 25

B.21 "Real Voters"

Sequence title	Real Voters
Resolution	1920×1080
Number of frames	997
Color space	YV12
Frames per second	24
Source	https://vimeo.com/188681554#t=83
Source resolution	FullHD
Bitrate	161.087 Mbps

The camera films close-ups of women dining and talking at a large table. The video contains frequent camera transitions.

FIGURE 63: Real Voters sequence, frame 25

B.22 "Road Runner"

Sequence title	Road Runner
Resolution	1920×1080
Number of frames	999
Color space	YV12
Frames per second	24
Source	https://vimeo.com/198635799#t=0
Source resolution	FullHD
Bitrate	118.639 Mbps

A music video consisting of timelapse, slowmotion, shaking camera and various close-up views of human movements.

FIGURE 64: Road Runner sequence, frame 25

B.23 "Roseman Bridge"

Sequence title	Roseman Bridge
Resolution	1920×1080
Number of frames	2549
Color space	YV12
Frames per second	30
Source	https://vimeo.com/130709443#t=49
Source resolution	4K
Bitrate	60 Mbps

The bridge filmed from the quadcopter. The camera moves slowly in different directions.

FIGURE 65: Roseman Bridge sequence, frame 25

B.24 “Sea Lions”

Sequence title	Sea Lions
Resolution	1920×1080
Number of frames	1293
Color space	YV12
Frames per second	24
Source	https://vimeo.com/171580634#t=0
Source resolution	4K
Bitrate	268.939 Mbps

Shots of sea lion pups and water surface with superimposed text.

FIGURE 66: Sea Lions sequence, frame 96

B.25 "Shakewalk"

Sequence title	Shakewalk
Resolution	1920×1080
Number of frames	805
Color space	YV12
Frames per second	25
Source resolution	FullHD
Bitrate	622.08 Mbps

A man walking in the park and holding camera in front of him and shaking this camera a lot.

FIGURE 67: Shakewalk sequence, frame 25

B.26 “Sita”

Sequence title	Sita
Resolution	1920×1080
Number of frames	1000
Color space	YV12
Frames per second	25
Source resolution	FullHD
Bitrate	622.08 Mbps

Part of a cartoon movie “Sita sings the blues”. Contains a lot of contrast shapes with hard edges. Scenes contain only monotonous movement.

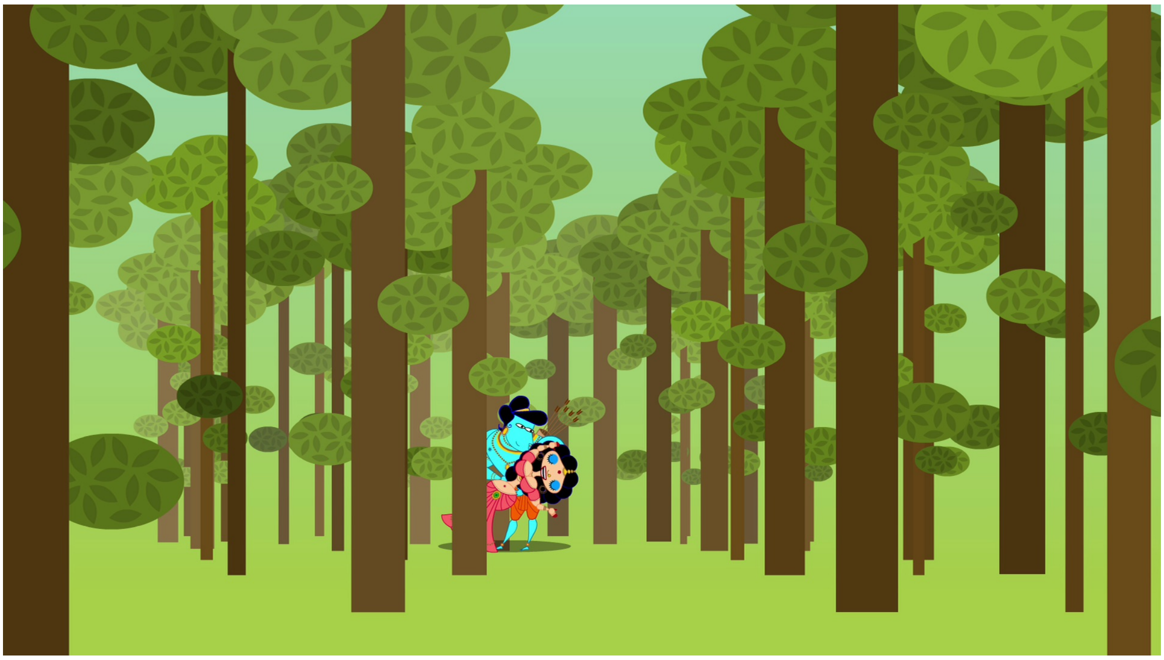


FIGURE 68: Sita sequence, frame 25

B.27 "Skiers"

Sequence title	Skiers
Resolution	1920×1080
Number of frames	1370
Color space	YV12
Frames per second	60
Source	https://vimeo.com/202605482#t=152
Source resolution	4K
Bitrate	120.597 Mbps

The sequence shows the group of skiers in mountains shot by head-mounted camera, then the same group is shot by quadcopter.

FIGURE 69: Skiers sequence, frame 25

B.28 "Steadicam"

Sequence title	Steadicam
Resolution	1920×1080
Number of frames	979
Color space	YV12
Frames per second	24
Source	https://vimeo.com/118449040#t=0
Source resolution	4K
Bitrate	118.699 Mbps

Interior of a church captured with steadicam.

FIGURE 70: Steadicam sequence, frame 96

B.29 "Twin Strangers"

Sequence title	Twin Strangers
Resolution	1920×1080
Number of frames	1026
Color space	YV12
Frames per second	25
Source	https://vimeo.com/194961299#t=0
Source resolution	FullHD
Bitrate	128.146 Mbps

Non-professional videoblog with simple CG and subtitles.

FIGURE 71: Twin Strangers sequence, frame 25

B.30 "Wedding"

Sequence title	Wedding
Resolution	1920×1080
Number of frames	948
Color space	YV12
Frames per second	24
Source	https://vimeo.com/180841074#t=625
Source resolution	FullHD
Bitrate	112.827 Mbps

Outdoor shooting of a wedding. The camera changes view several times.

FIGURE 72: Wedding sequence, frame 25

B.31 "Ziguinchor"

Sequence title	Ziguinchor
Resolution	1920×1080
Number of frames	994
Color space	YV12
Frames per second	25
Source	https://vimeo.com/184550115#t=120
Source resolution	FullHD
Bitrate	259.92 Mbps

Indoor and outdoor shooting of people's conversations.

FIGURE 73: Ziguinchor sequence, frame 25

C SEQUENCES SELECTION

In "MSU Video Codecs Comparison 2016" we introduced a new technique for test dataset sequences' selection. This technique was designed to create dataset containing representative set of sequences that encoders are facing in everyday life. In this report we use the same methodology for video sequences selection, but we have dramatically updated video database from which we sample videos for encoders' comparison.

We analyzed over 512,000 videos hosted at Vimeo looking for 4K and FullHD videos with high bitrates (50 Mbps was selected as a lower bitrate boundary). This enabled us to find and download, 662 new 4K videos and 1993 new FullHD videos. The bitrate distributions for previous year dataset and updated dataset are shown in Figure 74.

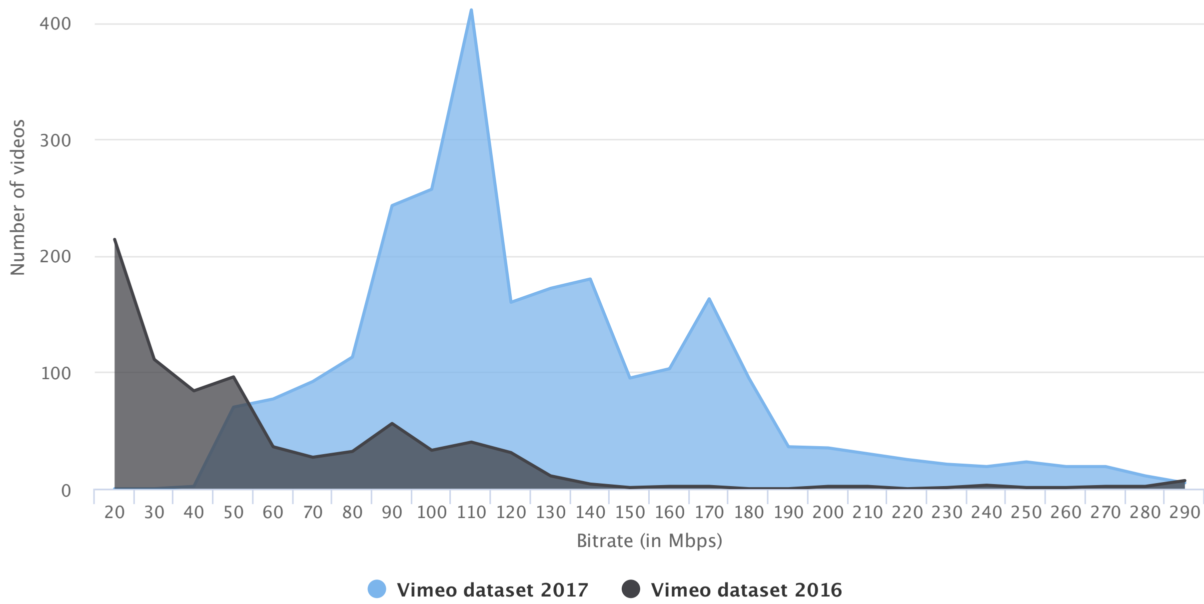


FIGURE 74: Bitrate distributions for videos dataset

We resized and cropped 4K videos to FullHD resolution to ensure the absence of compression artifacts. All videos were cut at scene change points to samples, with 1000 frames approximate length. Besides 6390 samples from 2655 newly downloaded videos, we also used 2900 samples from "MSU Video Codecs Comparison 2016". Thus, our samples database for this year consisted of 9290 items.

To evaluate spatial and temporal complexity we encoded all samples using x264 encoder with constant quantization parameter (QP). For all samples temporal and spatial complexity were calculated. We define spatial complexity as average size of I-frame normalized by sample's uncompressed frame size. Temporal complexity is defined as average size of P-frame divided by average size of I-frame.[1] Distribution of obtained samples compared to samples from previous codec comparison is shown in Figure 75.

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

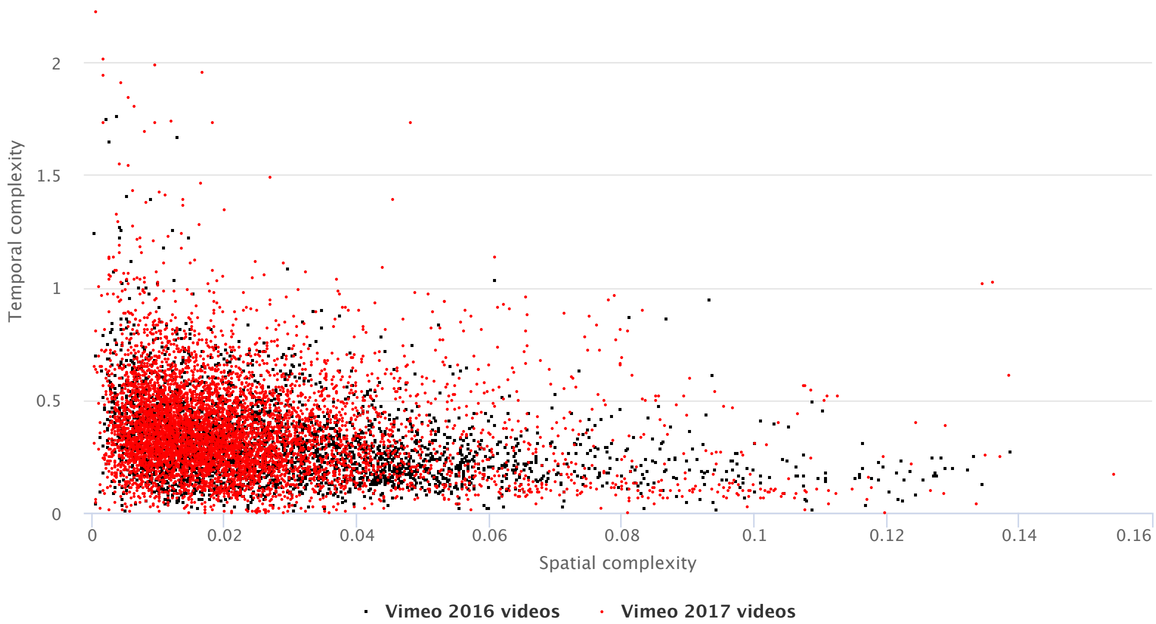


FIGURE 75: Distribution of obtained samples

Figure 75 reveals that new samples have similar distribution to samples from "MSU Video Codecs Comparison 2016". In order to prepare dataset we used the following process.

We divided the video database into 31 clusters with K-Means. To avoid complete update of sequences list, sequences from last year's FullHD dataset were given 10 times higher weight compared to other sequences. For each cluster we selected the video sequence closest to its center that has a license enabling derivatives and commercial usage. The clusters' boundaries and chosen sequences are shown at Figure 76.

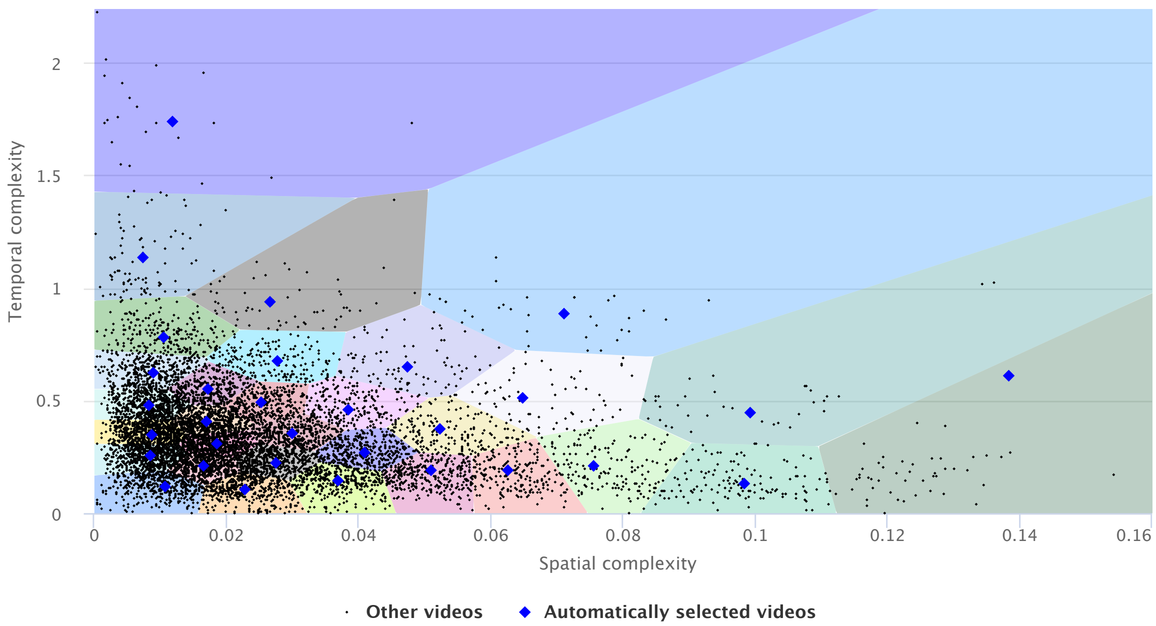


FIGURE 76: Segmentation of the samples

At Figure 77 we show correspondence of sequences from prior dataset to newly selected clusters. As can be seen from the figure, there are some clusters not covered by videos from old dataset.

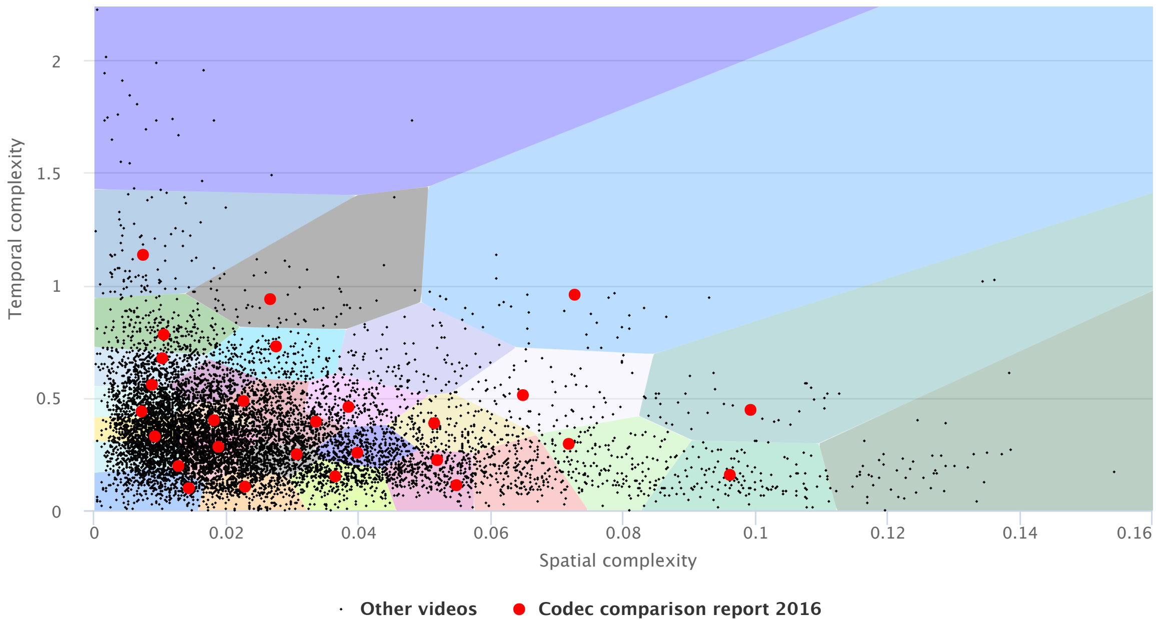


FIGURE 77: Segmentation of the samples compared to old dataset

Some of automatically chosen samples contain company names or have another copyright issues, so we replaced that samples with other samples in that clusters with suitable license. Figure 78 illustrates applied adjustments.

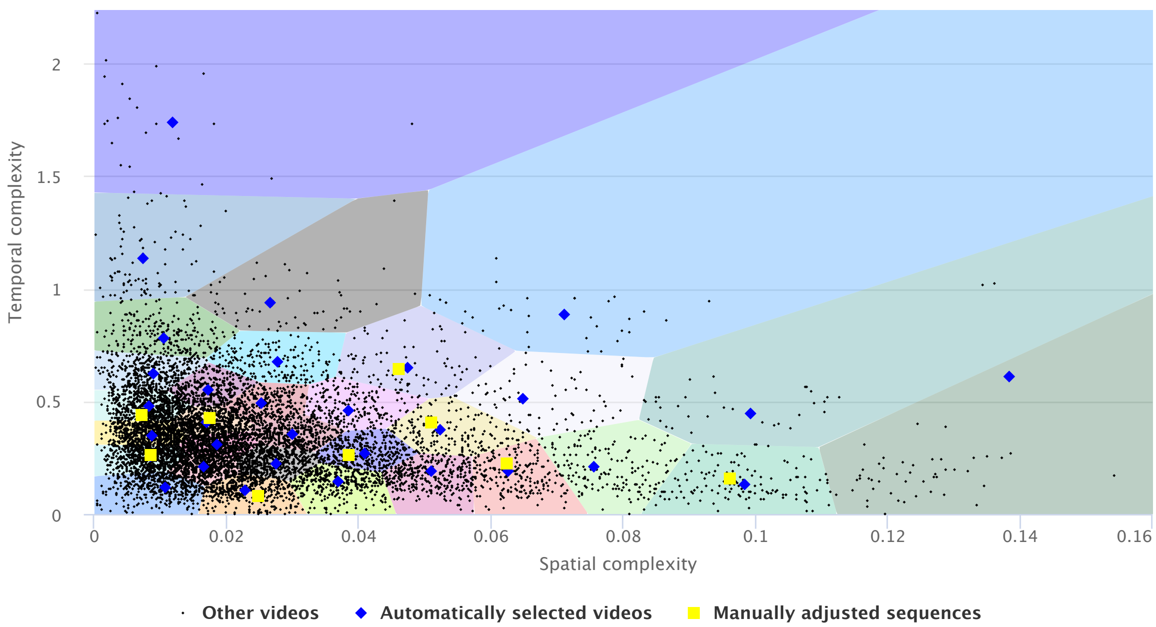


FIGURE 78: Adjustments to test dataset

Figure 79 shows final distribution of sequences in the dataset used in this report.

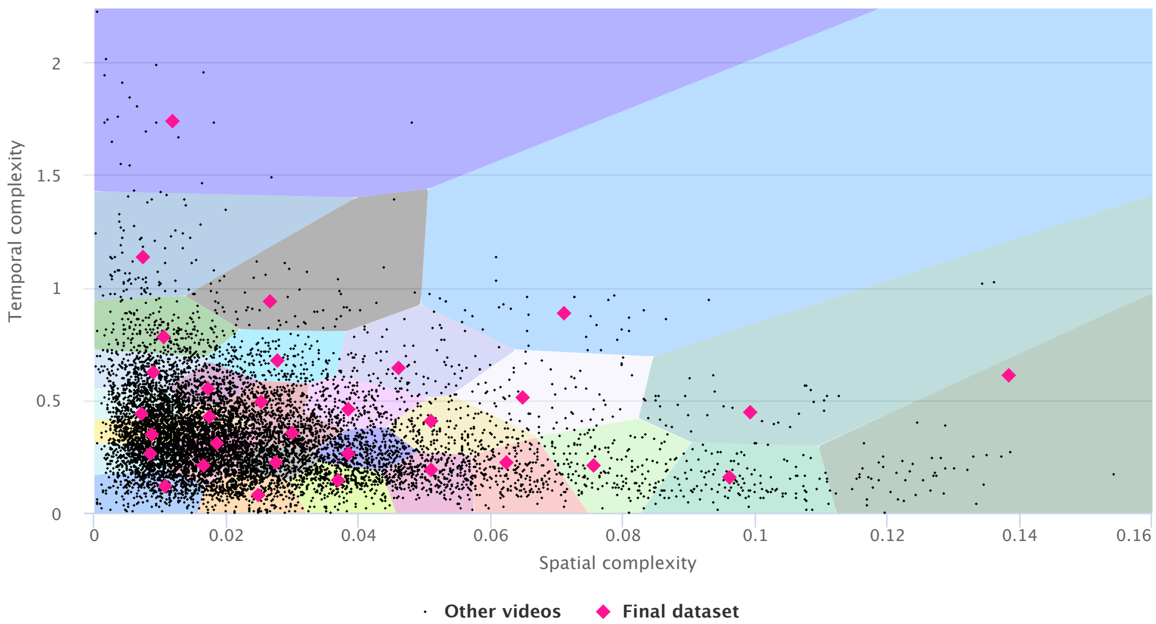


FIGURE 79: Distribution of sequences in final dataset

New dataset consists of 31 video sequences: 8 videos from old dataset, and 23 new videos from Vimeo. 19 sequences from old dataset were excluded. Average bitrate of all sequences in the final dataset is 272.79 Mbps. "Innershaq" (56 Mbps) and "Roseman Bridge" (60 Mbps) sequences have minimal bitrates, notably both of them have small temporal complexity. The complete list of sequences from new dataset can be found in Appendix B.

D CODECS

D.1 SIF-1

Encoder title	SIF
Version	1.43.0
Developed by	SIF Encoder Team

Preset name	Encoder parameters
Universal	`ConsoleEnc.exe %SOURCE_FILE% --fps=%FPS_TMP%/1000 --me_mode=ultrafast --sub_me_mode=fastest --comp_mode=vbr_all_p --out_bitrt=%BITRATE_KBPS% --rc_buf_s=6 --entropy_mode=8_threads -w %WIDTH% -h %HEIGHT% -o %TARGET_FILE%.avi`
Ripping	`ConsoleEnc.exe %SOURCE_FILE% --fps=%FPS_TMP%/1000 --comp_mode=vbr_all_p --sub_me_mode=fastest --out_bitrt=%BITRATE_KBPS% --rc_buf_s=2 --entropy_mode=8_threads -w %WIDTH% -h %HEIGHT% -o %TARGET_FILE%.avi`

D.2 x264

Encoder title	x264
Version	r2833 df79067
Developed by	x264 Developer Team

Preset name	Encoder parameters
Reference	`x264 --tune ssim --preset veryslow --bitrate %BITRATE_KBPS% %SOURCE_FILE% -o %TARGET_FILE% --input-res %WIDTH%x%HEIGHT% --fps %FPS%`
Fast	`x264 --preset fast --subme 4 --b-adapt 0 --keyint infinite --tune ssim --pass 1 --bitrate %BITRATE_KBPS% %SOURCE_FILE% --input-res %WIDTH%x%HEIGHT% --fps %FPS% -o NUL` `x264 --preset fast --subme 4 --b-adapt 0 --keyint infinite --tune ssim --pass 2 --bitrate %BITRATE_KBPS% %SOURCE_FILE% --input-res %WIDTH%x%HEIGHT% --fps %FPS% -o %TARGET_FILE%`
Universal	`x264 --preset slow --me hex --trellis 2 --subme 9 --keyint infinite --tune ssim --pass 1 --bitrate %BITRATE_KBPS% %SOURCE_FILE% --input-res %WIDTH%x%HEIGHT% --fps %FPS% -o NUL`

	x264 --preset slow --me hex --trellis 2 --subme 9 --keyint infinite --tune ssim --pass 2 --bitrate %BITRATE_KBPS% %SOURCE_FILE% --input-res %WIDTH%x%HEIGHT% --fps %FPS% -o %TARGET_FILE%
Ripping	x264 --preset placebo --me umh --merange 32 --keyint infinite --tune ssim --pass 1 --bitrate %BITRATE_KBPS% %SOURCE_FILE% --input-res %WIDTH%x%HEIGHT% --fps %FPS% -o NUL x264 --preset placebo --me umh --merange 32 --keyint infinite --tune ssim --pass 2 --bitrate %BITRATE_KBPS% %SOURCE_FILE% --input-res %WIDTH%x%HEIGHT% --fps %FPS% -o %TARGET_FILE%

D.3 x265

Encoder title	x265
Version	2.3+23-97435a0870befe35
Developed by	x265 Developer Team

Preset name	Encoder parameters
Fast	x265_64-8bit[gcc] -p ultrafast --tune ssim --me 1 --ref 2 --limit-refs 3 --signhide --b-intra --bitrate %BITRATE_KBPS% --ssim %SOURCE_FILE% -o %TARGET_FILE% --input-res %WIDTH%x%HEIGHT% --fps %FPS%
Universal	x265_64-8bit[gcc] -p medium --tune ssim --rd 2 --early-skip --bframes 3 --max-merge 3 --ref 4 --b-intra --bitrate %BITRATE_KBPS% --ssim %SOURCE_FILE% -o %TARGET_FILE% --input-res %WIDTH%x%HEIGHT% --fps %FPS%
Ripping	x265_64-8bit[gcc] -p veryslow --tune ssim --bitrate %BITRATE_KBPS% --ssim %SOURCE_FILE% -o %TARGET_FILE% --input-res %WIDTH%x%HEIGHT% --fps %FPS%

D.4 nj264

Encoder title	nj264
Version	V1.0
Developed by	Nanjing Yunyan

The encoder is recipient of the Frost & Sullivan 2016 Global Enabling Technology Leadership of the Year Award for AVC Video Encoding.

Preset name	Encoder parameters
Fast	`nj264.exe -s %WIDTH%x%HEIGHT% -framerate %FPS% -i %SOURCE_FILE% -c:v libnj264 -preset speed -nj264-params bitrate=%BITRATE_KBPS% -f h264 -y %TARGET_FILE%`
Universal	`nj264.exe -s %WIDTH%x%HEIGHT% -framerate %FPS% -i %SOURCE_FILE% -c:v libnj264 -preset balanced -nj264-params bitrate=%BITRATE_KBPS% -f h264 -y %TARGET_FILE%`
Ripping	`nj264.exe -s %WIDTH%x%HEIGHT% -framerate %FPS% -i %SOURCE_FILE% -c:v libnj264 -preset ripping -nj264-params bitrate=%BITRATE_KBPS% -f h264 -y %TARGET_FILE%`

D.5 nj265

Encoder title	nj265
Version	V1.0
Developed by	Nanjing Yunyan

Preset name	Encoder parameters
Fast	`nj265.exe -s %WIDTH%x%HEIGHT% -framerate %FPS% -i %SOURCE_FILE% -c:v libnj265 -preset speed -nj265-params bitrate=%BITRATE_KBPS% -f hevc -y %TARGET_FILE%`
Universal	`nj265.exe -s %WIDTH%x%HEIGHT% -framerate %FPS% -i %SOURCE_FILE% -c:v libnj265 -preset balanced -nj265-params bitrate=%BITRATE_KBPS% -f hevc -y %TARGET_FILE%`
Ripping	`nj265.exe -s %WIDTH%x%HEIGHT% -framerate %FPS% -i %SOURCE_FILE% -c:v libnj265 -preset ripping -nj265-params bitrate=%BITRATE_KBPS% -f hevc -y %TARGET_FILE%`

D.6 KS265

Encoder title	Kingsoft Encoder
Version	V2.5.2
Developed by	Kingsoft

Preset name	Encoder parameters
Fast	`AppEncoder_x64.exe -i %SOURCE_FILE% -preset medium -threads 0 -inxn 0 -ctx 0 -topdownth 40 -ltr 0 -asr 8 -me 2 -isubfac 20 -lratio 70 -mratio 40 -wdt %WIDTH% -hgt %HEIGHT% -fr %FPS% -br %BITRATE_KBPS% -b %TARGET_FILE%`
Universal	`AppEncoder_x64.exe -i %SOURCE_FILE% -preset slow -threads 0 -rdoq 1 -part 0 -skuv 0 -inxn 0 -wdt %WIDTH% -hgt %HEIGHT% -fr %FPS% -br %BITRATE_KBPS% -b %TARGET_FILE%`
Ripping	`AppEncoder_x64.exe -i %SOURCE_FILE% -preset placebo -threads 0 -wdt %WIDTH% -hgt %HEIGHT% -fr %FPS% -br %BITRATE_KBPS% -b %TARGET_FILE%`

D.7 uAVS2

Encoder title	uAVS2 Encoder
Version	V1.0
Developed by	Digital Media R&D Center, Peking University, Shenzhen Graduate School

Preset name	Encoder parameters
Fast	`Fast\utest_x64.exe -f Fast\encoder_ra.cfg -p InputFile=%SOURCE_FILE% -p OutputFile=%TARGET_FILE% -p SourceWidth=%WIDTH% -p SourceHeight=%HEIGHT% -p FrameRate=%FPS% -p FramesToBeEncoded=%FRAMES_NUM% -p TargetBitRate=%BITRATE_KBPS%`
Universal	`Universal\utest_x64.exe -f Universal\encoder_ra.cfg -p InputFile=%SOURCE_FILE% -p OutputFile=%TARGET_FILE% -p SourceWidth=%WIDTH% -p SourceHeight=%HEIGHT% -p FrameRate=%FPS% -p FramesToBeEncoded=%FRAMES_NUM% -p TargetBitRate=%BITRATE_KBPS%`
Ripping	`Ripping\utest_x64.exe -f Ripping\encoder_ra.cfg -p InputFile=%SOURCE_FILE% -p OutputFile=%TARGET_FILE% -p SourceWidth=%WIDTH% -p SourceHeight=%HEIGHT% -p FrameRate=%FPS% -p FramesToBeEncoded=%FRAMES_NUM% -p TargetBitRate=%BITRATE_KBPS%`

E FIGURES EXPLANATION

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

E.1 RD Curves

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

E.2 Relative Bitrate/Relative Time Charts

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

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

E.3 Graph Example

Figure 80 shows a case where these graphs can be useful. In the top left graph, it is apparent that the "Green" codec encodes with significantly 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: it is clearly visible in the bottom graph that one of the codecs is slower, but yields higher visual quality, and that the other codec is faster, but yields lower visual quality.

As a result of these advantages, relative bitrate/relative time graphs are used frequently in this report since they assist in the evaluation of the codecs in the test set, especially when number of codecs is large.

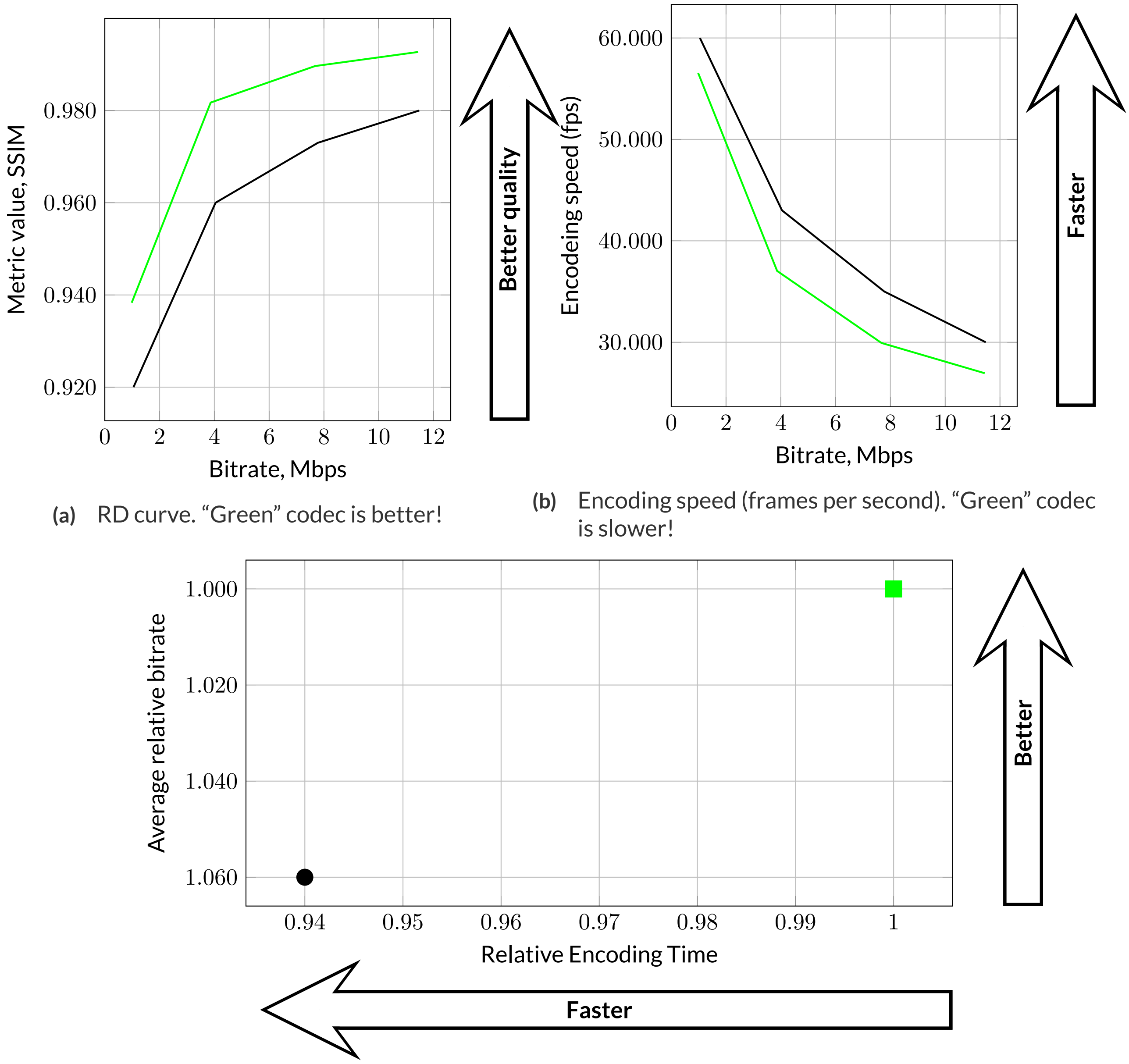
A more detailed description of the preparation of these graphs is given below.

E.4 Bitrates Ratio with the Same Quality

The first step in computing the average bitrate ratio for a fixed quality is inversion of the axes of the bitrate/quality graph (see Figure 81b). All further computations are performed using the inverted graph.

The second step involves averaging the interval over which the quality axis is chosen. Averaging is performed only over those segments for which there are results for both codecs. This limitation is due to the difficulty of developing extrapolation methods for classic RD curves; nevertheless, for interpolation of RD curves, even linear methods are acceptable.

The final step is calculation of the area under the curves in the chosen interpolation segment and determination of their ratio (see Figure 81c). This result is an average bitrate ratio for a fixed quality for the two codecs. If

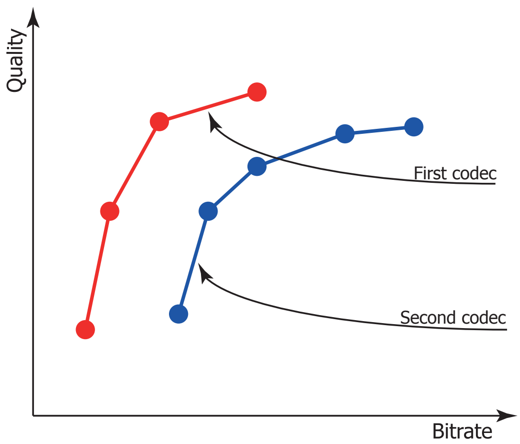


(a) RD curve. "Green" codec is better!

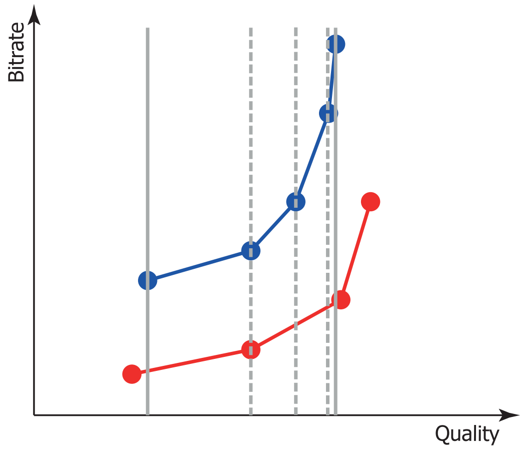
(b) Encoding speed (frames per second). "Green" codec is slower!

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

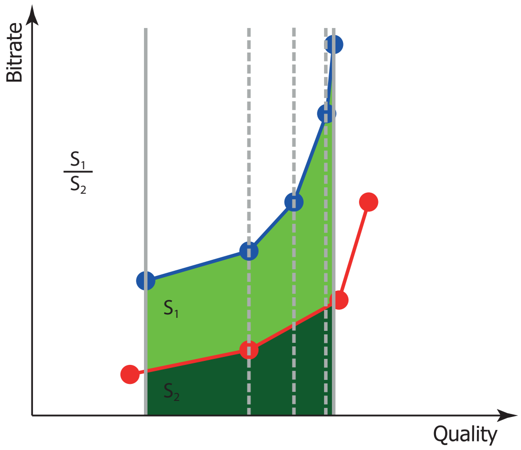
FIGURE 80: Speed/Quality trade-off example



(a) Source RD curves



(b) Axes' inversion and averaging interval choosing



(c) Areas under curves ratio

FIGURE 81: Average bitrate ratio computation

more than two codecs are considered, then one of them is defined as a reference codec and the quality of others is compared to that of the reference.

E.5 Relative Quality Analysis

While most figures in this report provide codec scores relative to reference encoder (i.e. x264) the "Relative Quality Analysis" sections show bitrate ratio with fixed quality (see Section E.4) score for each codec pair. This might be useful if one is interested in comparison of codec A relative to codec B only.

Below we show simplified example of "Average bitrate ratio for a fixed quality" table for two codecs only:

	A	B
A	100% ☺	75% ☹
B	134% ☹	100% ☺

TABLE 7: Example of average bitrate ratio for a fixed quality table

Let's consider column "B" row "A" of the table containing value 75% this should be read in the following way: average bitrate for a fixed quality of codec B is 75% less relative to codec A. The icon in the cell depicts confidence of this estimate. If projections of codecs' RD curves on quality axis (see Figure 81) have relatively large common area you will see happy icon. If size of this intersection is small and thus bitrate score can't be computed reliably the sad icon will be shown.

"Average bitrate ratio for a fixed quality" plots are visualizations of these tables. Each line in such plot depicts values from one column of corresponding table.

F OBJECTIVE QUALITY METRICS DESCRIPTION

F.1 SSIM (Structural SIMilarity)

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

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

Brief description of SSIM metric computation is given below.

F.1.1 Brief Description

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

The scheme of SSIM calculation can be presented as follows. 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 the following:

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

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 grayscale images), and $K1, K2 \ll 1$.

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

For the implementation used in this comparison, 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 of

[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.

the advantages of the SSIM metric is that it better represents human visual perception than does PSNR. SSIM is more complex, however, and takes more time to calculate.

F.1.2 Examples

Figure 82 shows the example of an SSIM result for an original and processed (compressed with lossy compression) image. The resulting value of 0.9 demonstrates that the two images are very similar.

(a) Original

(b) Compressed

(c) SSIM

FIGURE 82: SSIM example for compressed image

Figure 83 depicts various distortions applied to original image and Figure 84 shows SSIM values for these distortions.

(a) Original image

(b) Image with added noise

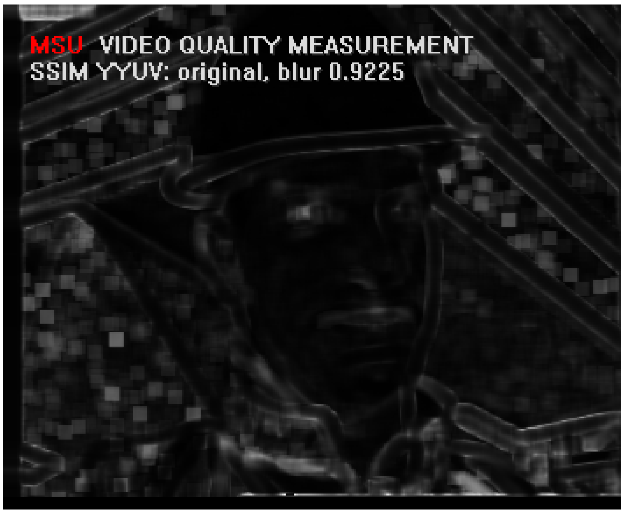
(c) Blurred image

(d) Sharpen image

FIGURE 83: 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 84: SSIM values for original and processed images

G ABOUT THE GRAPHICS & MEDIA LAB VIDEO GROUP

The Graphics & Media Lab Video Group is part of the Computer Science Department of 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`

H LIST OF MINARY FIXES

We are sorry for mistakes and formatting defects in the release version of our report. This year we used new version of report generation system, that caused some inaccuracies passed while manual report checking. In this report version the following mistakes were corrected:

1. x265 codec version was unified and corrected in all mentions and report parts. Before this changes, some of the x265 mentions included an old (1.9+169-e5b5bdc3c154) version. This happened due to cut&paste from previous 2016 report and some mentions was passed while changing to a correct version (2.3+23-97435a0870befe35)
2. The name uAVS2 was corrected on the title page of Part 1
3. In Part 3, overlapping of x264 description was fixed (in an appendix with codecs)
4. In Part 4, text overlapping in Section 2 (with codecs descriptions) was corrected
5. List of video sequences and their descriptions were completed in Part 4
6. All screenshots from all sequences were converted to JPEG due to make the PDF file size smaller

MSU Video Quality Measurement Tool

MSU Graphics & Media Lab. Video Group.

GRAPHICS & MEDIA LAB VIDEO GROUP

3 reasons to use VQMT:

- **Fastest implementation of VMAF**
- **Fastest SSIM/MS-SSIM speed on 4K/8K video**
- **Professional 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

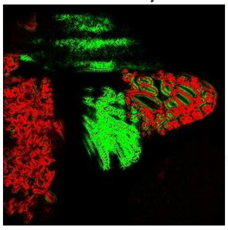
2. Fastest Video Quality Measurement

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

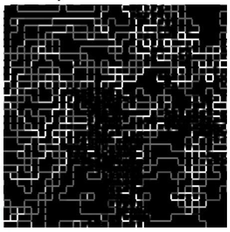
2.2 Multi-core Processors Support

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:

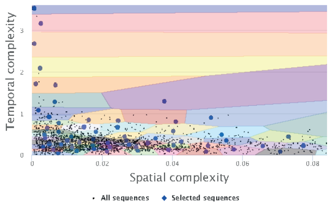
Reduce video file size or encoding speed with optimal codec settings

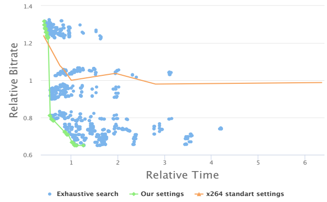
Effective Video Transcoding

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

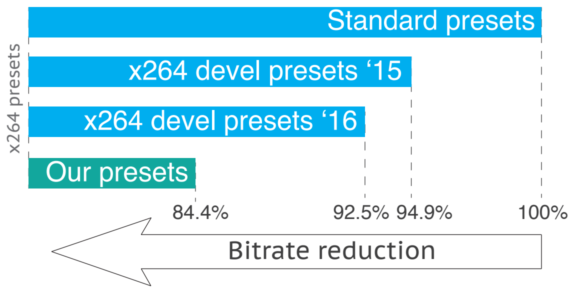




Full-size charts are available on our [project page]

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