HEVC/H.265 Video Codecs Comparison

Graphics & Media Lab
Video Group

4K Appendix

Video group head:	Dr. Dmitriy Vatolin
Project head:	Dr. Dmitriy Kulikov
Measurements & analysis:	Mikhail Erofeev,
	Stanislav Dolganov

Codecs:

H.265

- Intel® MSS HEVC GPU-acc Encoder
- nj265
- SHBP H.265 Real time encoder
- x265

Non H.265

- nj264
- x264

CS MSU Graphics & Media Lab, Video Group
February 19, 2016
`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:

- Intel Corporation
- ”System house ”Business partners” company
- x264 developer team
- x265 developer team
- Nanjing Yunyan

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.	Canary Wharf	1722	24	4096 × 2304
2.	Coastguard	240	30	3840 × 2160
3.	Cobra	352	30	3840 × 2160
4.	Driving	1747	24	4096 × 2160
5.	Foreman	248	24	3840 × 2160
6.	Little Girl	1531	30	4096 × 2160
7.	Mobile	355	24	3840 × 2160
8.	News	256	24	3840 × 2160
9.	Sintel	2000	24	4096 × 1744
10.	Susie	588	30	3840 × 2160
11.	Dirt Trail	3426	24	3840 × 2160

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 A.

2.2 Codecs

	Codec	Developer	Version
1.	Intel® MSS HEVC GPU-acc Encoder	Intel	Intel Media Server Studio 2015 R7 – Professional Edition
2.	nj264	Nanjing Yunyan Email: jtwen@tsinghua.edu.cn	1.0
3.	nj265	Nanjing Yunyan Email: jtwen@tsinghua.edu.cn	1.0
4.	SHBP H.265 Real time encoder	SHBP Codec's development team Email: lobasso@hotmail.com	0.8
5.	x264	x264 Developer Team	0.148.2638 7599210
6.	x265	MulticoreWare, Inc.	1.8+127-687f397dcd65

TABLE 2: Short codec 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 B.

3 OBJECTIVES AND TESTING RULES

The main goal of this report is the presentation of a comparative evaluation of the quality of new HEVC codecs and codecs of other standards using objective measures of assessment. All test video sequences were 4K video sequences. The reported study complements previously released report containing HEVC encoders evaluation results on 2K content. The comparison was done using settings provided by the developers of each codec. Nevertheless, we required all presets to satisfy minimum speed requirements. The main task of the comparison is to analyze different encoders for the task of transcoding video—e.g., compressing video for personal use. The comparison was performed on Core i7 4770R @3.9 GHz, RAM 8 GB, Windows 8.1.

4 RD CURVES

Intel MSS HEVC GAcc encoder takes the first place on most of the test sequences according to RD-curves below. x265 is typically the second at average.

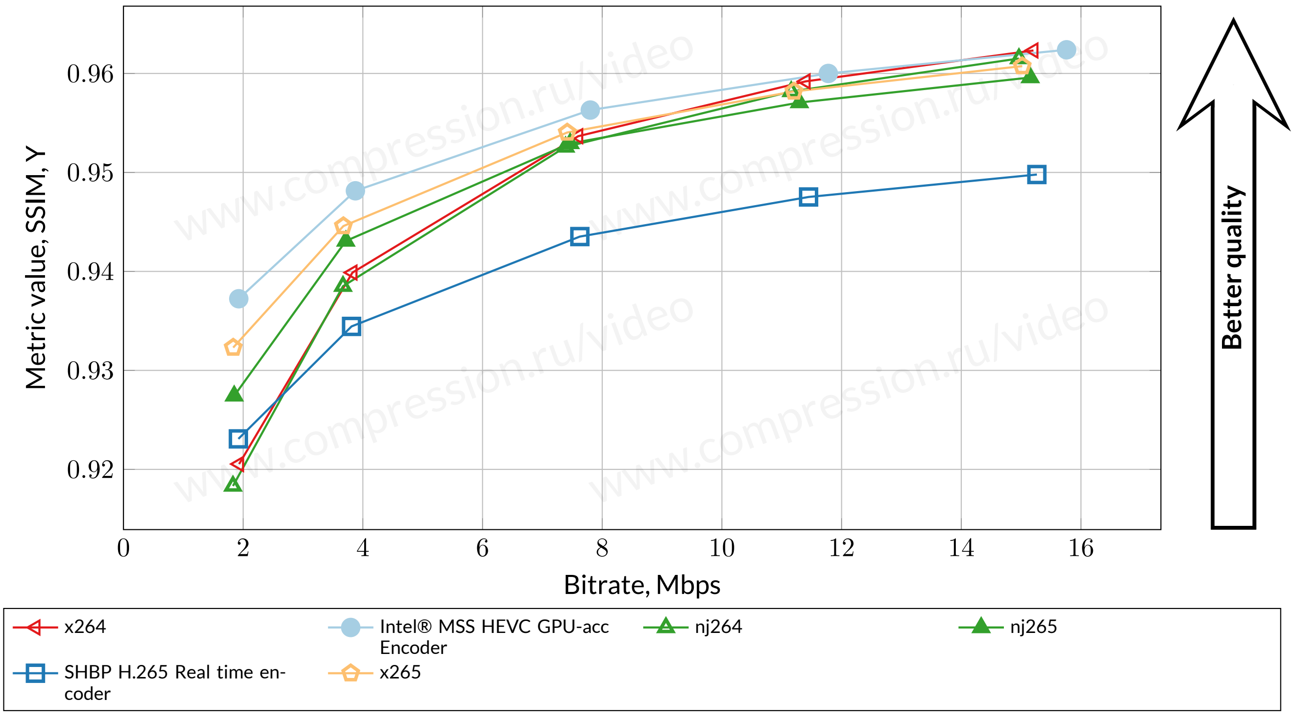


FIGURE 1: Bitrate/quality—usecase "4K Encoding," Driving sequence, Y-SSIM metric

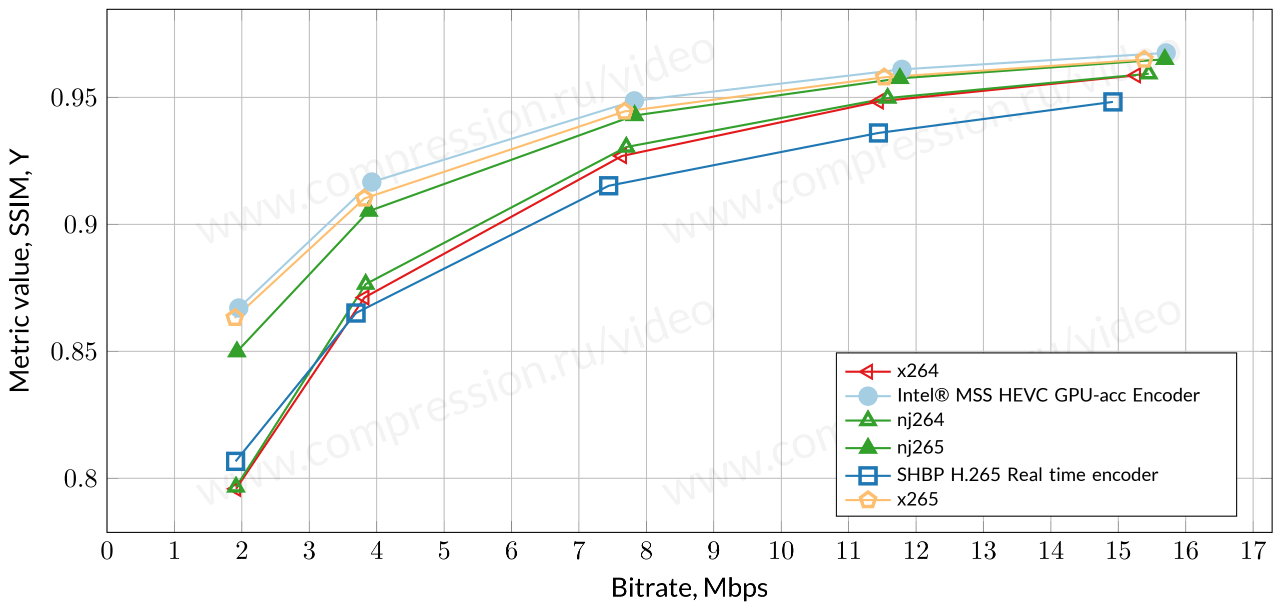


FIGURE 2: Bitrate/quality—usecase "4K Encoding," Dirt Trail sequence, Y-SSIM metric

5 ENCODING SPEED

SHBP H.265 Real time encoder is the first by encoding speed and it is the only encoder with constant speed at all bitrates. The next places are for Intel MSS HEVC GAcc and x265.

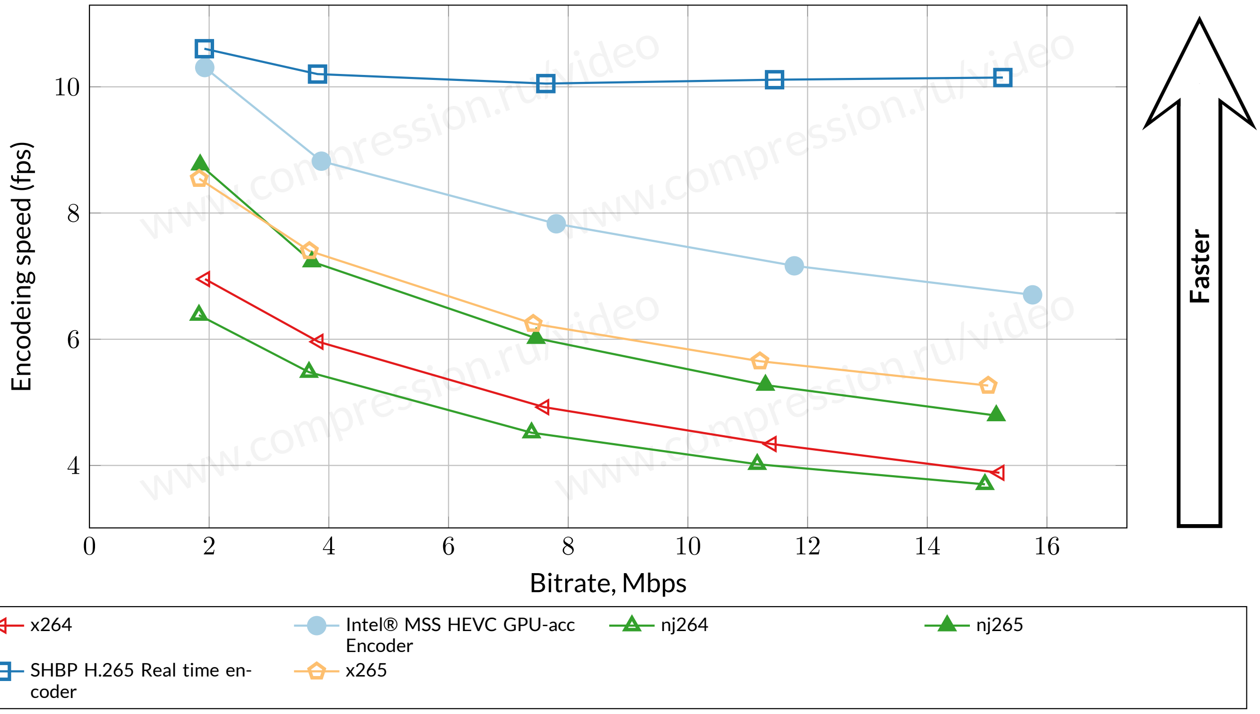


FIGURE 3: Encoding speed—usecase "4K Encoding," Driving sequence

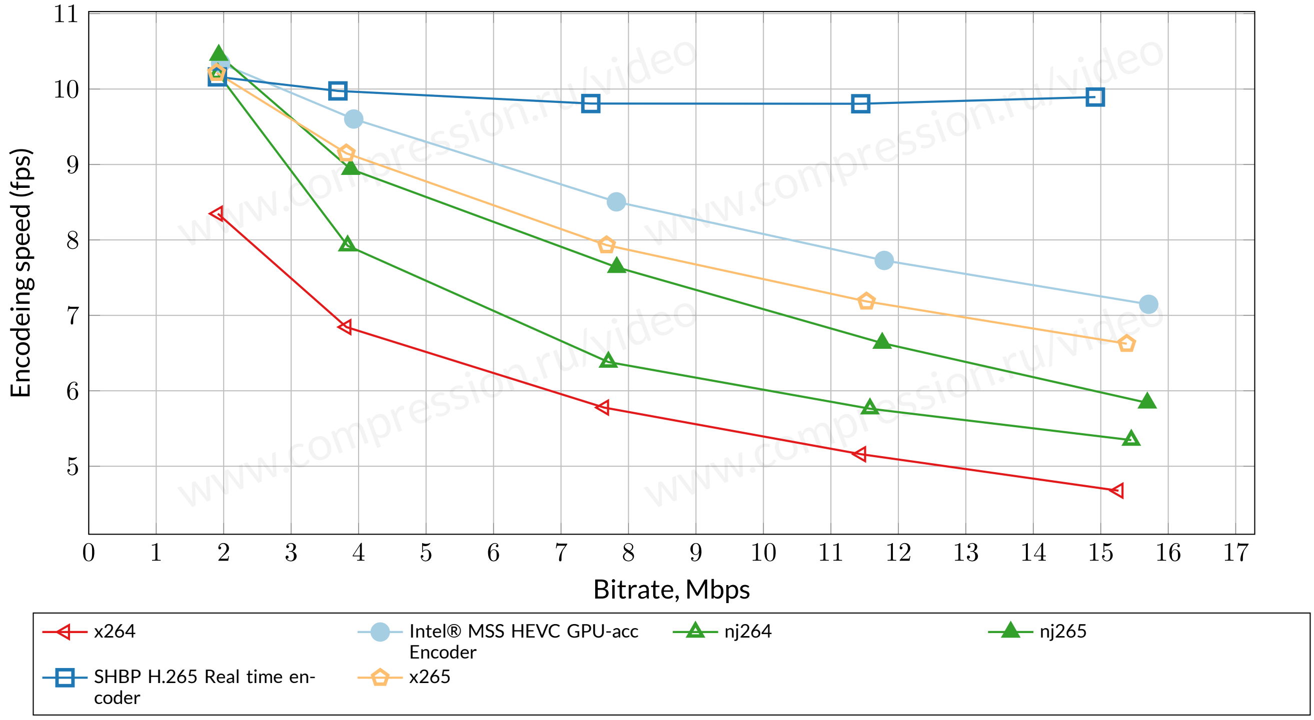


FIGURE 4: Encoding speed—usecase "4K Encoding," Dirt Trail sequence

6 SPEED/QUALITY TRADE-OFF

Detailed descriptions of the speed/quality trade-off graphs can be found in Appendix C. Sometimes, codec results are not present in the particular graph owing to the codec's extremely poor performance. The codec's RD curve has no intersection with the reference's RD curve.

The speed/quality trade-off graphs simultaneously show relative quality and encoding speed for the encoders tested in this comparison. x264 is the reference codec, for which both quality and speed are normalized to unity for all of the graphs. The terms "better" and "worse" are used to compare codecs in the same manner as in previous portions of this comparison.

Pareto optimal encoders in terms of speed and quality (at average) are SHBP H.265 Real time encoder and Intel MSS HEVC GAcc encoders. But this situation slightly differs from sequence to sequence. "Pareto optimal" encoder means there are no encoder that is faster and better then current in this test.

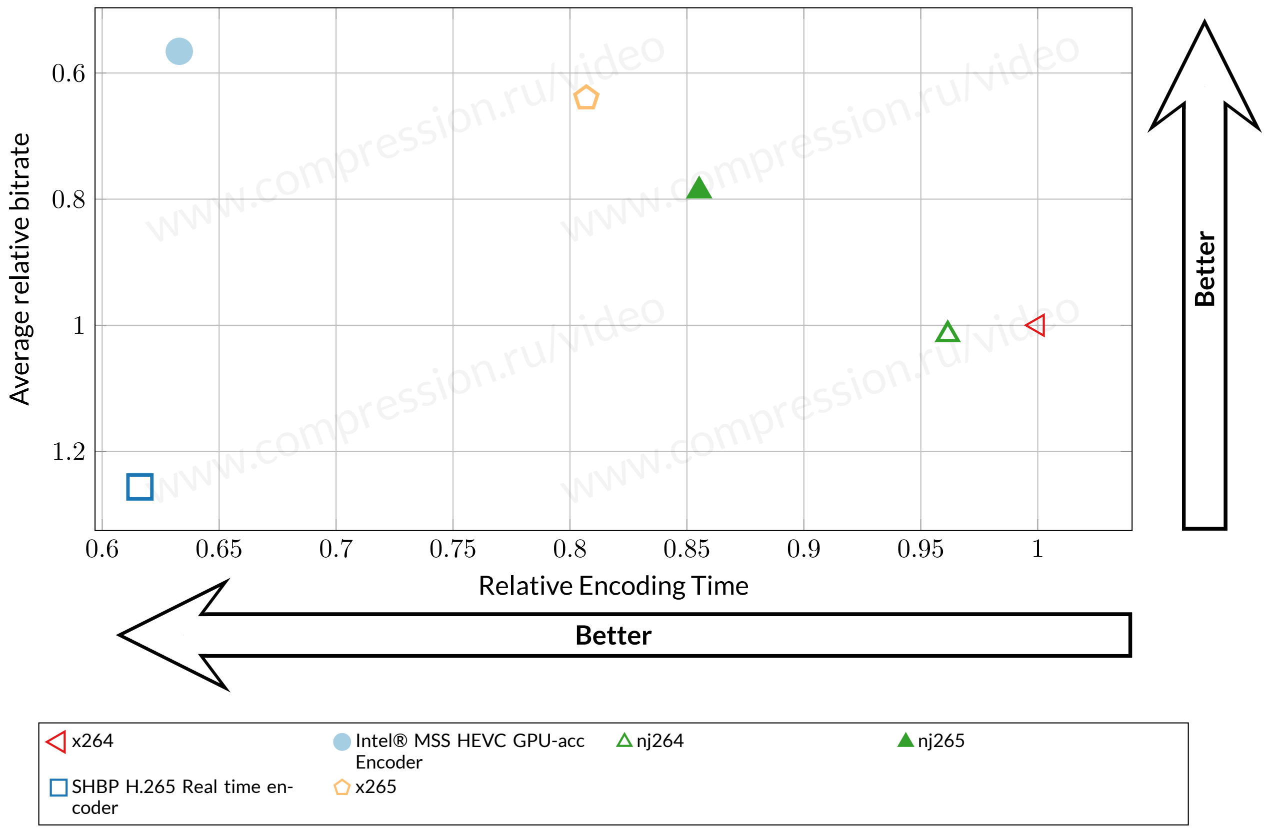


FIGURE 5: Speed/quality trade-off—usecase "4K Encoding," all sequences, Y-SSIM metric

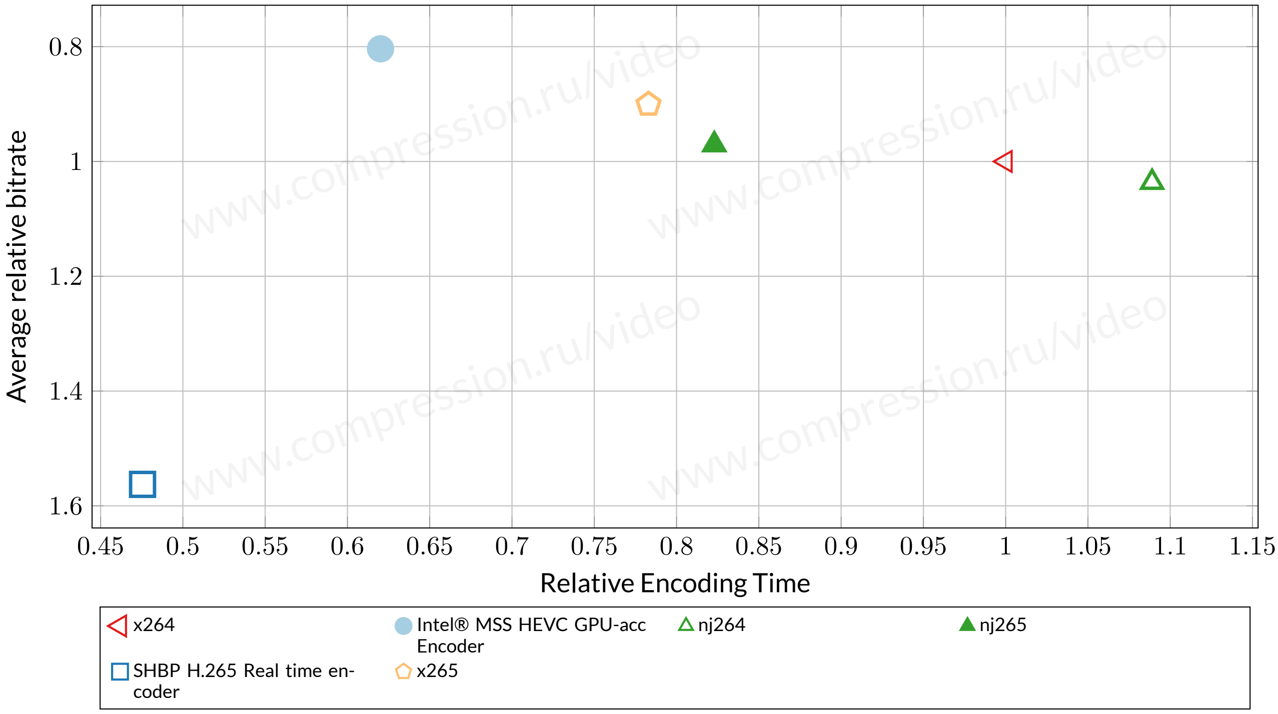


FIGURE 6: Speed/quality trade-off—usecase “4K Encoding,” Driving sequence, Y-SSIM metric

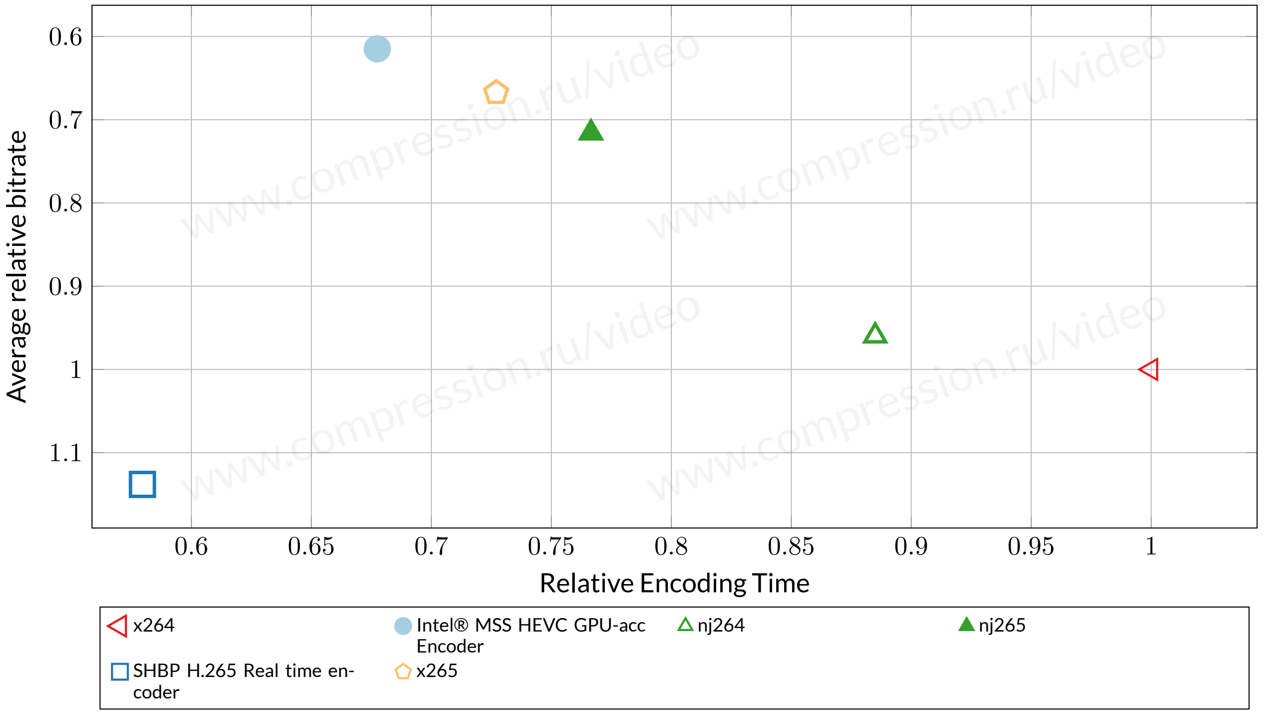


FIGURE 7: Speed/quality trade-off—usecase “4K Encoding,” Dirt Trail sequence, Y-SSIM metric

7 BITRATE HANDLING

The plots below show how accurately encoded stream's real bitrate matches bitrate requested by user. Encoders sometimes fail to correctly handle bitrate on some video sequences. SHBP has problems with bitrate handling (sometimes it decreases target bitrate up to 40%).

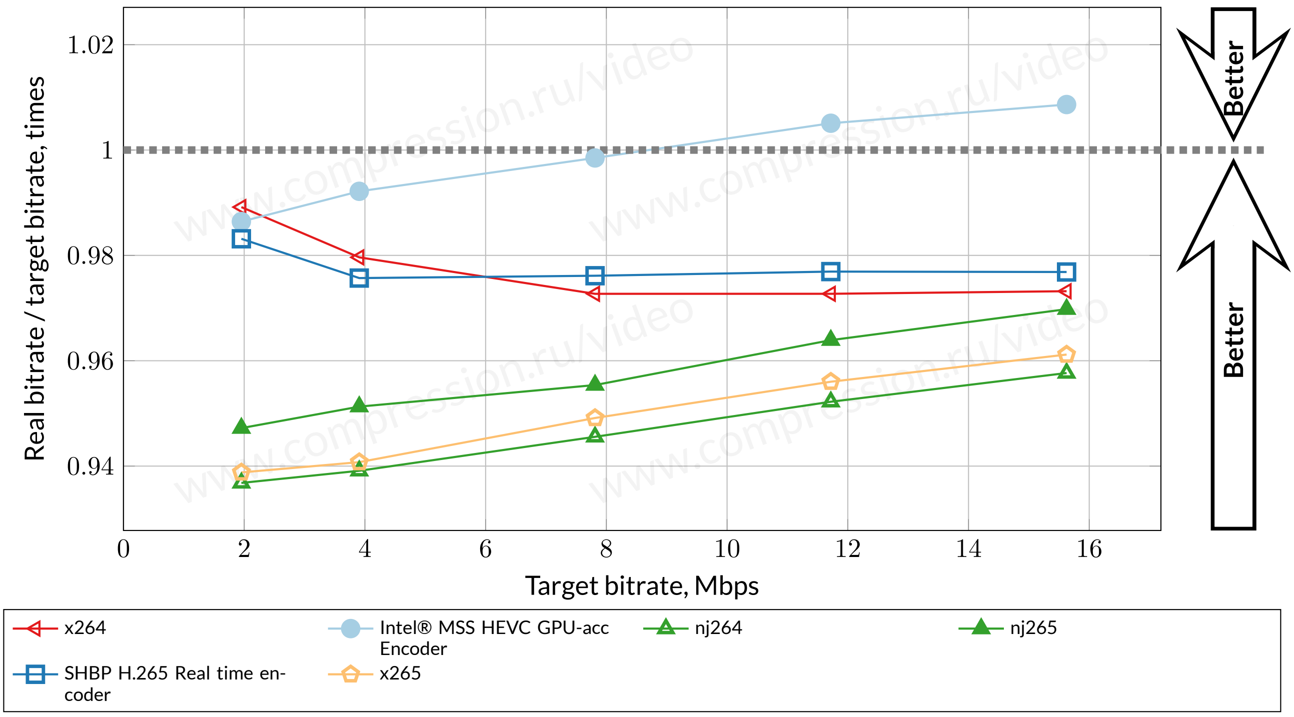


FIGURE 8: Bitrate handling—usecase "4K Encoding," Driving sequence

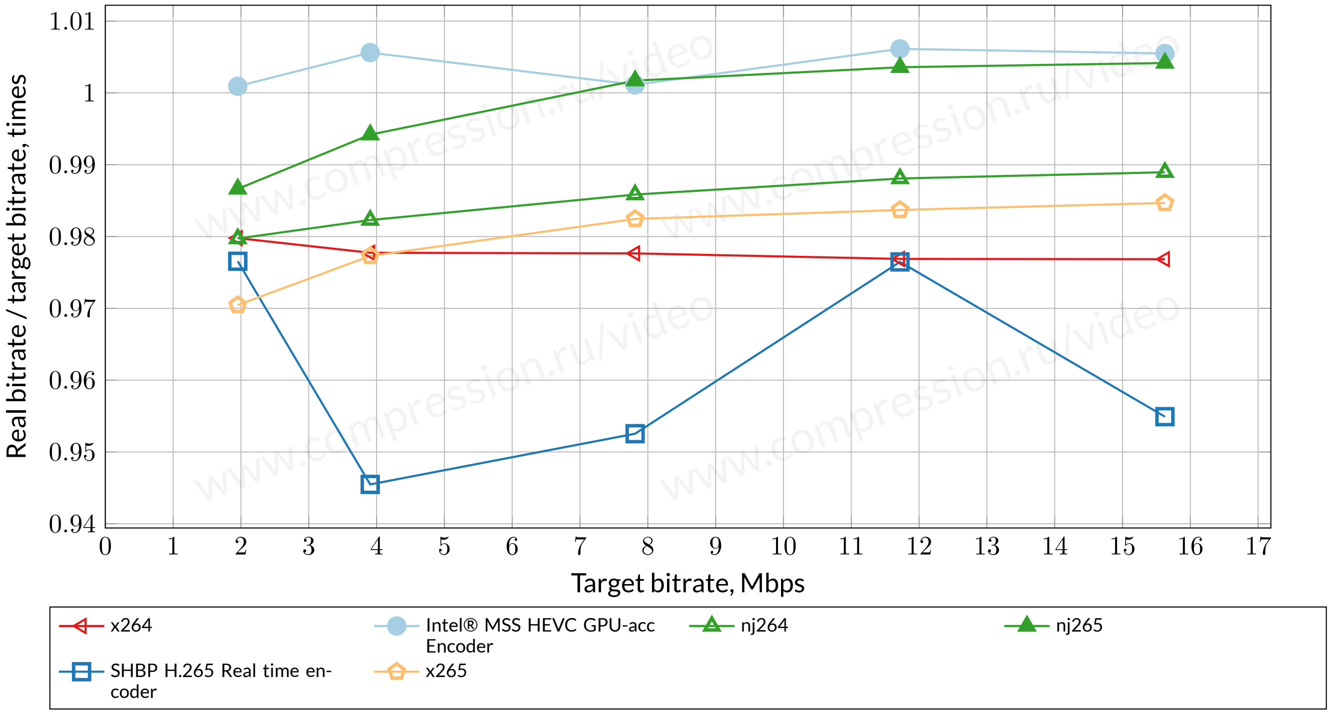


FIGURE 9: Bitrate handling—usecase “4K Encoding,” Dirt Trail sequence

8 RELATIVE QUALITY ANALYSIS

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

	x264	Intel® MSS HEVC GPU-acc Encoder	nj264	nj265	SHBP H.265 Real time encoder	x265
x264	100% ☺	64% ☺	105% ☺	85% ☺	141% ☺	69% ☺
Intel® MSS HEVC GPU-acc Encoder	177% 😐	100% ☺	188% 😐	142% ☺	306% 😐	111% ☺
nj264	99% ☺	60% ☺	100% ☺	80% ☺	137% ☺	66% ☺
nj265	127% ☺	73% ☺	137% ☺	100% ☺	215% ☺	81% ☺
SHBP H.265 Real time encoder	80% ☺	38% 😐	81% ☺	52% ☺	100% ☺	43% ☺
x265	156% ☺	91% ☺	167% ☺	125% ☺	250% 😐	100% ☺

Confidence ☹ 0% — 😐 50% — ☺ 100%

TABLE 3: Average bitrate ratio for a fixed quality—usecase "4K Encoding," Y-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.

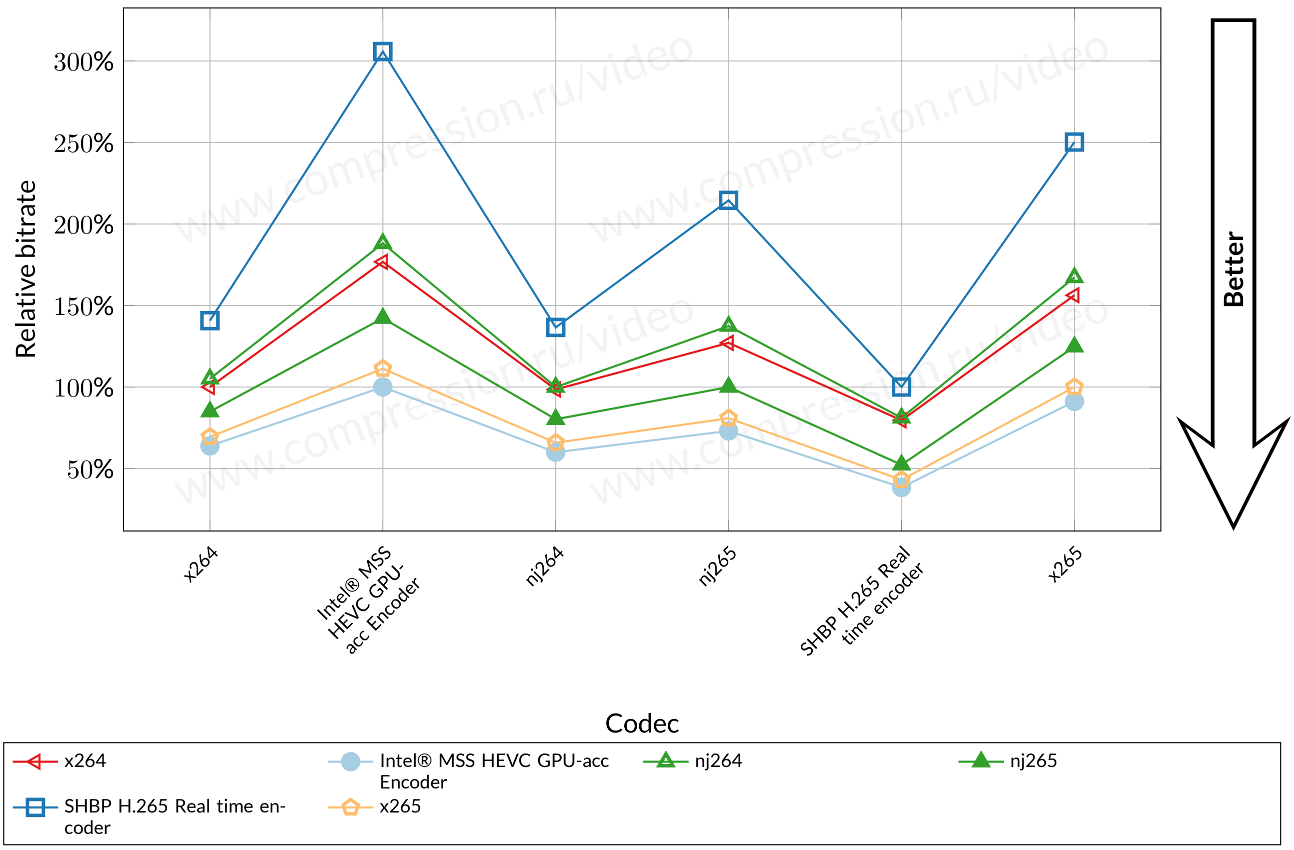


FIGURE 10: Average bitrate ratio for a fixed quality—usecase "4K Encoding," Y-SSIM metric

9 CONCLUSION

All encoders could be ranged by quality in the following way:

- First place is for Intel MSS HEVC GAcc encoder
- Second place is for x265
- Third place is for nj265 encoder.

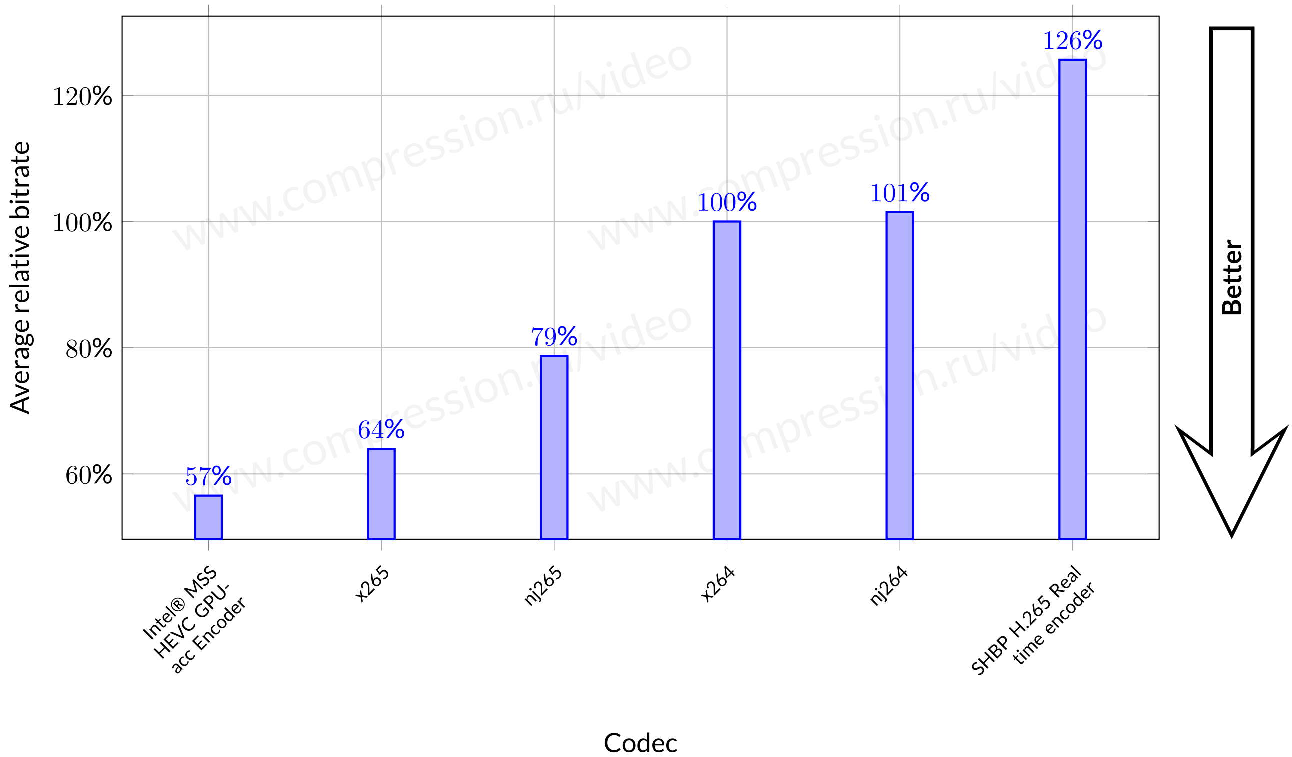


FIGURE 11: Average bitrate ratio for a fixed quality—usecase "4K Encoding," Y-SSIM metric.

10 PARTICIPANTS' COMMENTS

10.1 MulticoreWare, Inc. (x265 developer)

1. Optimal 4K encoding performance requires at least 12 GB of RAM. 16 GB is preferred. x265 can run much faster (and therefore, you can use settings that can achieve higher quality) on a system with more physical memory.

2. It is likely that the encoding speed would also be affected by disk I/O. To fairly evaluate software encoding speed, it is preferable to use a RAMdisk.

3. Haswell generation processors have memory bandwidth limitations that can affect 4K encoding. Depending on your settings, you can achieve more than 2x the encoding speed (and therefore, higher quality at a given speed) with Skylake generation processors.

4. x265 offers psycho-visual optimizations that improve visual quality, but we must turn them off for tests that use SSIM as the scoring metric. The benefits of these algorithms will be seen when measuring visual quality subjectively (with the human eye).

5. Y-SSIM is limited as a quality comparison tool, as it completely ignores the chroma planes (does not measure color accuracy).

A SEQUENCES

A.1 "Canary Wharf"

Sequence title	Canary Wharf
Resolution	4096×2304
Number of frames	1722
Color space	YV12
Frames per second	24

Time-Lapse Of Canary Wharf, London. Static camera. Boats are floating in various directions producing some waves. The end of the sequence was captured at the night time.

FIGURE 12: Canary Wharf sequence, frame 20

A.2 "Coastguard"

Sequence title	Coastguard
Resolution	3840×2160
Number of frames	240
Color space	YV12
Frames per second	24
Source	http://www.elementaltechnologies.com/resources/4k-test-sequences

The coastguard boat rapidly floats forward. The camera is initially static and then starts to follow the boat.

FIGURE 13: Coastguard sequence, frame 20

A.3 "Cobra"

Sequence title	Cobra
Resolution	3840×2160
Number of frames	352
Color space	YV12
Frames per second	30
Source	http://www.harmonicinc.com/resources/videos/4k-video-clip-center

The cobra in front of fence and leaves is turning around. The camera is slightly moving.

FIGURE 14: Cobra sequence, frame 20

A.4 "Driving"

Sequence title	Driving
Resolution	4096×2160
Number of frames	1747
Color space	YV12
Frames per second	24

The camera is set on the car rapidly driving Mulholland Drive in the evening.

FIGURE 15: Driving sequence, frame 20

A.5 "Foreman"

Sequence title	Foreman
Resolution	3840×2160
Number of frames	248
Color space	YV12
Frames per second	24
Source	http://www.elementaltechnologies.com/resources/4k-test-sequences

The foreman in front emotionally tells something and waves his hand. Then camera quickly rotates to the right showing some object under construction.

FIGURE 16: Foreman sequence, frame 20

A.6 "Little Girl"

Sequence title	Little Girl
Resolution	4096×2160
Number of frames	1531
Color space	YV12
Frames per second	30

The little girl is playing toy blocks. The camera slowly moves to the left.

FIGURE 17: Little Girl sequence, frame 20

A.7 "Mobile"

Sequence title	Mobile
Resolution	3840×2160
Number of frames	355
Color space	YV12
Frames per second	24
Source	http://www.elementaltechnologies.com/resources/4k-test-sequences

The toy train is pushing the ball in front of the picture. The camera slowly follows the train in second part of the video.

FIGURE 18: Mobile sequence, frame 20

A.8 "News"

Sequence title	News
Resolution	3840×2160
Number of frames	256
Color space	YV12
Frames per second	24
Source	http://www.elementaltechnologies.com/resources/4k-test-sequences

The man and the woman are reporting news. The large screen behind them shows ballet. The camera is static.

FIGURE 19: News sequence, frame 20

A.9 "Sintel"

Sequence title	Sintel
Resolution	4096×1744
Number of frames	2000
Color space	YV12
Frames per second	24
Source	`http://media.xiph.org/sintel/`

The sequence consists of multiple scenes from Sintel movie created with computer graphics.

FIGURE 20: Sintel sequence, frame 20

A.10 "Susie"

Sequence title	Susie
Resolution	3840×2160
Number of frames	588
Color space	YV12
Frames per second	30
Source	http://www.harmonicinc.com/resources/videos/4k-video-clip-center

The young woman in front of almost uniform static background is talking on the cell phone and smiling. At the end of the sequence she stops talking and looks at the phone's screen. The camera is static.

FIGURE 21: Susie sequence, frame 20

A.11 “Dirt Trail”

Sequence title	Dirt Trail
Resolution	3840×2160
Number of frames	3426
Color space	YV12
Frames per second	24

The camera slowly flies through dirt trail in the forest.

FIGURE 22: Dirt Trail sequence, frame 20

B CODECS

B.1 Intel® Media Server Studio HEVC GPU-accelerated Encoder

Encoder title	Intel® Media Server Studio HEVC GPU-accelerated Encoder
Version	Intel® Media Server Studio 2015 R7 – Professional Edition
Developed by	Intel

The following parameters were used to run encoder:

```
mfx_transcoder.exe h265 -encode_plugin mfxplugin64_hevce_hw.dll -hw -sys -i
%SOURCE_FILE% -w %WIDTH% -h %HEIGHT% -f %FPS% -o %TARGET_FILE% -b %BITRATE_BPS%
-avbr -u 7 -async 3
```

B.2 SHBP H.265 Real time encoder

Encoder title	SHBP H.265 Real time encoder
Version	0.8
Developed by	SHBP Codec's development team

```
Usage: sh_hevc_enc.exe <options>

  -help        display this information
  -i <s>       s - input YUV filename
  -w <n>       n - input frames width
  -h <n>       n - input frames height
  -f <f>       f - frames per second value (25.0)
  -n <n>       n - number of frames to encode(0 - all)
  -o <s>       s - output binary filename
  -r <s>       s - reconstructed YUV filename(none)
  -c <s>       s - config txt filename with advanced parameters(none)
  -id <n>      n - input device id(0) (0 - file, 1 - hw emu)
  -od <n>      n - output device id(0) (0 - file, 1 - hw emu)
  -b <n>       n - target bitrate in kb per second
  -g <n>       n - GOP size in frames (104)
  -q <n>       n - quantization parameter [1, 51] (disabled -b option)
  -p <f>       f - performance level in fps (0 - auto)
```

FIGURE 23: SHBP H.265 Real time encoder

The following parameters were used to run encoder:
`sh_hevc_enc.exe -w %WIDTH% -h %HEIGHT% -f %FPS% -n %FRAMES_NUM% -p 10 -b %BITRATE_KBPS% -i %SOURCE_FILE% -o %TARGET_FILE%`

B.3 x264

Encoder title	x264
Version	146 r2538 121396c
Developed by	x264 Developer Team

```
x264 core:148 r2638 7599210
Syntax: x264 [options] -o outfile infile

Infile can be raw (in which case resolution is required),
  or YUV4MPEG (*.y4m),
  or Avisynth if compiled with support (yes).
  or libav* formats if compiled with lavf support (yes) or ffms support (no).
Outfile type is selected by filename:
 .264 -> Raw bytestream
 .mkv -> Matroska
 .flv -> Flash Video
 .mp4 -> MP4 if compiled with GPAC or L-SMASH support (no)
Output bit depth: 8 (configured at compile time)

Options:

  -h, --help                  List basic options
      --longhelp              List more options
      --fullhelp              List all options
```

FIGURE 24: x264 encoder

The following parameters were used to run encoder:
x264 --tune ssim --preset medium --me umh --merange 32 --trellis 2 --partitions all --keyint infinite --pass 1 --bitrate %BITRATE_KBPS% %SOURCE_FILE% -o %TARGET_FILE% --input-res %WIDTH%x%HEIGHT% --fps %FPS%
x264 --tune ssim --preset medium --me umh --merange 32 --trellis 2 --partitions all --keyint infinite --pass 2 --bitrate %BITRATE_KBPS% %SOURCE_FILE% -o %TARGET_FILE% --input-res %WIDTH%x%HEIGHT% --fps %FPS%

B.4 x265

Encoder title	x265
Version	1.5+460-ac85c775620f
Developed by	x265 Developer Team

```
Syntax: x265 [options] infile [-o] outfile
    infile can be YUV or Y4M
    outfile is raw HEVC bitstream

Executable Options:
-h/--help                        Show this help text and exit
-V/--version                     Show version info and exit

Output Options:
-o/--output <filename>           Bitstream output file name
-D/--output-depth 8|10|12        Output bit depth (also internal bit depth)
```

FIGURE 25: x265 encoder

The following parameters were used to run encoder:

```
x265 -p faster --tune ssim --cutree --ref 2 --limit-refs 3 -F2 --no-weightp
--min-cu-size 16 --bitrate %BITRATE_KBPS% %SOURCE_FILE% -o %TARGET_FILE%
--input-res %WIDTH%x%HEIGHT% --fps %FPS%
```

B.5 nj264

Encoder title	nj264
Version	1.0
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.

The following parameters were used to run encoder:

```
nj264.exe -s %WIDTH%x%HEIGHT% -framerate %FPS% -i %SOURCE_FILE% -c:v libnj264
-preset quality -nj264-params bitrate=%BITRATE_KBPS%:keyint=40 -f h264 -y
%TARGET_FILE%
```

B.6 nj265

Encoder title	nj265
Version	1.0
Developed by	Nanjing Yunyan

The following parameters were used to run encoder:

```
nj265.exe -s %WIDTH%x%HEIGHT% -framerate %FPS% -i %SOURCE_FILE% -c:v libnj265
-preset speed -nj265-params bitrate=%BITRATE_KBPS%:keyint=40 -f hevc -y
%TARGET_FILE%
```

C 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.

C.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.

C.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.

C.3 Graph Example

Figure 26 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.

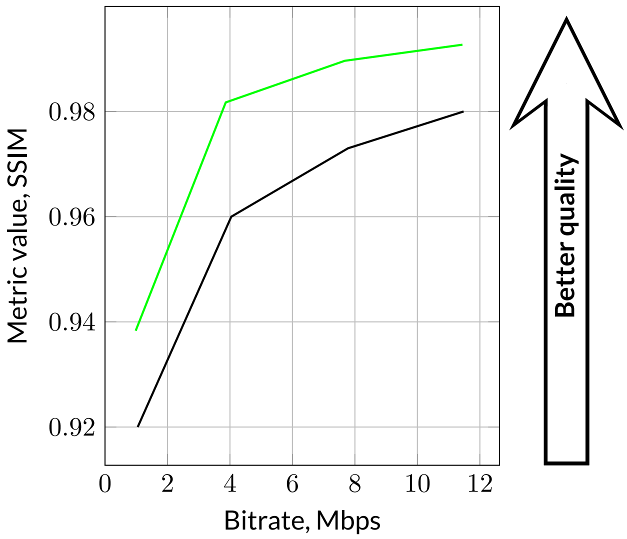
A more detailed description of the preparation of these graphs is given below.

C.4 Bitrates Ratio with the Same Quality

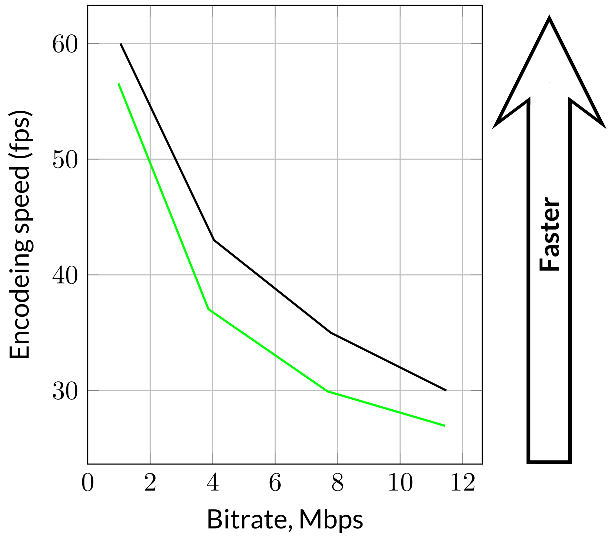
The first step in computing the average bitrate ratio for a fixed quality is inversion of the axes of the bitrate/quality graph (see Figure 27b). All further computations are performed using the inverted graph.

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.

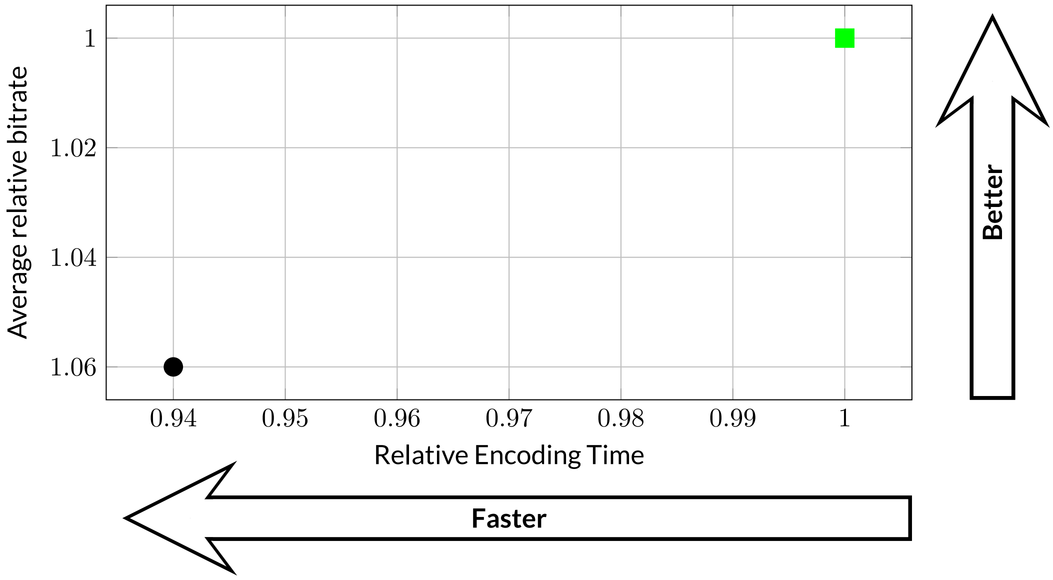
The final step is calculation of the area under the curves in the chosen interpolation segment and determination of their ratio (see Figure 27c). This result is an average bitrate ratio for a fixed quality for the two codecs. If



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

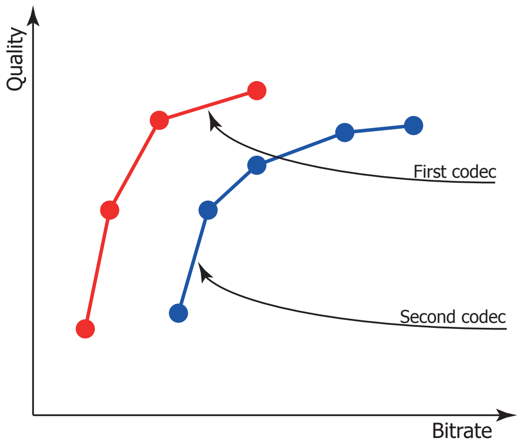


(b) Encoding speed (frames per second). "Green" codec is slower!

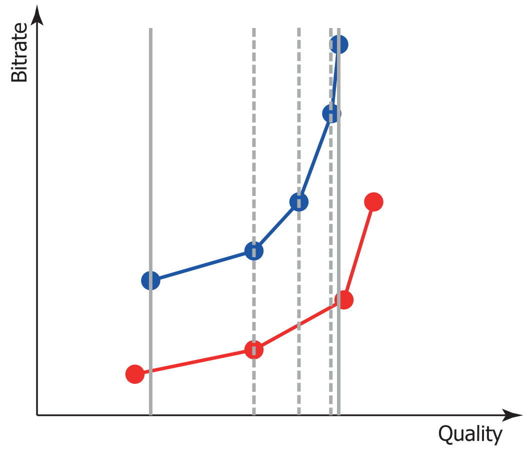


(c) Integral situation with codecs. This plot shows the situation more clearly

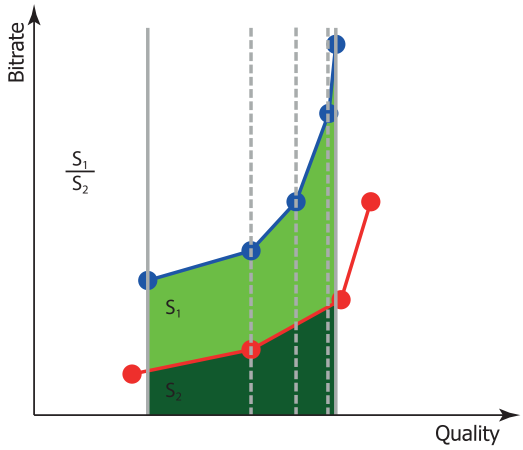
FIGURE 26: Speed/Quality trade-off example



(a) Source RD curves



(b) Axes' inversion and averaging interval choosing



(c) Areas under curves ratio

FIGURE 27: 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.

C.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 C.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 4: 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 27) 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 these tables. Each line in such plot depicts values from one column of corresponding table.

D OBJECTIVE QUALITY METRICS DESCRIPTION

D.1 SSIM (Structural SIMilarity)

D.1.1 Brief Description

The original paper on the SSIM metric was published by Wang, et al.[1] 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:

$$\textbf{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x + \mu_y + C_1)(\sigma_x + \sigma_y + C_2)}, \tag{1}$$

where

$$\mu_x = \sum_{i=1}^{N} \omega_i x_i, \tag{2}$$

$$\sigma_x = \sqrt{\sum_{i=1}^{N} \omega_i (x_i - \mu_x)}, \tag{3}$$

$$\sigma_{xy} = \sum_{i=1}^{N} \omega_i (x_i - \mu_x)(y_i - \mu_y). \tag{4}$$

Finally, $C_1 = (K_1L)^2$ and $C_2 = (K_2L)^2$, 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 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.

D.1.2 Examples

Figure 28 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.

[1] 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.

(a) Original

(b) Compressed

(c) SSIM

FIGURE 28: SSIM example for compressed image

Figure 29 depicts various distortions applied to original image and Figure 30 shows SSIM values for these distortions.

(a) Original image

(b) Image with added noise

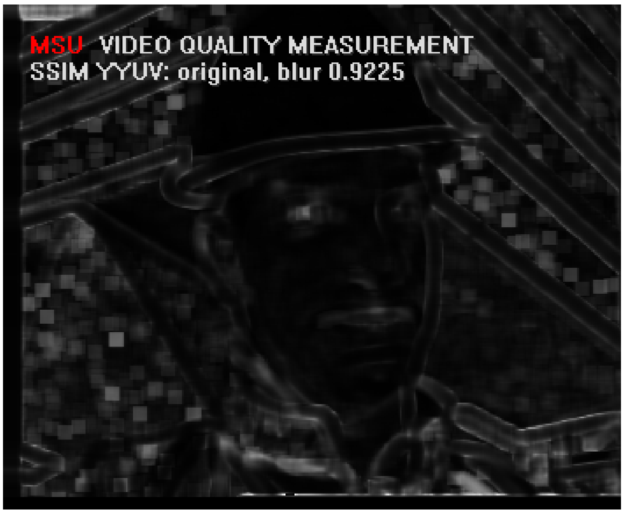
(c) Blurred image

(d) Sharpen image

FIGURE 29: 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 30: SSIM values for original and processed images

E 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