MPEG-4 AVC/H.264 Video Codecs Comparison

Video group head: Dr. Dmitriy Vatolin
Project head: Dr. Dmitriy Kulikov
Measurements, analysis: Alexander Parshin, Marat Arsaev

Report correction: Alexander Voronov

Codecs:

- **H.264**
    - DivX H.264
    - Elecard H.264
    - Intel SandyBridge Transcoder (GPU encoder)
    - MainConcept H.264 (software)
    - MainConcept H.264 (CUDA based encoder)
    - Microsoft Expression Encoder
    - DiscretePhoton
    - x264
- **Non H.264**
    - VP8 (WebM project)
    - XviD (MPEG-4 ASP codec)

May 2011

CS MSU Graphics&Media Lab, Video Group

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:

- Elecard Ltd
- Intel Corporation
- MainConcept GmbH
- x264 Development Team
- WebM project team
- DiscretePhoton team
- Microsoft Expression Encoder

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

2 Overview

2.1 Sequences

Table 1. Summary of video sequences.

Sequence	Number of frames	Frame rate	Resolution
VideoConference			
1. Videoconference CIF	1374	30	352x288
2. VideoConference 4CIF	3600	30	640x480
3. VideoConference 720p	1500	30	1280x720
Movies (SD sequences)			
4. Ice Age	2014	24	720x480
5. City	600	60	704x576
6. Crew	600	60	704x576
7. Indiana Jones	5000	30	704x288
8. Harbour	600	60	704x576
9. Ice Skating	480	60	704x576
10. Soccer	600	60	704x576
11. Race Horses	300	30	832x480
12. State Enemy	6500	24	720x304
13. Party Scene	500	50	832x480
HDTV sequences			
14. Park Joy	500	50	1280x720
15. Riverbed	250	25	1920x1080
16. Rush Hour	500	25	1920x1080
17. Blue Sky	217	25	1920x1080
18. Station	313	25	1920x1080
19. Stockholm	604	50	1280x720
20. Sunflower	500	25	1920x1080
21. Tractor	690	25	1920x1080
22. Bunny	600	24	1920x1080
23. Dream	600	24	1920x1080
24. Troy	300	24	1920x1072

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 5. Test Set of Video Sequences.

2.2 Codecs

Table 2. Short codec descriptions

	Codec	Developer	Version
1.	DivX AVC/H.264 Video Encoder	DivX, Inc.	1.1.1.9
2.	Elecard AVC Video Encoder 8-bit edition,	Elecard Ltd	2.1.022202.091207
3.	MainConcept AVC/H.264 Video Encoder Console Application	MainConcept GmbH	1.5.0
4.	Microsoft Expression Encoder 4	Microsoft Corp.	encoder_core.dll version 4.0.3205.0 mc_enc_avc.dll version 8.7.0.37256
5.	x264	x264 Development Team	x264 core:114 r1900 60ef1f8
6.	XviD raw mpeg4 bitstream encoder	XviD Development Team	xvid-1.3.0-dev
7.	Discrete Photon	Discrete Photon Development Team	unversioned
8.	WebM vp8 Codec	WebM	v0.9.2-522-gddd260e

Brief descriptions of the codecs used in our comparison are given in Table 2. XviD was used as a good quality MPEG-4 ASP reference codec for comparison purposes. Detailed descriptions of all codecs used in our comparison can be found in Appendix 6. Tested Codecs.

3 Objectives and Testing Rules

3.1 H.264 Codec Testing Objectives

The main goal of this report is the presentation of a comparative evaluation of the quality of new H.264 codecs using objective measures of assessment. The comparison was done using settings provided by the developers of each codec.

The main task of the comparison is to analyze different H.264 encoders for the task of transcoding video—e.g., compressing video for personal use. Speed requirements are given for a sufficiently fast PC; fast presets are analogous to real-time encoding for a typical home-use PC.

3.2 Testing Rules

- The entire test set was divided into two primary types of applications. These applications differ by resolution, bitrate and encoding speed requirements:
    - VideoConference (one pass only)
    - Movies (bitrates of 500-2000 kbps)
    - High-definition television (“HDTV”; bitrates of 0.7-10 mbps)
- There are special presets and speed limitations for every type of application:
    - Video Conference (one pass only, good local bitrate handling)
        - Minimum 60 fps at 4CIF sequence
    - Movies (speed requirements for 750 kbps 4CIF sequences):
        - Minimum 120 fps for "High Speed" preset
        - Minimum 80 fps for "Normal" preset Minimum 40 fps for "High Quality" preset
    - HDTV (speed requirements for 3 mbps 1280x720 sequences):
        - Minimum 100 fps for "High Speed" preset
        - Minimum 50 fps for "Normal" preset
        - Minimum 20 fps for "High Quality" preset
- Each codec’s developer provided settings for each type of application. Each setting’s individual parameters were, to a large extent, chosen by the developers, except the following:
    - DivX H.264
    - XviD (last year presets were used)
- Each codec was tested for speed three times; the minimum score was then used as the representative time.
- During the testing process, source video sequences were in the YV12 format (.yuv file extension) for all codecs.

- For all measurements the PRO version of the YUVsoft Video Codec Scoring System was used (http://www.yuvsoft.com/technologies/vicos/index.html).
- The following computer configuration was used for the main tests:
    - 4-cores processor: Intel Core i7 920, 2.67GHz
    - OS Name: Microsoft Windows 7 Professional 64-bit
    - Total Physical Memory: 12 GB

During the evaluation the following measures were used:

- SSIM (Y component)
- PSNR (Y component)

Enterprise version of report contains:

- SSIM, Y-SSIM, U-SSIM, V-SSIM
- PSNR, Y-PSNR, U-PSNR, V-PSNR
- MSE
- 3-SSIM
- MS-SSIM

More detailed information about these measures may be found on the Internet at the following URL:

http://www.compression.ru/video/quality_measure/info.html

4 Comparison Results

4.1 Video Conferences

4.1.1 RD Curves

Next figures show RD curves for three video conference sequences. The leader by quality is x264. DivX H.264 is typically second.

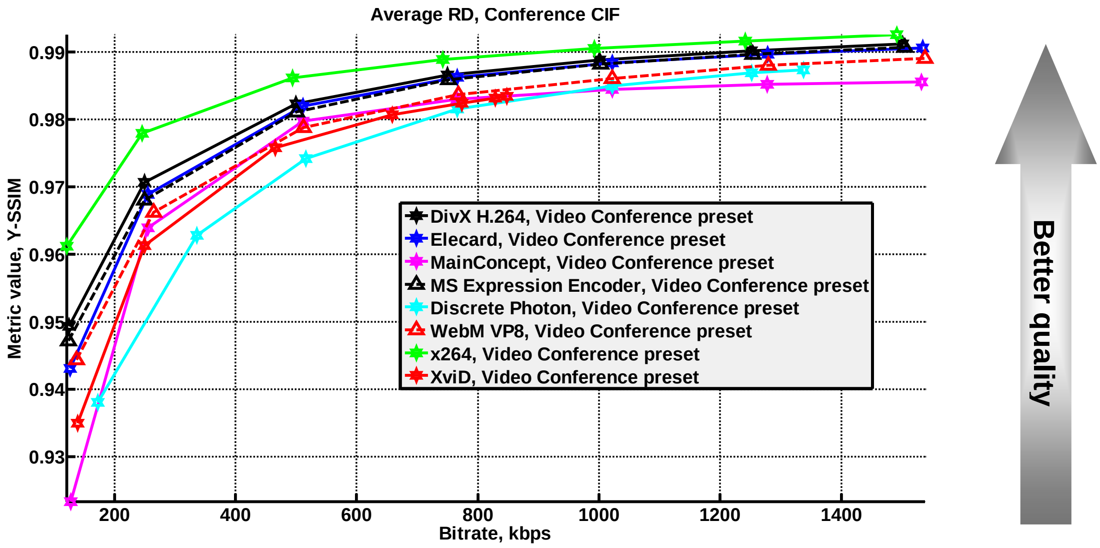
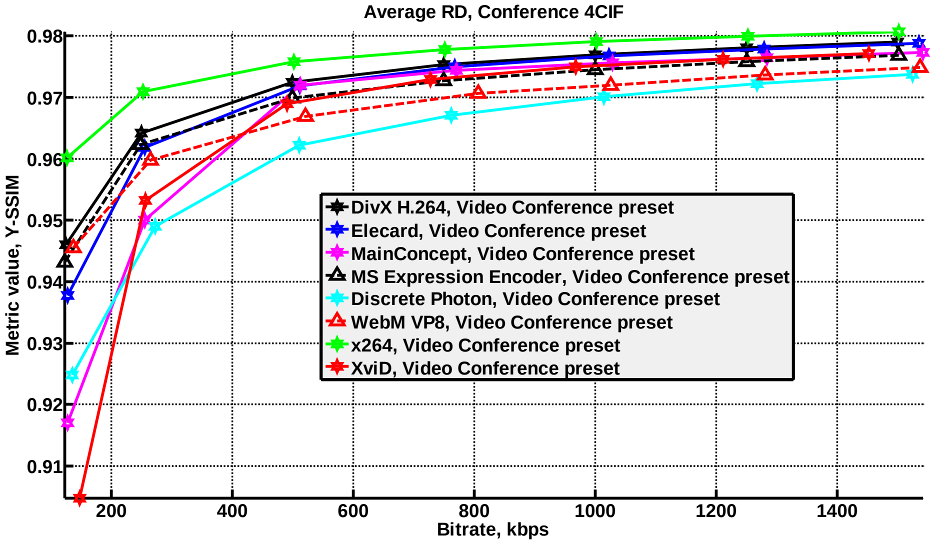


Figure 1. Bitrate/quality—usage area “Video Conference,” CIF sequence, Y-SSIM metric



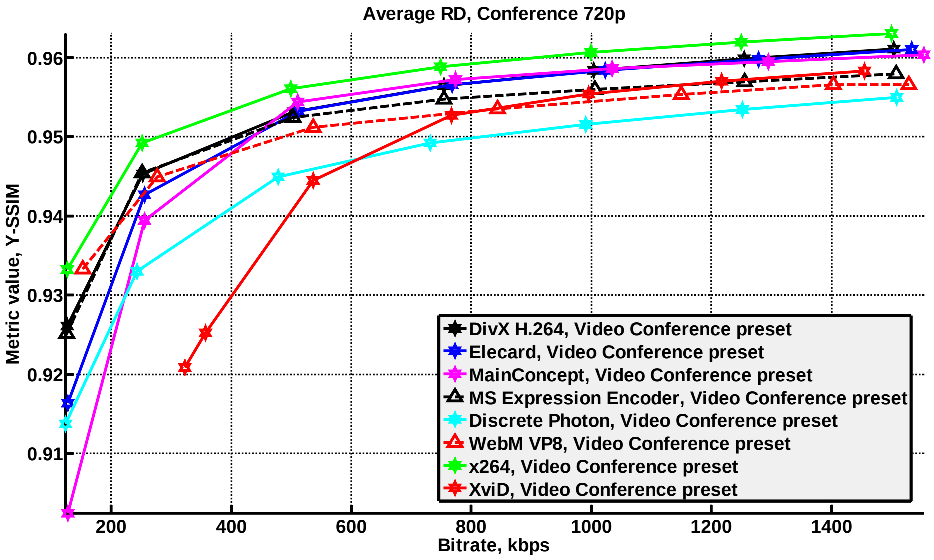


Figure 2. Bitrate/quality—usage area "Video Conference," 720p sequence, Y-SSIM metric

4.1.2 Encoding Speed

Absolute speed results are presented in Figure 3 through Figure 5. All the encoders have a similar growth rate for encoding time as the bitrate is increased. Discrete Photon is the fastest.

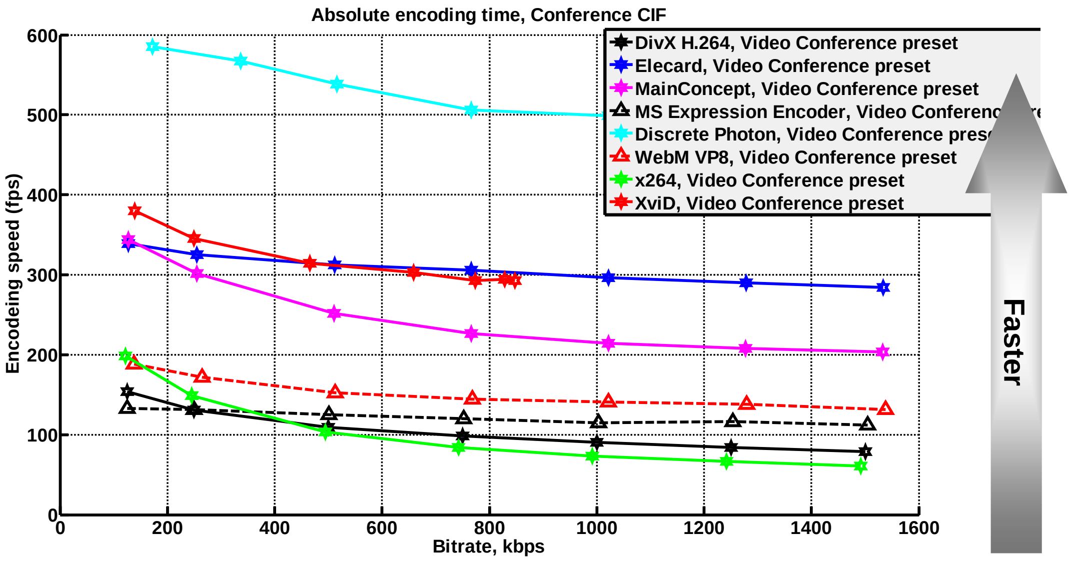


Figure 3. Encoding speed—usage area "Video Conference" CIF sequence

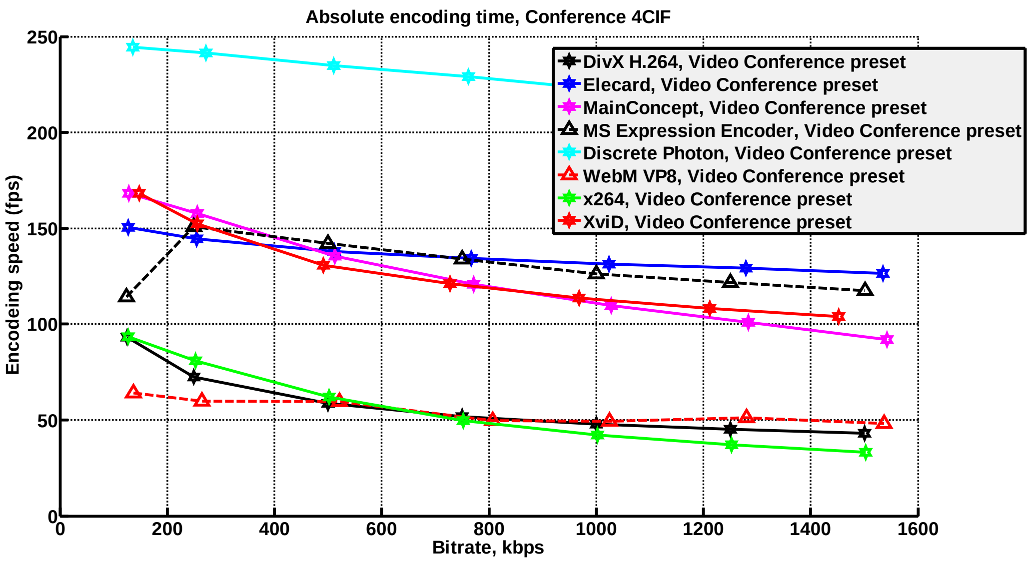


Figure 4. Encoding speed—usage area "Video Conference" 4CIF sequence

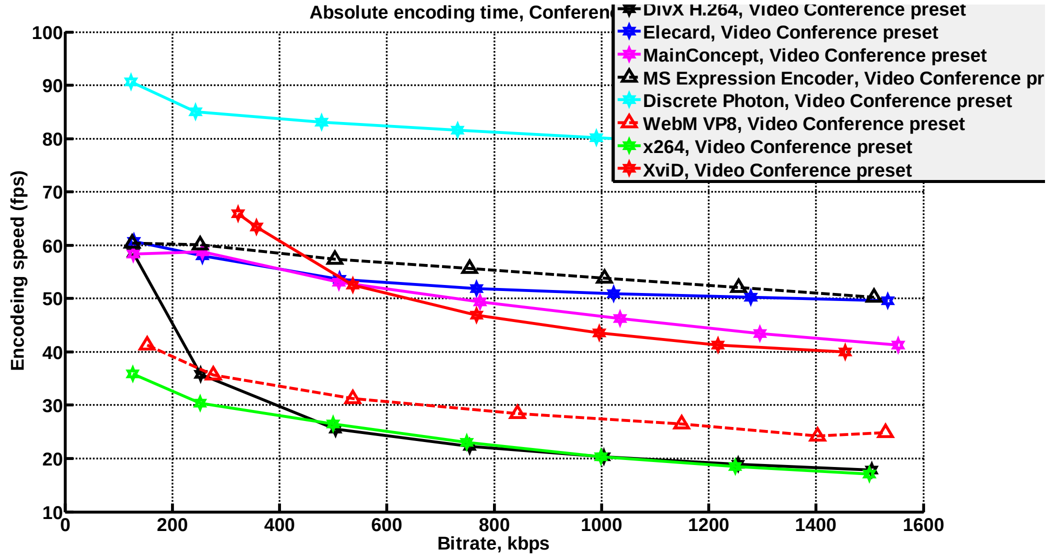


Figure 5. Encoding speed—usage area "Video Conference" 720p sequence

4.1.3 Speed/Quality Trade-Off

Detailed descriptions of the speed/quality trade-off graphs can be found in Appendix 7. Figures Explanation. 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. XviD 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.

Please note that the method of averaging among all sequences assumes that all codecs produced results for each sequence. When this is not the case, only existing results are taken into account.

The three best codecs (no codec performs faster with higher quality) in terms of speed/quality are DiscretePhoton, Elecard and x264 at average.

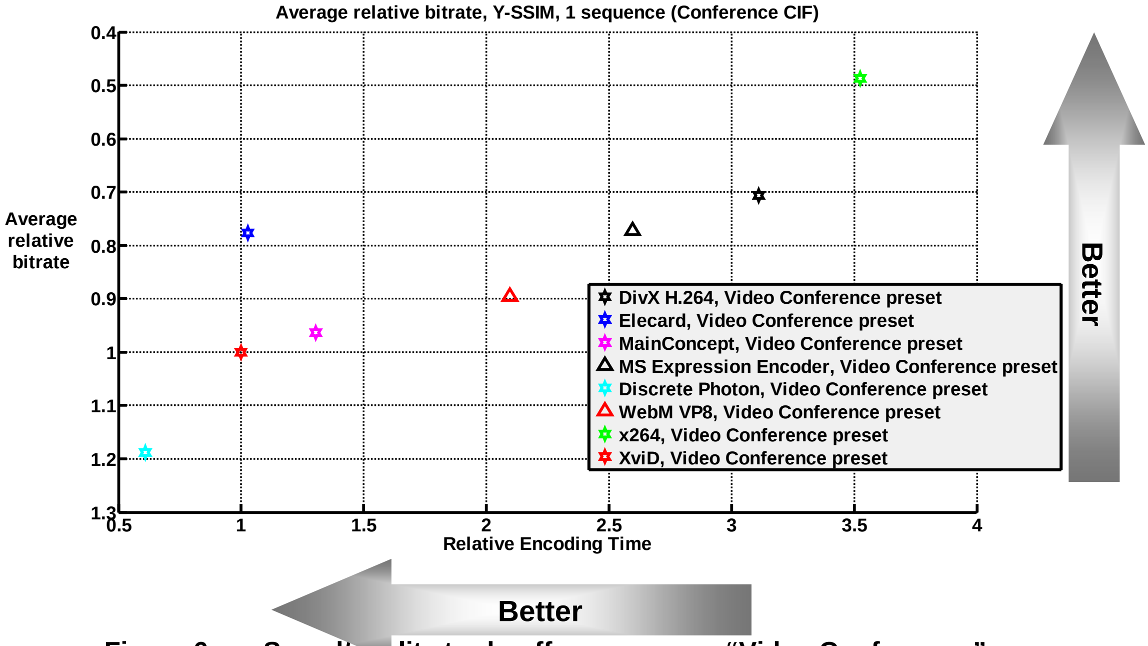


Figure 6. Speed/quality trade-off—usage area "Video Conference," CIF sequence, Y-SSIM metric

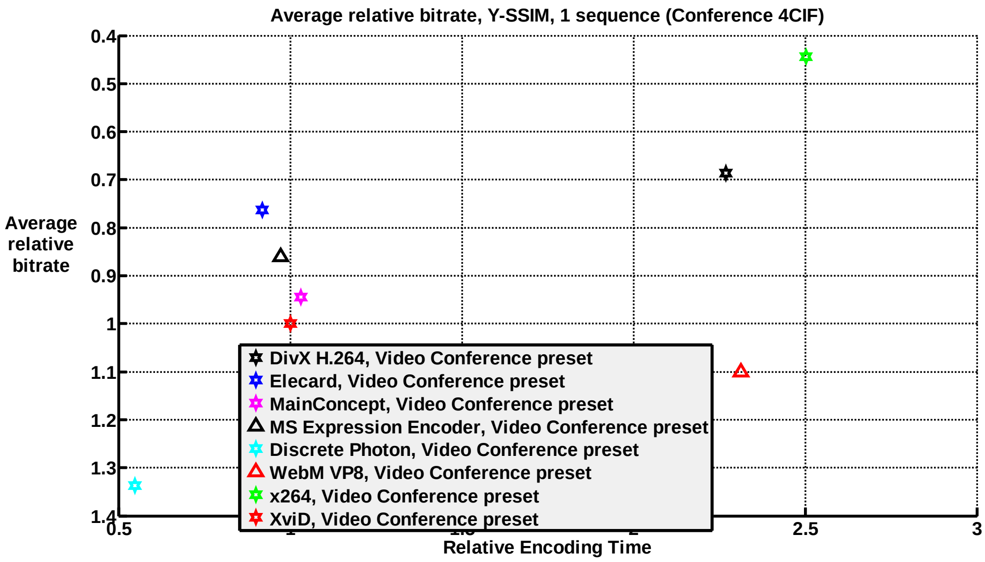


Figure 7. Speed/quality trade-off—usage area "Video Conference," 4CIF sequence, Y-SSIM metric

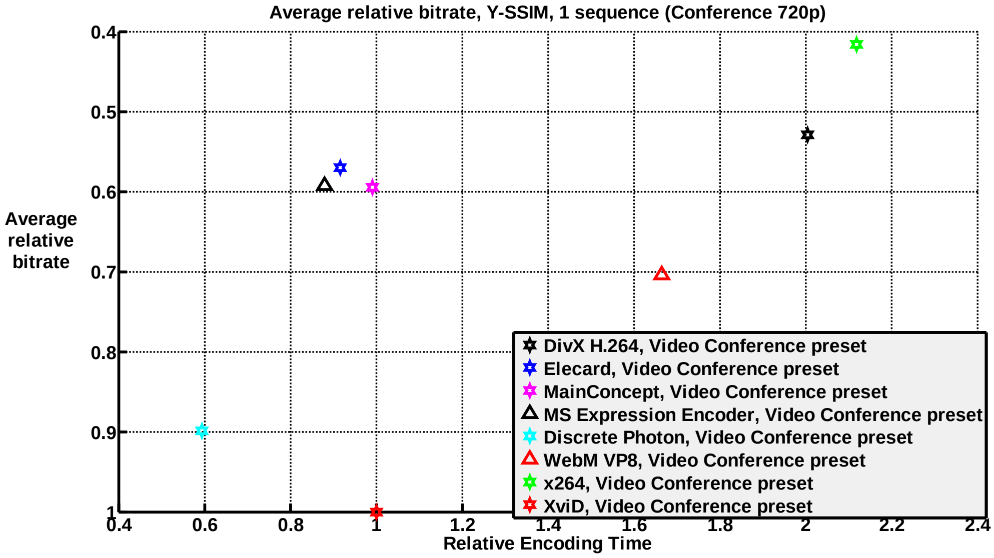


Figure 8. Speed/quality trade-off—usage area "Video Conference," 720p sequence, Y-SSIM metric

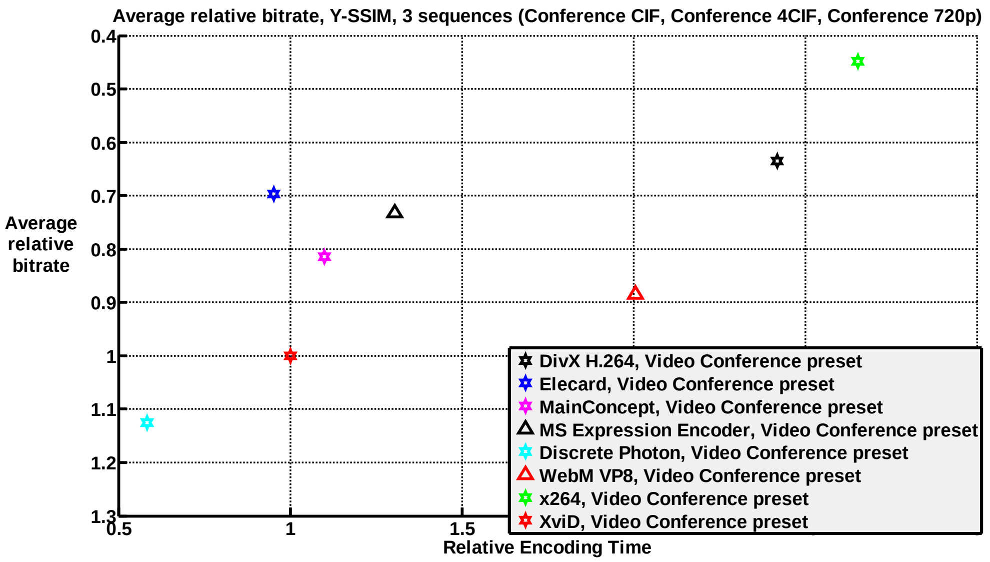


Figure 9. Speed/quality trade-off—usage area "Video Conference," all sequences, Y-SSIM metric

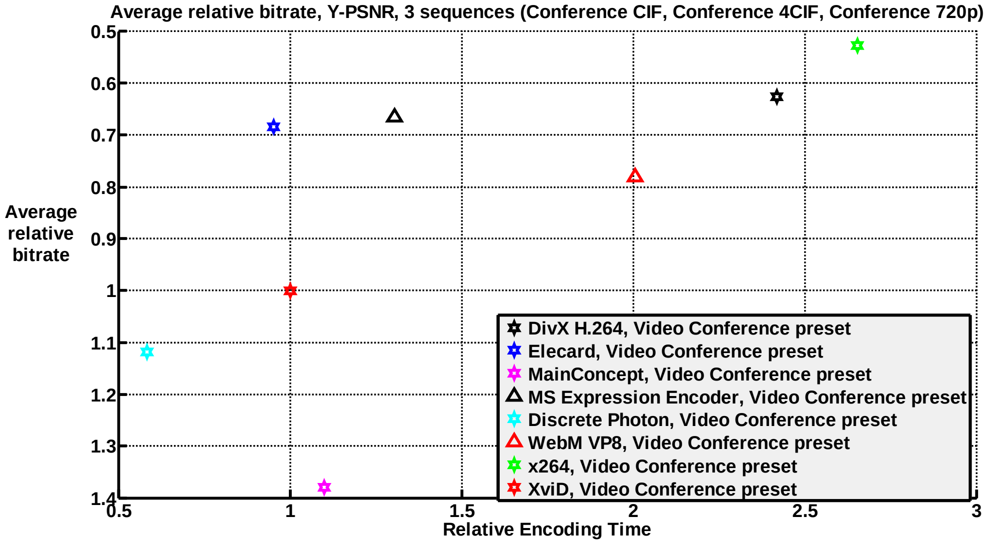


Figure 10. Speed/quality trade-off—usage area "Video Conference," all sequences, Y-PSNR metric

4.1.4 Bitrate Handling

Encoders with High Speed presets, except the XviD encoder, demonstrate good bitrate handling for all sequences. There are some issues with bitrate handling for DiscretePhoton encoders for CIF sequence.

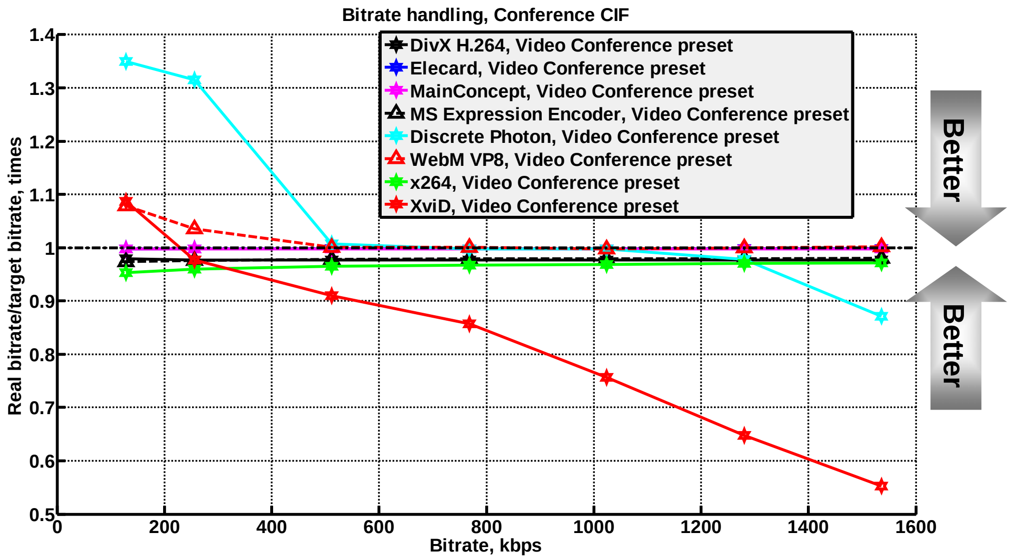


Figure 11. Bitrate handling—usage area "Video Conference," CIF sequence

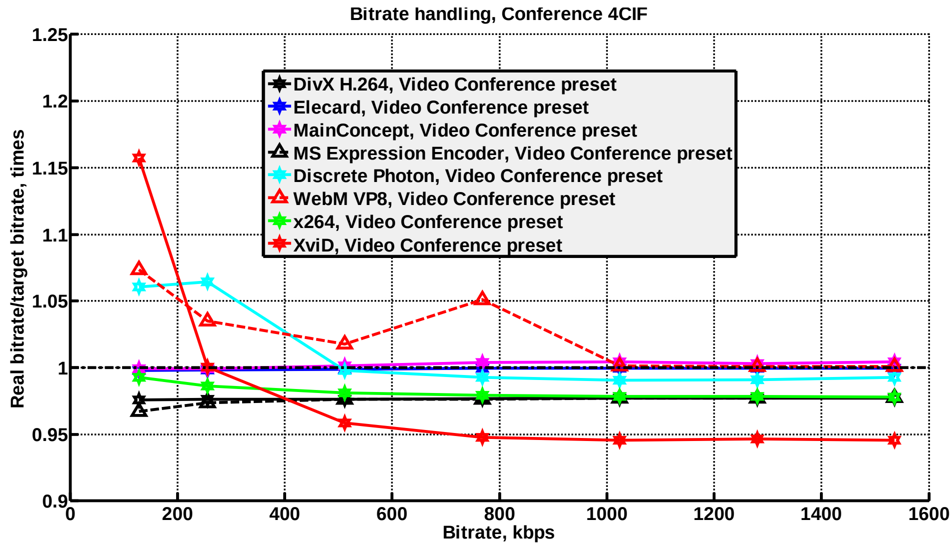


Figure 12. Bitrate handling—usage area “Video Conference,” 4CIF sequence

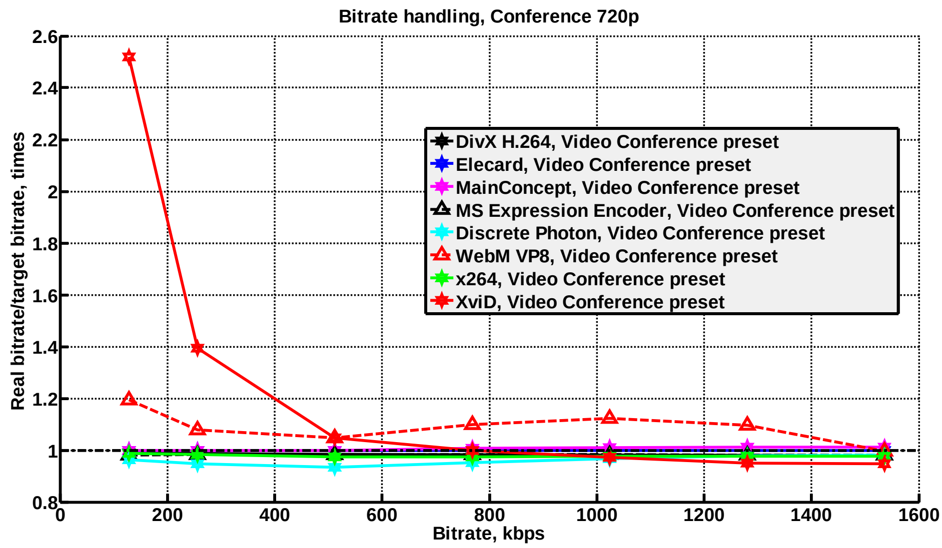


Figure 13. Bitrate handling—usage area “Video Conference,” 720p sequence

4.1.5 Local Bitrate Handling

For video conference encoding is very important not only keep average bitrate for all the sequence but keep local bitrate for example for 1 second window. In this part of comparison we analyze local bitrate handling by next formula

$$LBH = \frac{\max(mfps) \cdot 8 \cdot fps}{1024 \cdot \text{target\_bitrate}},$$

where mfps – average frame size in 1 sec (fps) window

And overall results is maximal value of LBH for sequence.

Elecard shows best result for this analysis. And x264 shows lowest result.

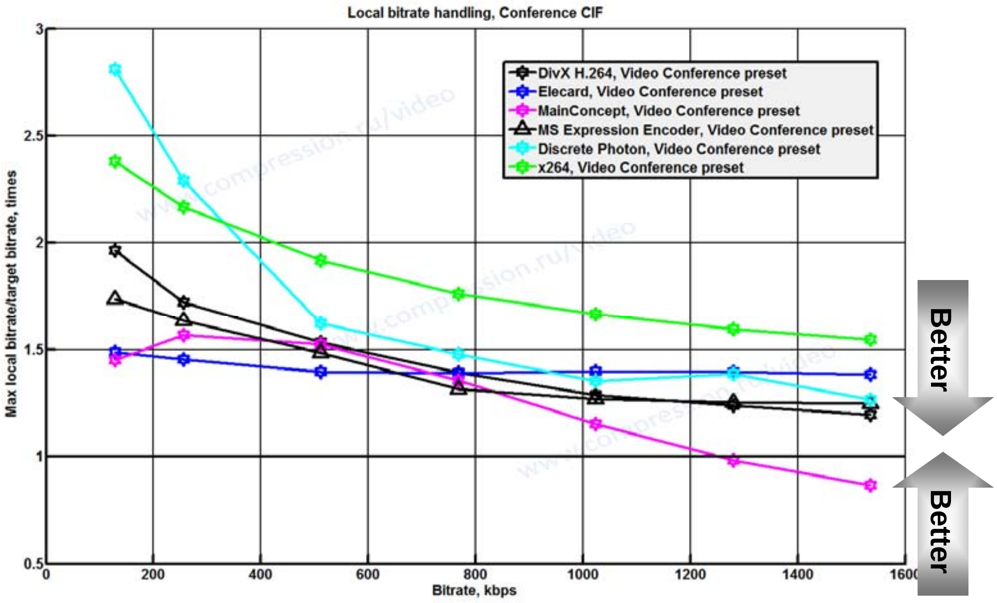


Figure 14. Bitrate handling—usage area "Video Conference," CIF sequence

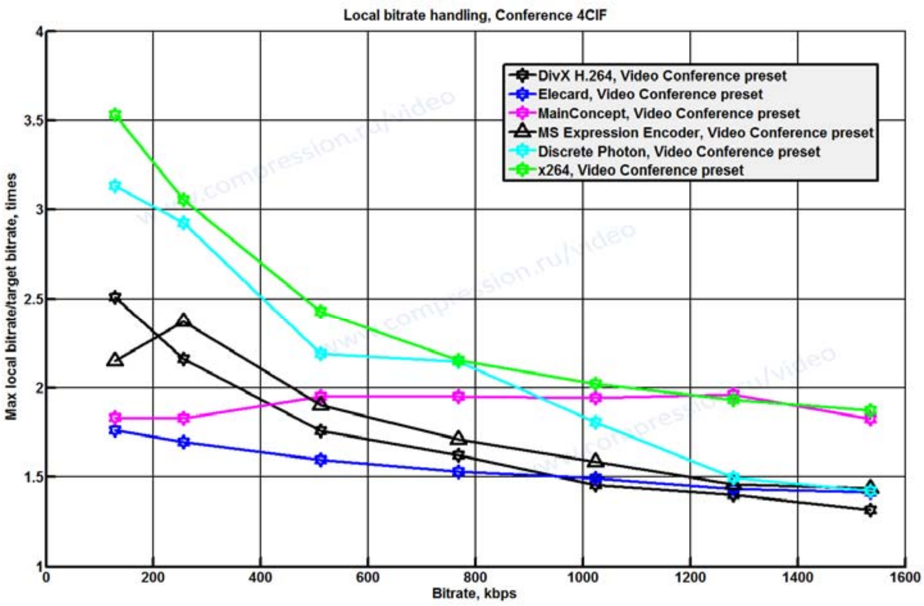


Figure 15. Bitrate handling—usage area "Video Conference," 4CIF sequence

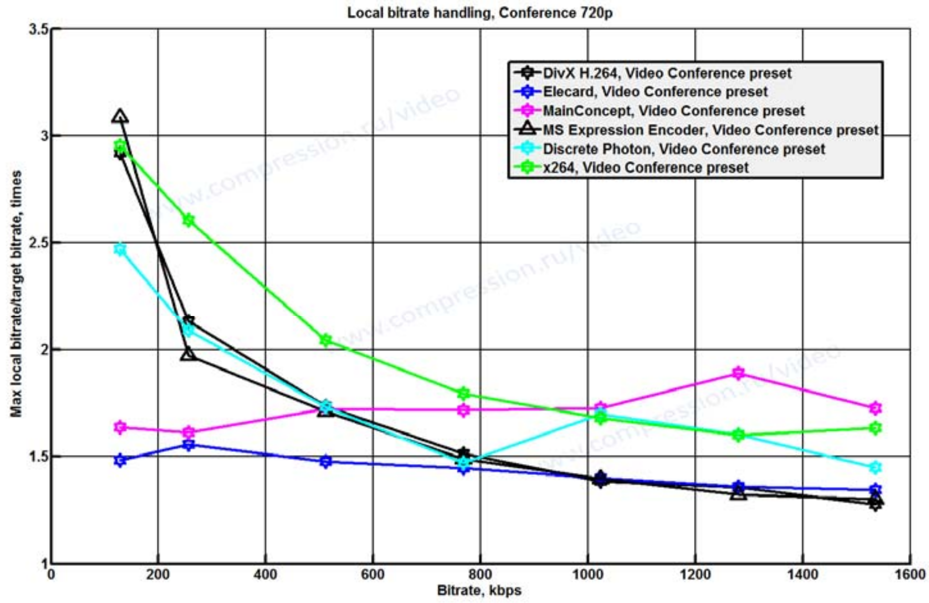


Figure 16. Bitrate handling—usage area "Video Conference," 720p sequence

4.1.6 Relative Quality Analysis

Table 3 and Table 4 show relative bitrates for a fixed quality output for all codecs and presets. Note that these tables do not include information about the speed of the encoder.

Note that each number in the tables below corresponds to some range of bitrates (see Appendix 7. Figures Explanation for more details). 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.

Consider the Y-SSIM results in Table 5 and Y-PSNR results in Table 4. On average, the leader is the x264 encoder followed by DivX H.264 encoder.

Table 3. Average bitrate ratio for the same quality. Usage area "Video Conference". Y-SSIM.

	DivX H.264	Elecard	MainConcept	MSE encoder	Discrete Photon	Webm	x264	XviD
DivX H.264	100%	110%	129%	117%	193%	141%	69%	157%
Elecard	91%	100%	119%	105%	167%	127%	64%	144%
MainConcept	77%	84%	100%	89%	134%	107%	55%	123%
MSE encoder	86%	96%	113%	100%	174%	125%	58%	137%
Discrete Photon	52%	60%	75%	57%	100%	69%	35%	89%
Webm	71%	79%	93%	80%	144%	100%	47%	113%
x264	145%	157%	182%	172%	286%	213%	100%	223%
XviD	64%	70%	81%	73%	113%	89%	45%	100%

Table 4. Average bitrate ratio for the same quality. Usage area "Video Conference". Y-PSNR.

	DivX H.264	Elecard	MainConcept	MSE encoder	Discrete Photon	Webm	x264	XviD
DivX H.264	100%	109%	252%	109%	185%	125%	88%	157%
Elecard	92%	100%	214%	99%	165%	115%	81%	144%
MainConcept	40%	47%	100%	41%	74%	41%	31%	123%
MSE encoder	92%	101%	246%	100%	177%	117%	80%	137%
Discrete Photon	54%	60%	135%	57%	100%	62%	45%	89%
Webm	80%	87%	244%	86%	161%	100%	68%	113%
x264	113%	123%	326%	124%	224%	148%	100%	223%
XviD	63%	68%	138%	67%	112%	78%	53%	100%

Figure 17 and Figure 18 depict the data from the tables above. Each line in the figures 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.

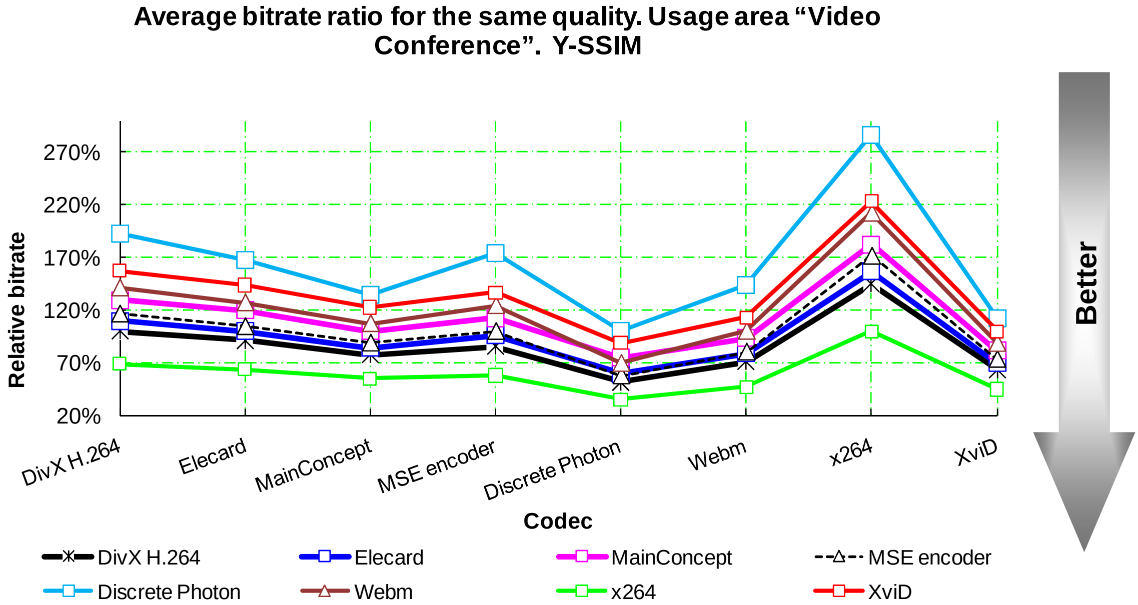


Figure 17. Average bitrate ratio for a fixed quality—usage area "Video Conference," Y-SSIM metric

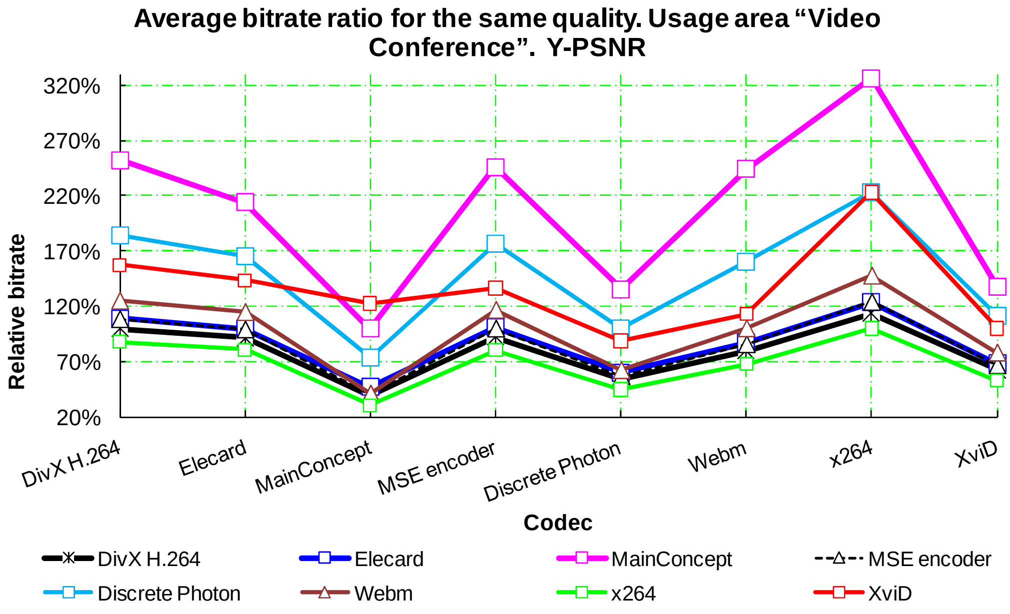


Figure 18. Average bitrate ratio for a fixed quality—usage area "Video Conference," Y-PSNR metric

4.2 Movies

4.2.1 RD Curves

4.2.1.1 High Speed Preset

Figure 19 and Figure 20 show typical situation for all encoders (except some sequences). Figure 19 shows all encoders results and Figure 20 – only for encoders that fit encoding speed requirements. For quality analysis we used only encoders that fit encoding speed requirements encoders. The leader for almost all video sequences (except Crew and Harbour) is x264. MainConcept is typically second. Third place is for Elecard. Discrete Photon exhibited the poorest result (even lower than XviD sometimes).

PSNR results are shown at Figure 21 and Figure 22. PSNR usage as main metric changes results strongly for example, MainConcept shows lower quality and MSE shows better quality at average.

MSE and WebM High Speed presets does not fit the encoding speed requirements.

Results for all the sequences, all metrics and all encoders are available in ***Enterprise version*** *report only.*

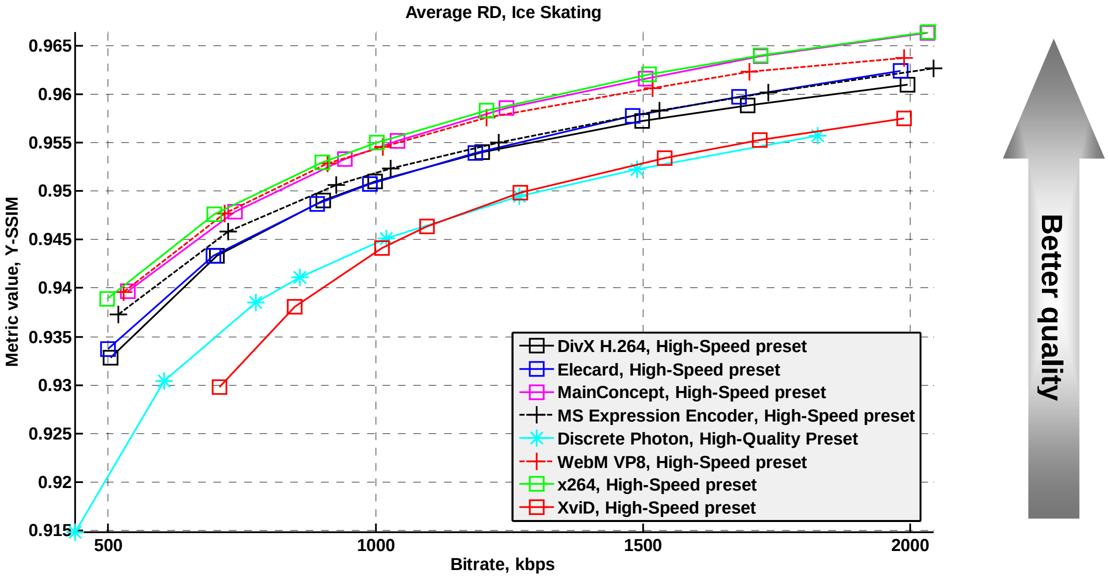


Figure 19. Bitrate/quality—usage area "Movies," "Ice Skating" sequence, High Speed preset, Y-SSIM metric. All encoders

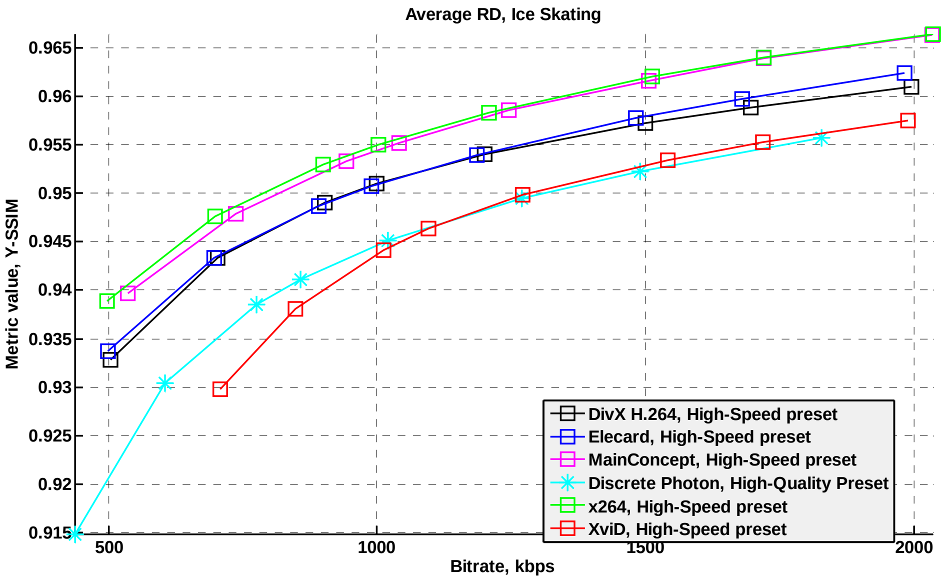


Figure 20. Bitrate/quality—usage area "Movies," "Ice Skating" sequence, High Speed preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

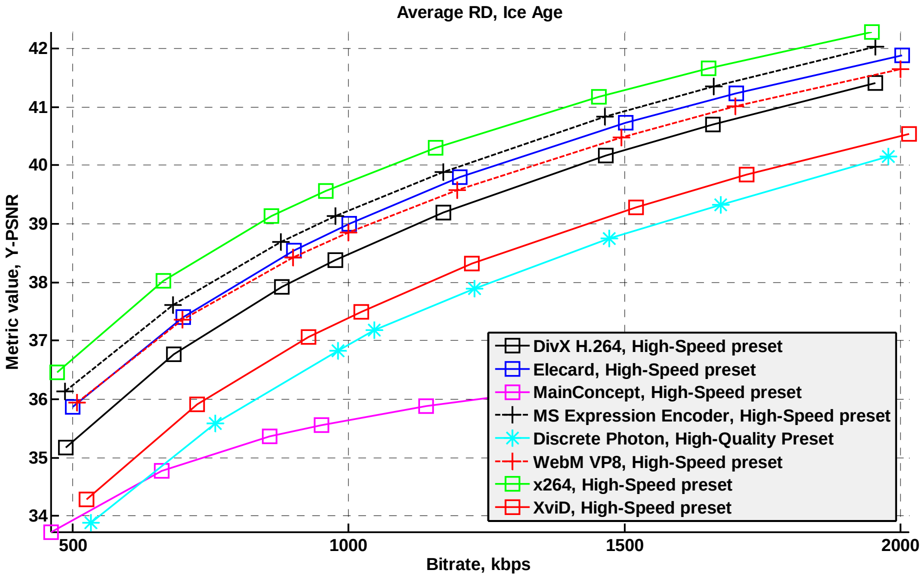


Figure 21. Bitrate/quality—usage area "Movies," "Ice Age" sequence, High Speed preset, Y-PSNR metric. All encoders

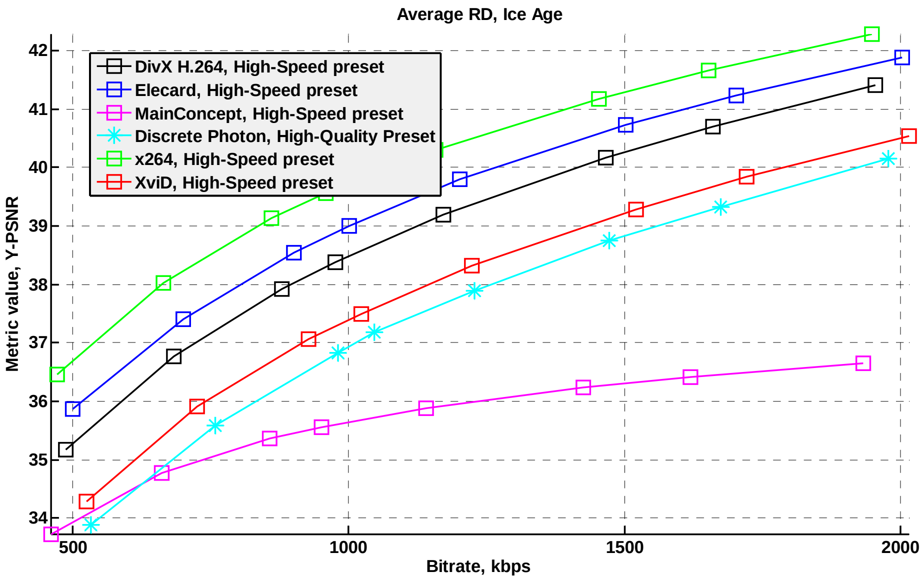


Figure 22. Bitrate/quality—usage area "Movies," "Ice Age" sequence, High Speed preset, Y-PSNR metric. Encoders that fit encoding speed requirements.

4.2.1.2 Normal Preset

The Normal preset results for each sequence are presented in Figure 23 through Figure 28. The first two figures show the Y-SSIM results, and the last two figures show the Y-PSNR results. The results depend on the metric used.

SSIM metric: The leader is x264; three encoders (MainConcept, DivX H.264 and Elecard) placed second – the position depends on sequence tested.

PSNR metric: Results differ strongly, for example Elecard shows better results as for SSIM and MainConcept shows lower results.

MSE and WebM Normal presets does not fit the encoding speed requirements.

*Results for all the sequences, all metrics and all encoders are available in **Enterprise version** report only.*

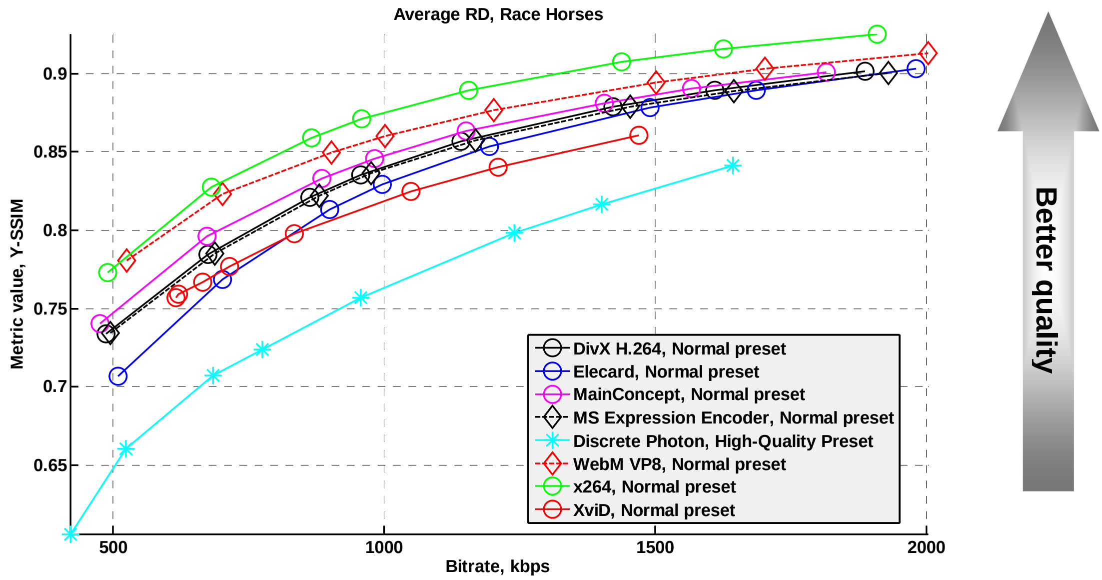


Figure 23. Bitrate/quality—usage area "Movies," "Race Horses" sequence, Normal preset, Y-SSIM metric. All encoders

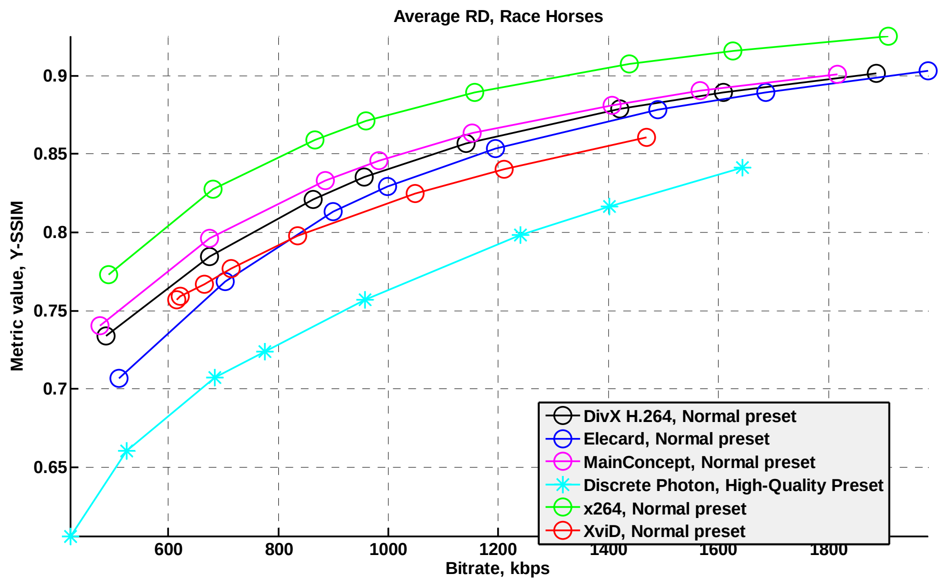


Figure 24. Bitrate/quality—usage area "Movies," "Race Horses" sequence, Normal preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

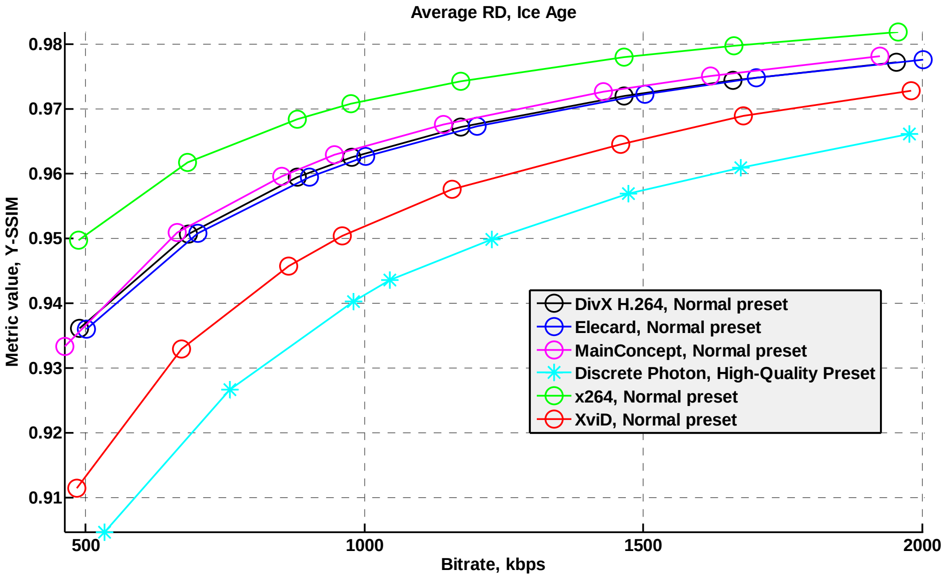


Figure 25. Bitrate/quality—usage area "Movies," "Ice Age" sequence, Normal preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

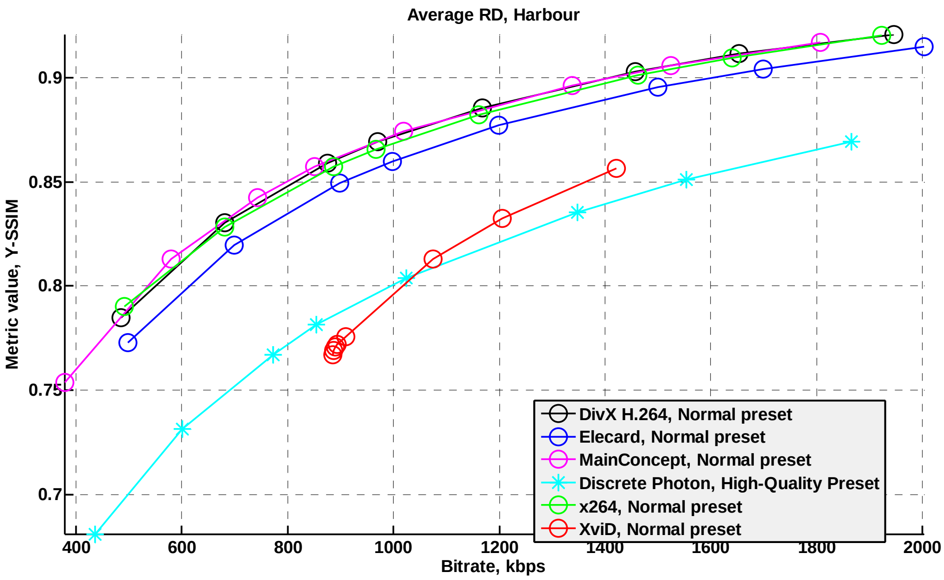


Figure 26. Bitrate/quality—usage area "Movies," "Harbour" sequence, Normal preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

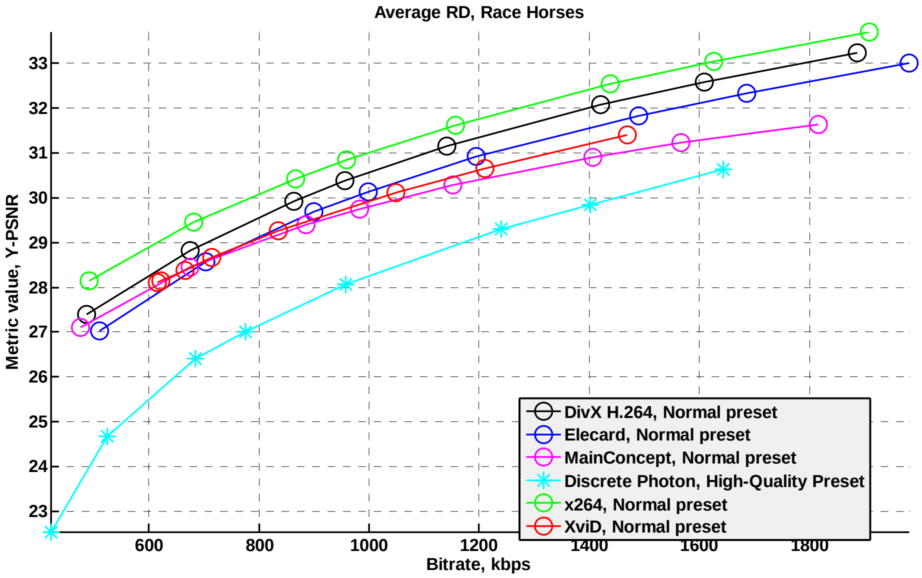


Figure 27. Bitrate/quality—usage area "Movies," "Race Horses" sequence, Normal preset, Y-PSNR metric. Encoders that fit encoding speed requirements.

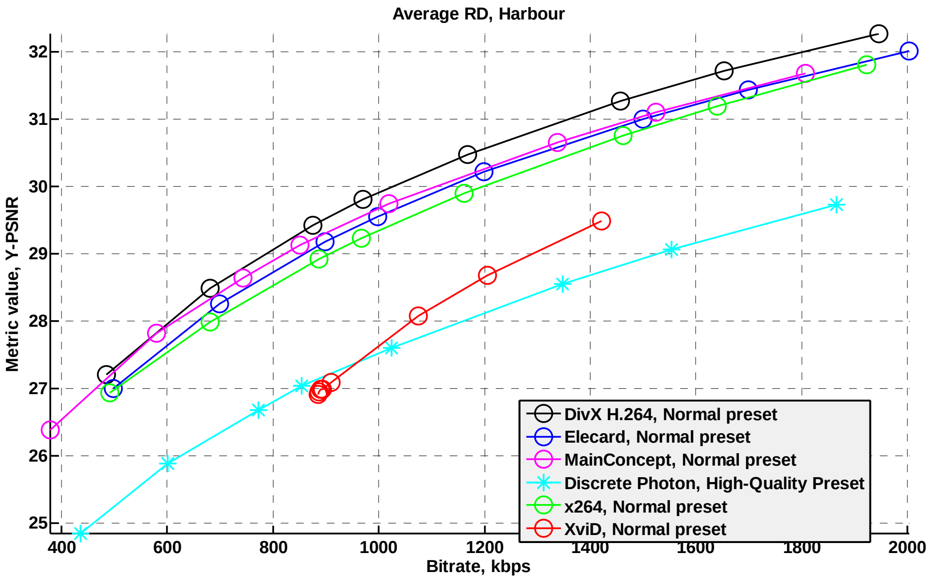


Figure 28. Bitrate/quality—usage area "Movies," "Harbour" sequence, Normal preset, Y-PSNR metric. Encoders that fit encoding speed requirements.

4.2.1.3 High Quality Preset

The High Quality preset results for each sequence are presented in Figure 29 through Figure 34. The first four graphs show the Y-SSIM results, and the last two graphs show the Y-PSNR results. The results change depending on the metric used.

SSIM metric: The leader is x264, followed by MainConcept in second place and the DivX H.264, Elecard and MSE encoders in third place. And these four encoders show close results that vary on different sequences. DiscretePhoton shows the lowest results.

PSNR metric: The leader is the same, but encoders at second place exchange their places.

MSE and WebM High Quality presets do not fit the encoding speed requirements.

Results for all the sequences, all metrics and all encoders are available in ***Enterprise version*** *report only.*

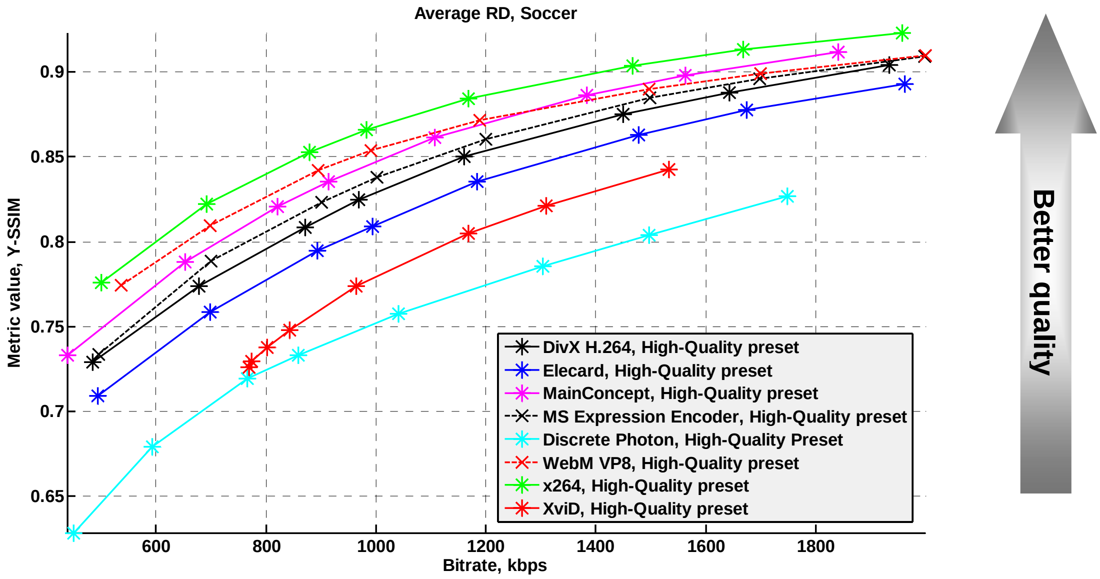


Figure 29. Bitrate/quality—usage area "Movies," "Soccer" sequence, High Quality preset, Y-SSIM metric. All encoders

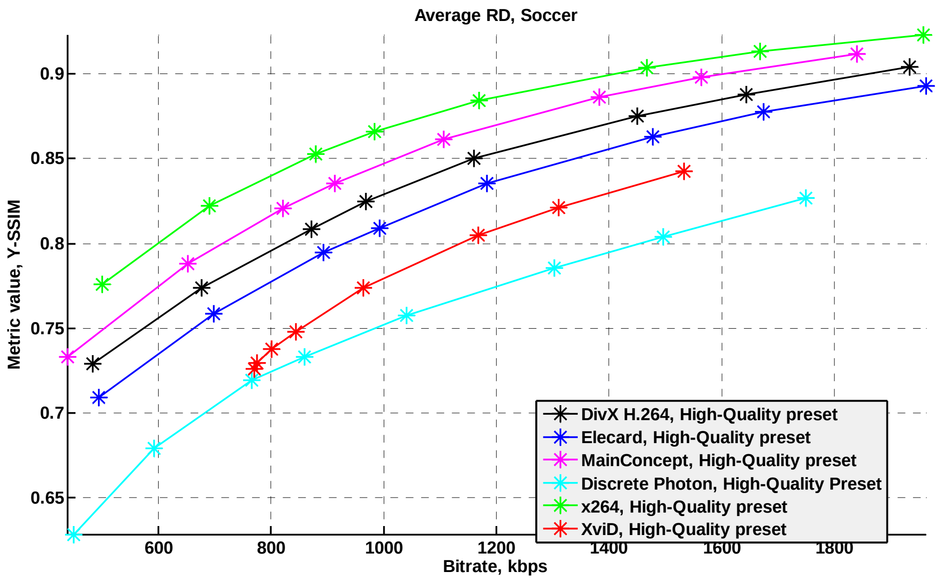


Figure 30. Bitrate/quality—usage area "Movies," "Soccer" sequence, High Quality preset, Y-SSIM metric. Encoders that fit encoding speed

requirements.

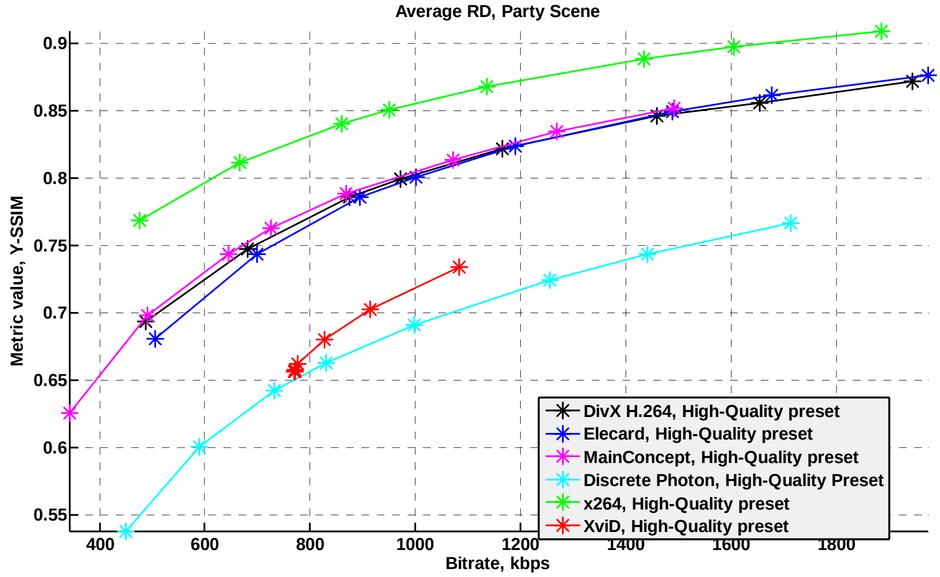


Figure 31. Bitrate/quality—usage area "Movies," "Party Scene" sequence, High Quality preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

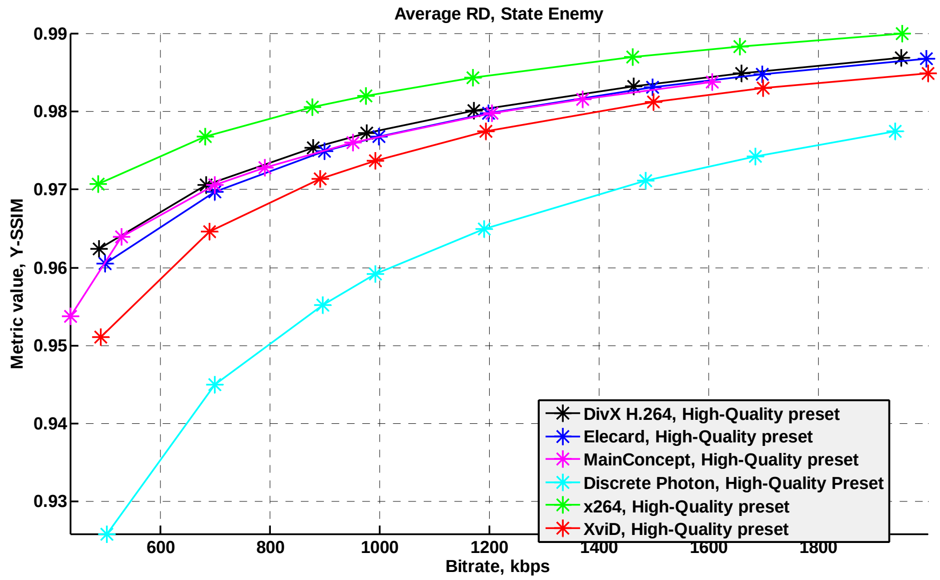


Figure 32. Bitrate/quality—usage area "Movies," "State Enemy" sequence, High Quality preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

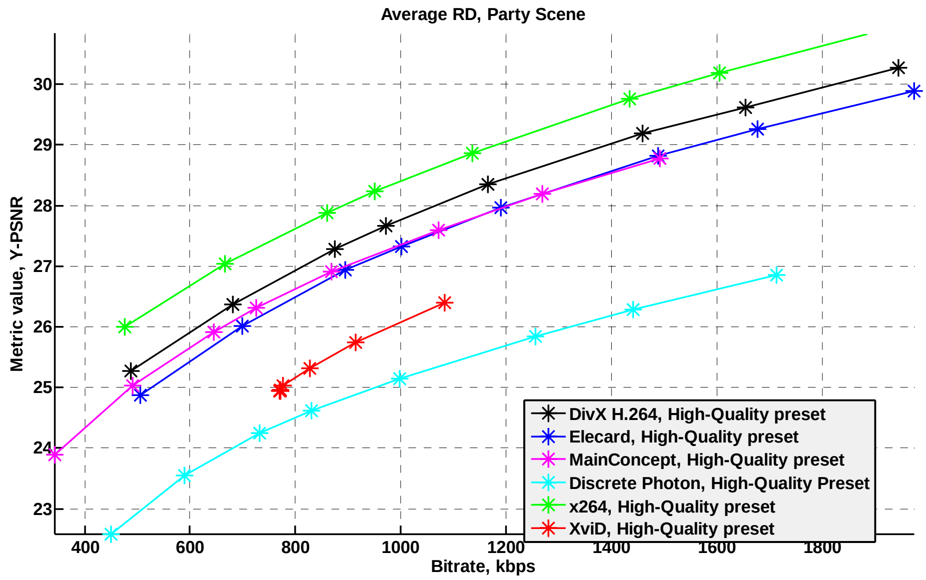


Figure 33. Bitrate/quality—usage area "Movies," "Party Scene" sequence, High Quality preset, Y-PSNR metric. Encoders that fit encoding speed requirements.

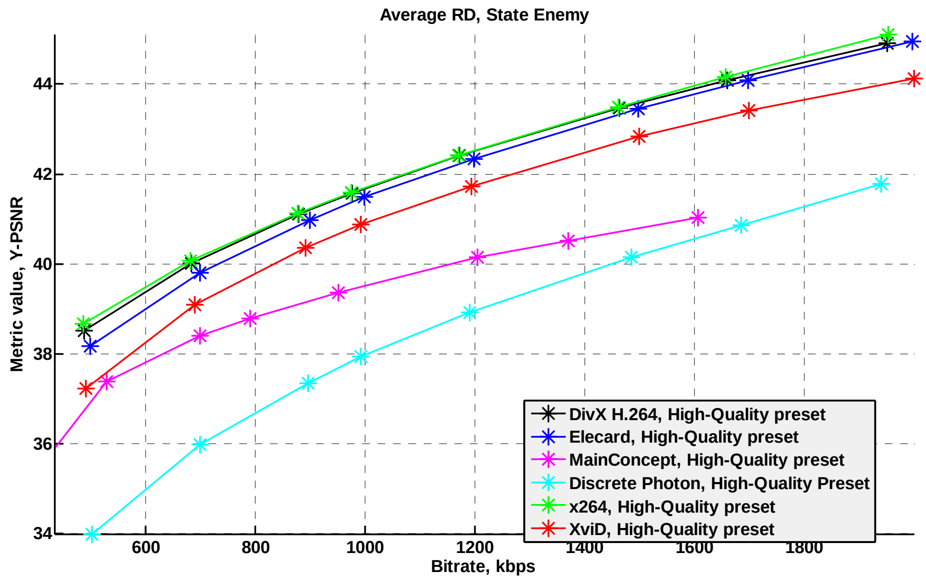


Figure 34. Bitrate/quality—usage area "Movies," "State Enemy" sequence, High Quality preset, Y-PSNR metric. Encoders that fit encoding speed requirements.

4.2.2 Encoding Speed

4.2.2.1 High Speed Preset

Absolute speed results are presented in Figure 35 through Figure 36. All the encoders except MSE have a similar growth rate for encoding time as the bitrate is increased. XviD is the fastest.

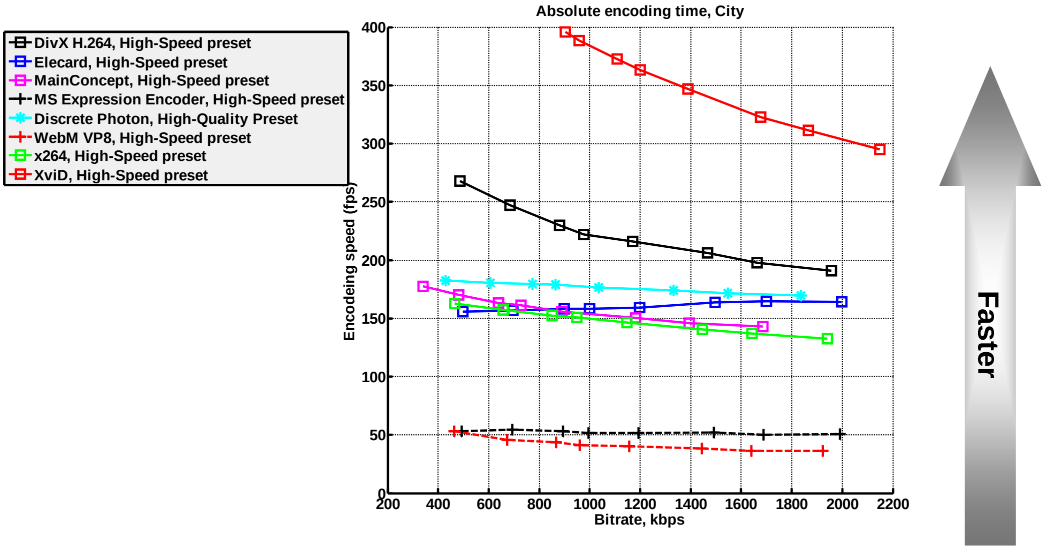


**Figure 35. Encoding speed—usage area "Movie"
"City" sequence, "High Speed" preset**

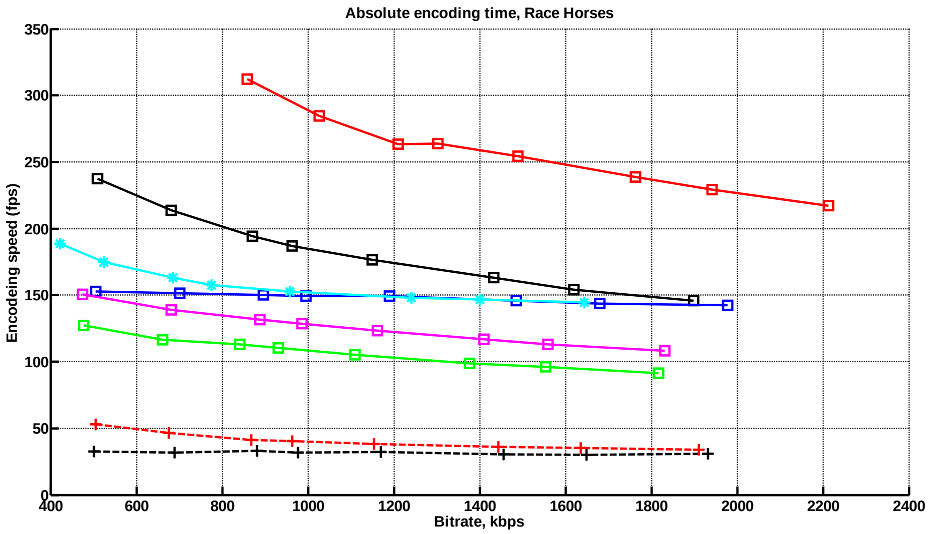


**Figure 36. Encoding speed—usage area "Movies"
"Race Horses" sequence, High Speed preset**

4.2.2.2 Normal Preset

Absolute speed results are presented in Figure 37 through Figure 39. All the encoders except MSE and WebM have a similar growth rate for encoding time versus increasing bitrate. DiscretePhoton is the fastest encoder at almost all the sequences.

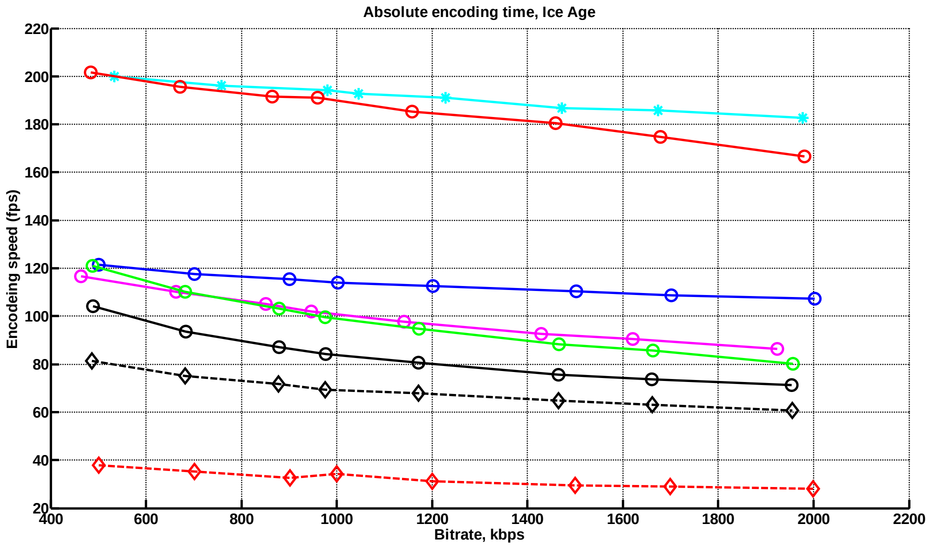


Figure 37. Encoding speed—usage area "Movies" "Ice Age" sequence, Normal preset

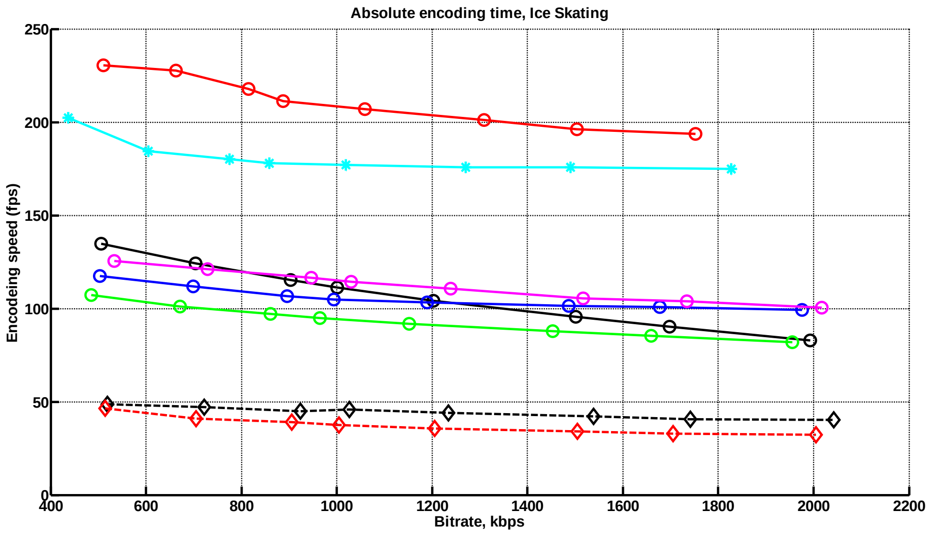


Figure 38. Encoding speed—usage area "Movies" "Ice Skating" sequence, Normal preset

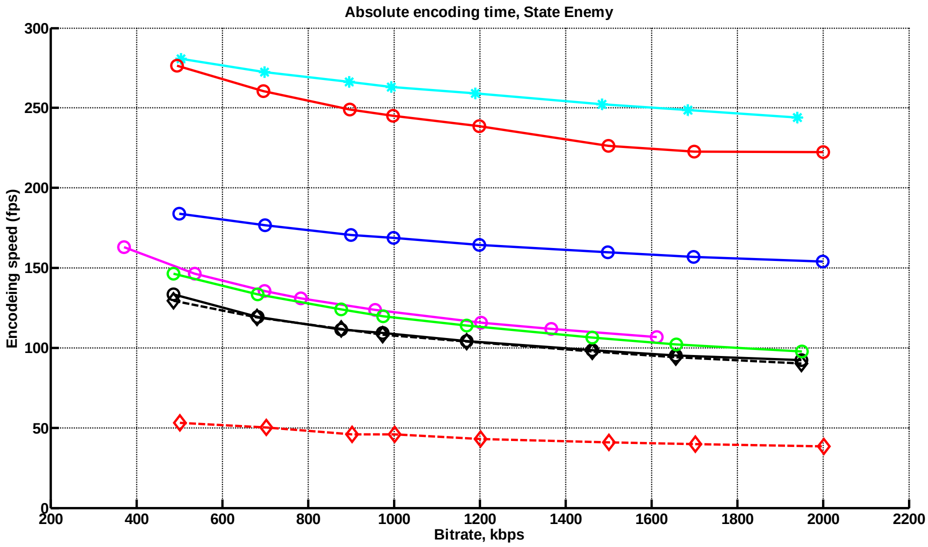


**Figure 39. Encoding speed—usage area "Movies"
"State Enemy" sequence, Normal preset**

4.2.2.3 High Quality Preset

Absolute speed results are presented in Figure 40 through Figure 41. The situation is close to Normal Speed preset at average.

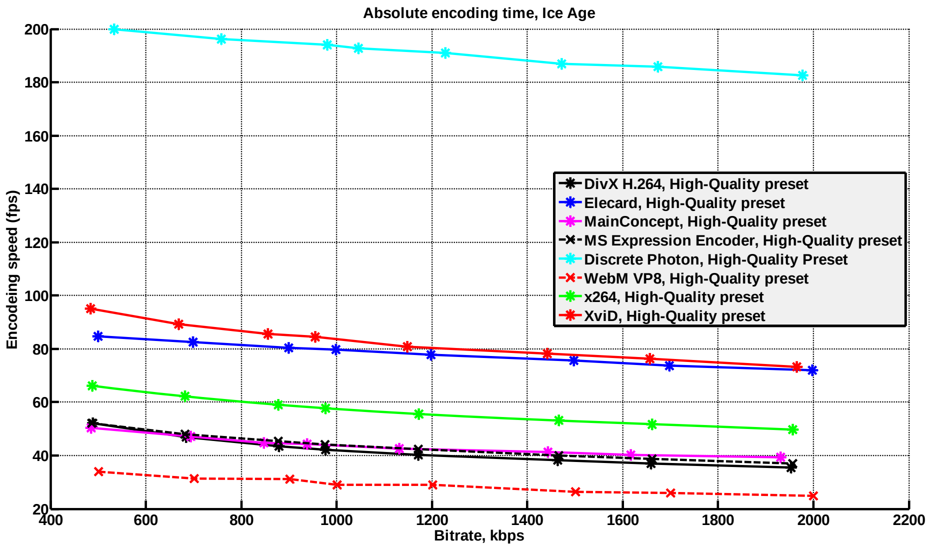


**Figure 40. Encoding speed—usage area "Movies"
"Ice Age" sequence, High Quality preset**

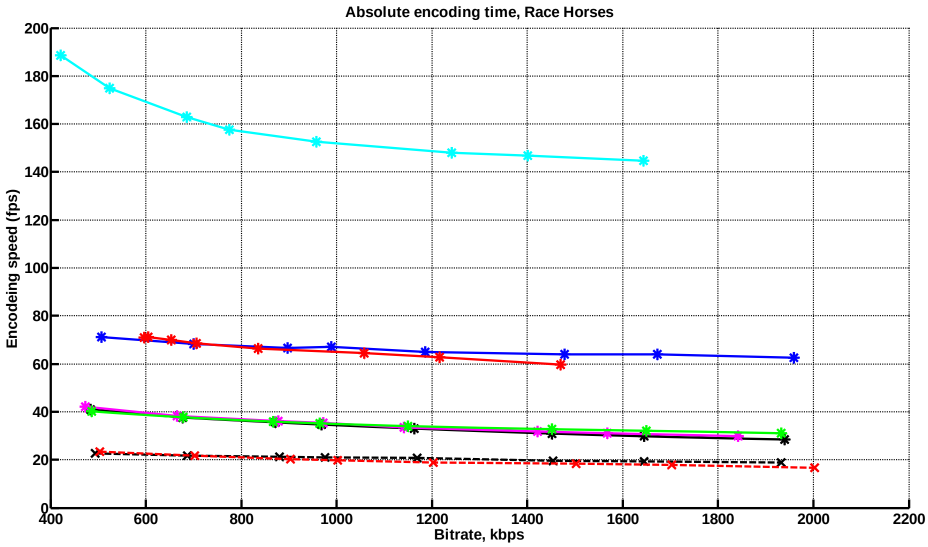


Figure 41. Encoding speed—usage area "Movies" "Race Horses" sequence, High Quality preset

4.2.3 Speed/Quality Trade-Off

Detailed descriptions of the speed/quality trade-off graphs can be found in Appendix 7. Figures Explanation. 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. XviD 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.

Please note that the method of averaging among all sequences assumes that all codecs produced results for each sequence. When this is not the case, only existing results are taken into account.

4.2.3.1 High Speed Preset

Figure 42 through Figure 48 show results for the High Speed preset. The chosen metric has an influence on results.

The five best codecs (no codec performs faster with higher quality) in terms of speed/quality are XviD, DivX H.264, Elecard, MainConcept and x264 at average. But there are sequences where it is not true, for example at City sequence MainConcept is better than x264 and Elecard. PSNR metric usage changes the result: Elecard and DivX H.264 exhibited better results on average than did MainConcept.

*Results for all the sequences, all metrics and all encoders are available in **Enterprise version** report only.*

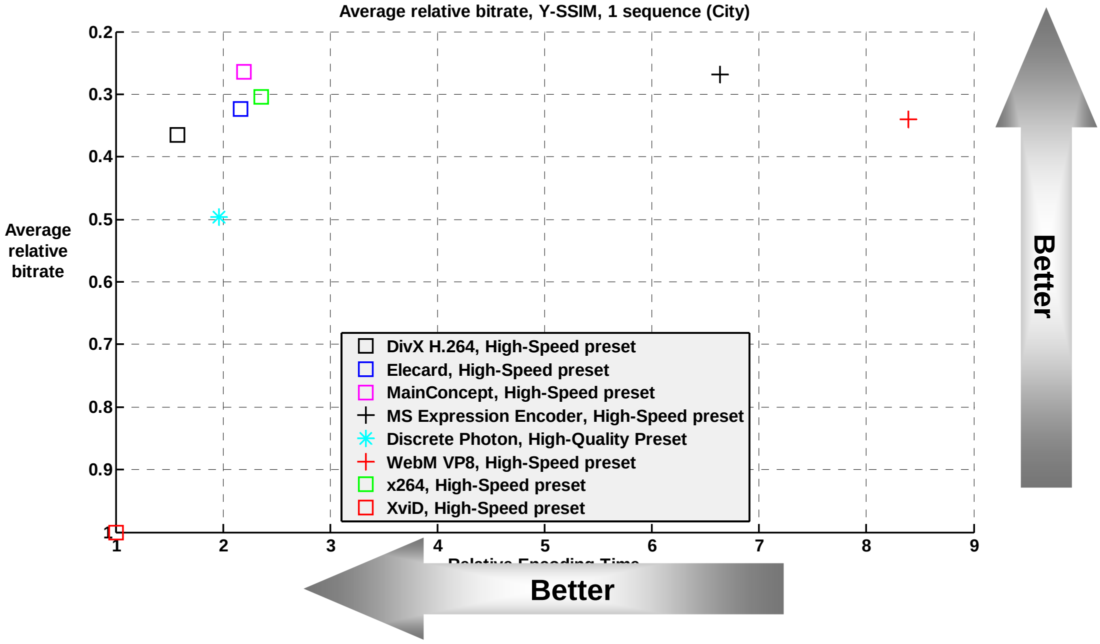


Figure 42. Speed/quality trade-off—usage area "Movies," "City" sequence, High Speed preset, Y-SSIM metric

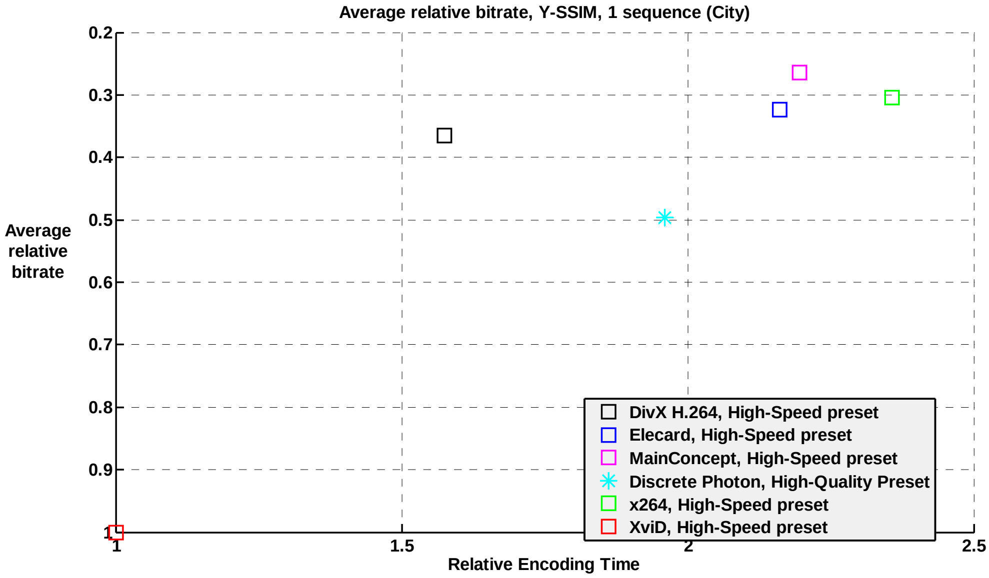


Figure 43. Speed/quality trade-off—usage area "Movies," "City" sequence, High Speed preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

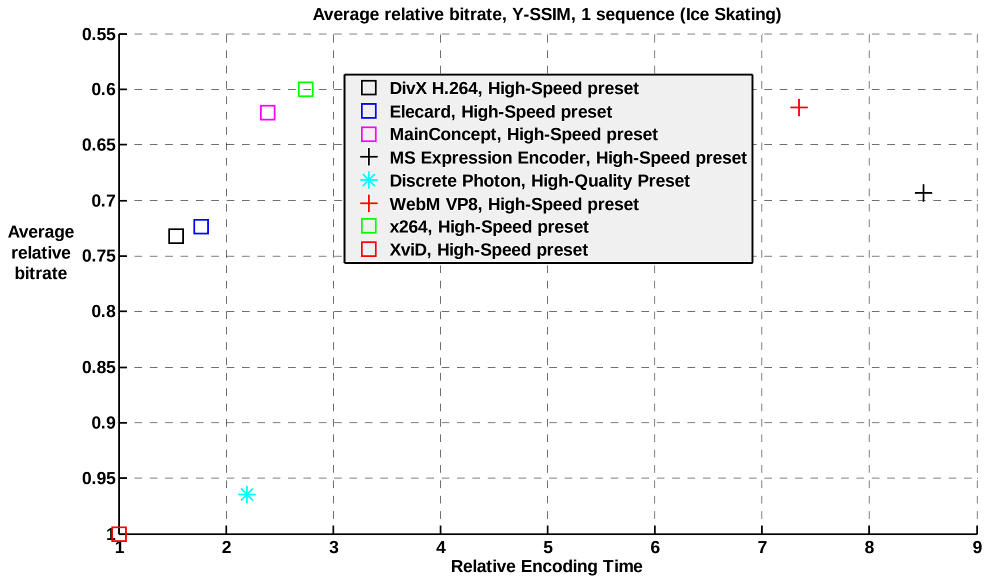


Figure 44. Speed/quality trade-off—usage area "Movies," "Ice Skating" sequence, High Speed preset, Y-SSIM metric

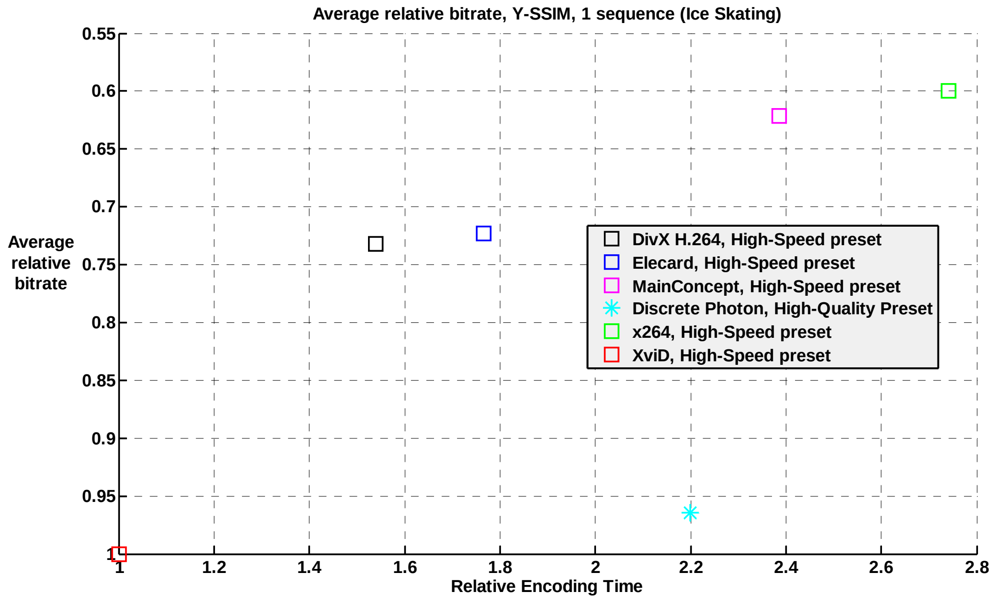


Figure 45. Speed/quality trade-off—usage area "Movies," "Ice Skating" sequence, High Speed preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

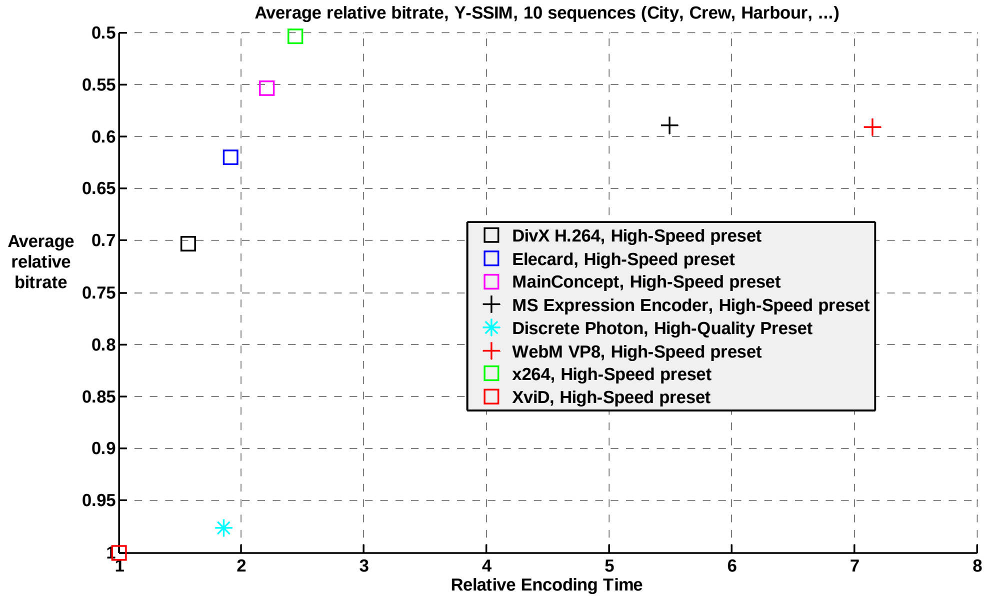


Figure 46. Speed/quality trade-off—usage area "Movies," All "Movie" sequences, High Speed preset, Y-SSIM metric

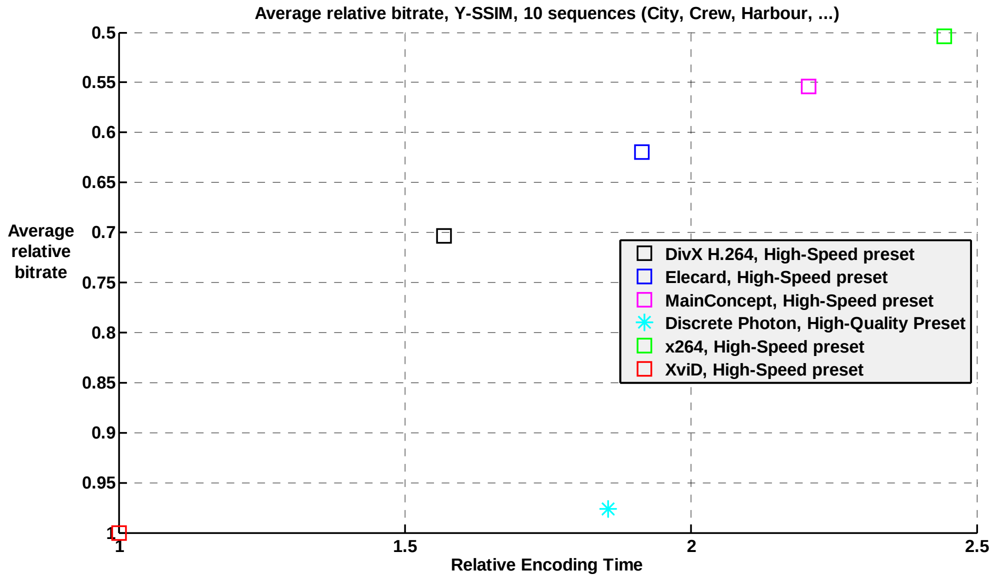


Figure 47. Speed/quality trade-off—usage area "Movies," All "Movie" sequences, High Speed preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

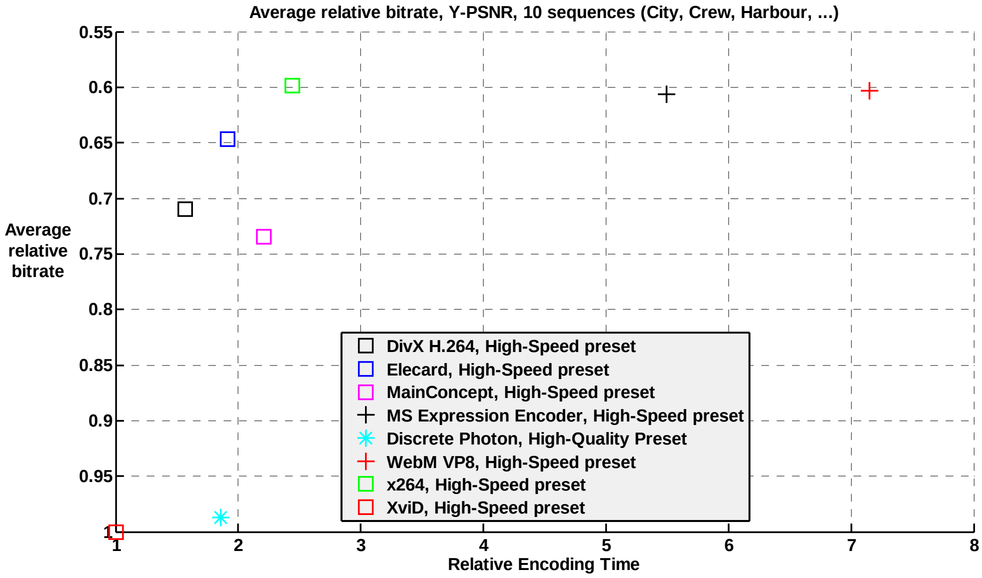


Figure 48. Speed/quality trade-off—usage area “Movies,” All “Movie” sequences, High Speed preset, Y-PSNR metric

4.2.3.2 Normal Preset

Figure 49 through Figure 62 show results for the Normal preset. The results differ depending on the chosen metric.

All codecs that fit requirements except DivX H.264 are best (no codec performs faster with higher quality) in terms of speed/quality at average. But there are sequences where it is not true, for example at Indiana Jones sequence codecs change their places. PSNR metric usage changes the result: there are only three best codecs: XviD, Elecard and x264.

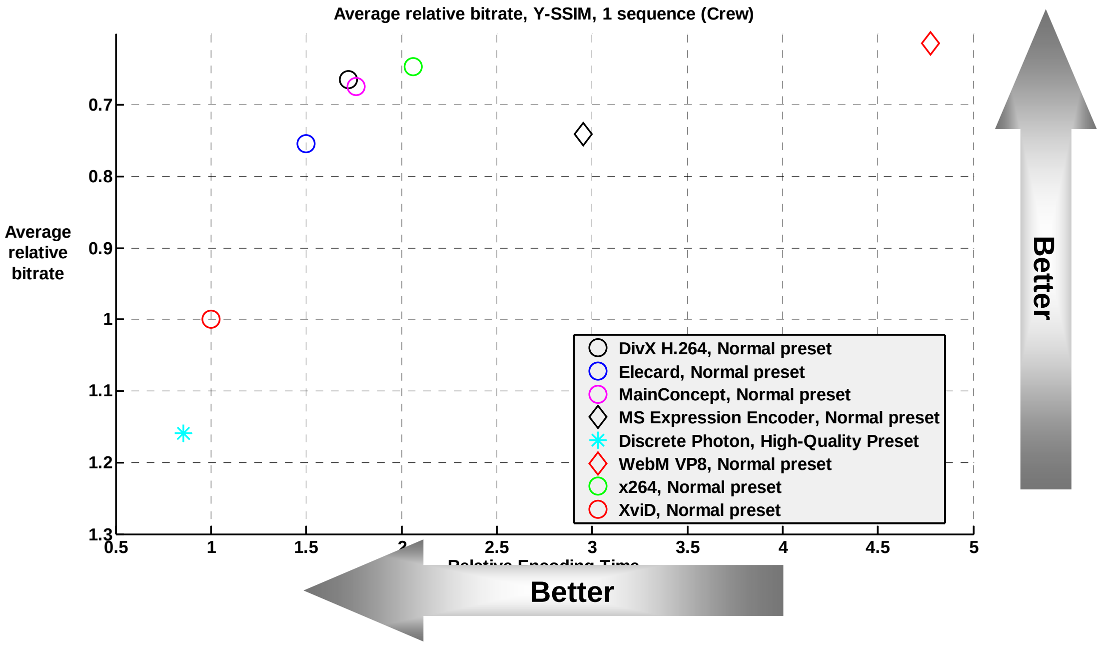


Figure 49. Speed/quality trade-off—usage area “Movies,” “Crew” sequence, Normal preset, Y-SSIM metric

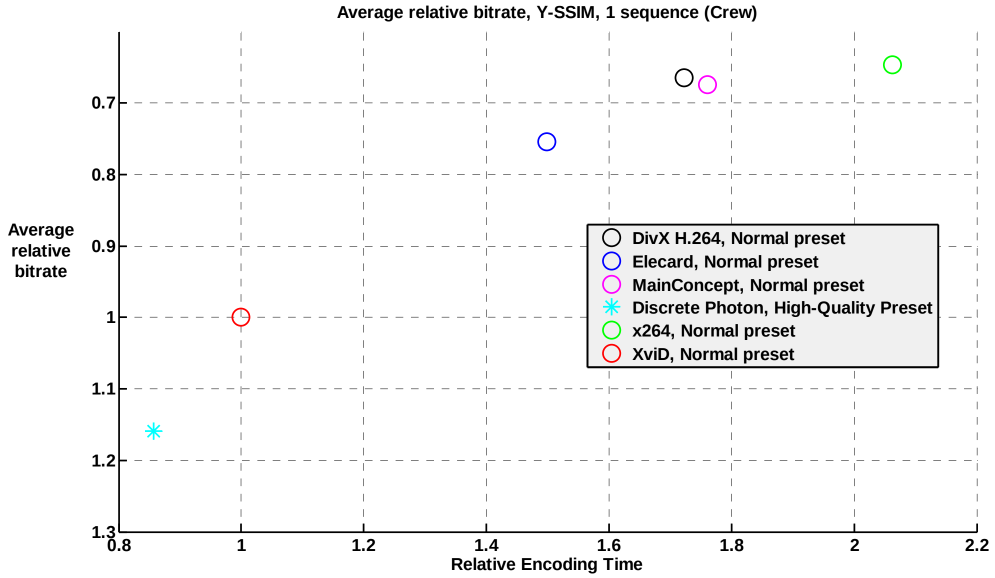


Figure 50. Speed/quality trade-off—usage area “Movies,” “Crew” sequence, Normal preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

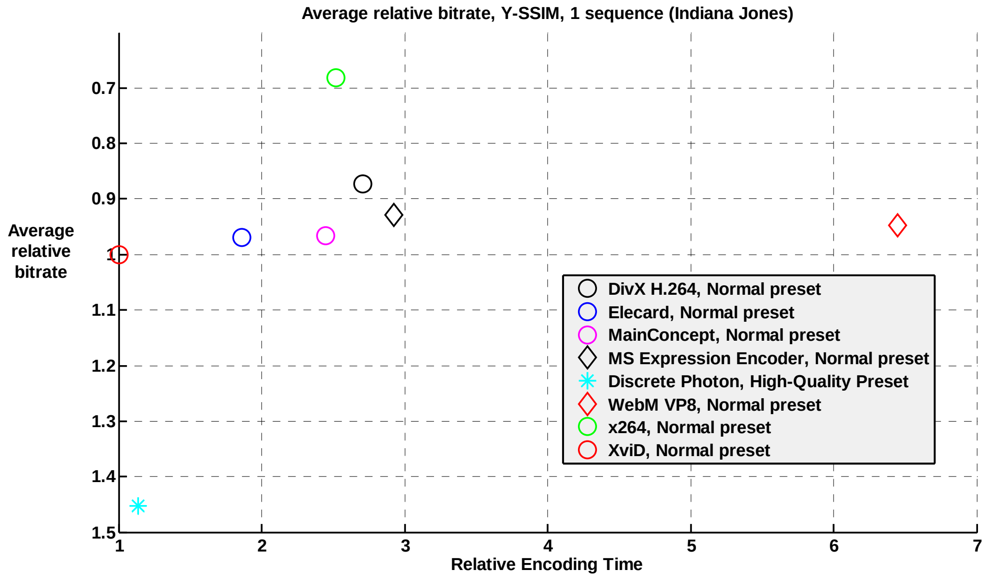


Figure 51. Speed/quality trade-off—usage area “Movies,” “Indiana Jones” sequence, Normal preset, Y-SSIM metric

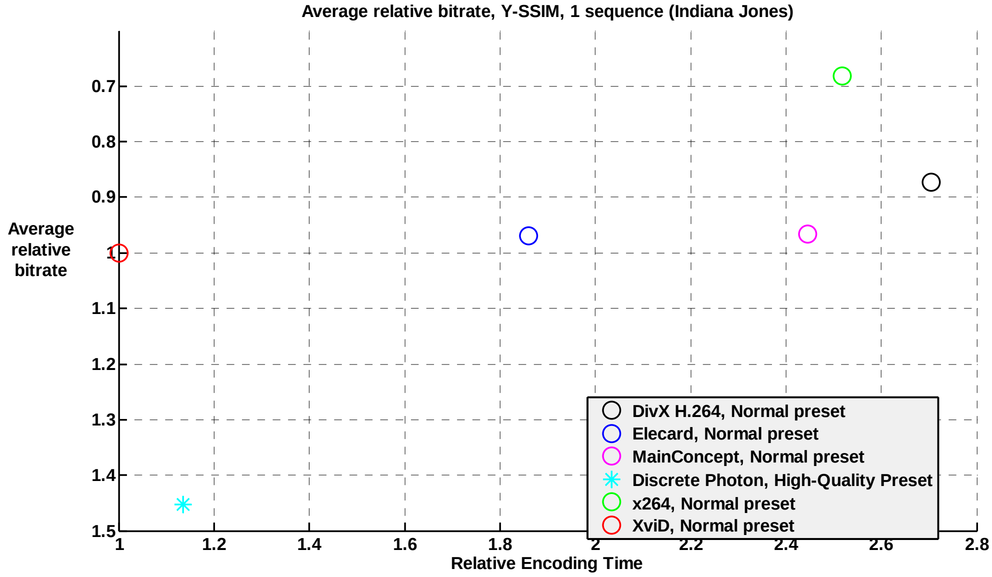


Figure 52. Speed/quality trade-off—usage area “Movies,” “Indiana Jones” sequence, Normal preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

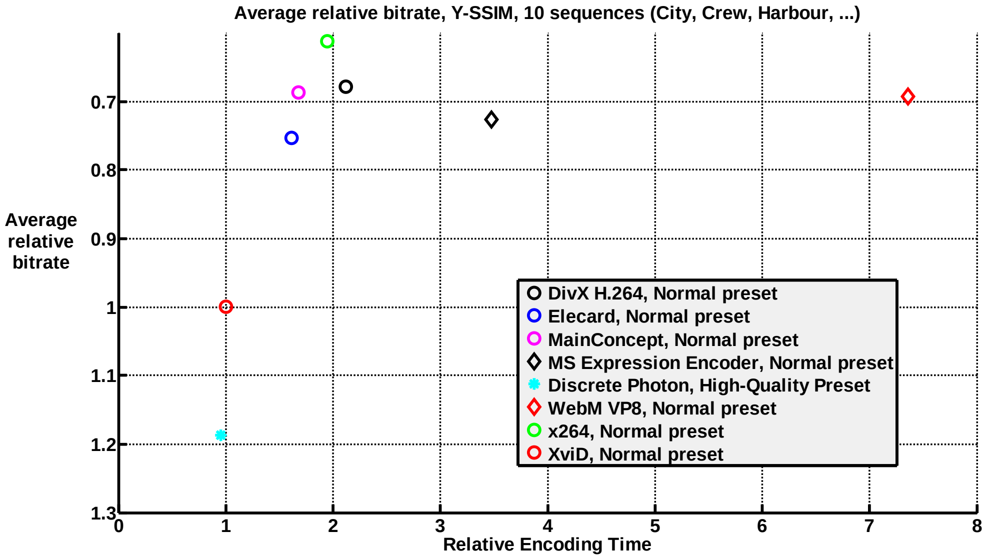


Figure 53. Speed/quality trade-off—usage area “Movies,” All “Movie” sequences, Normal preset, Y-SSIM metric

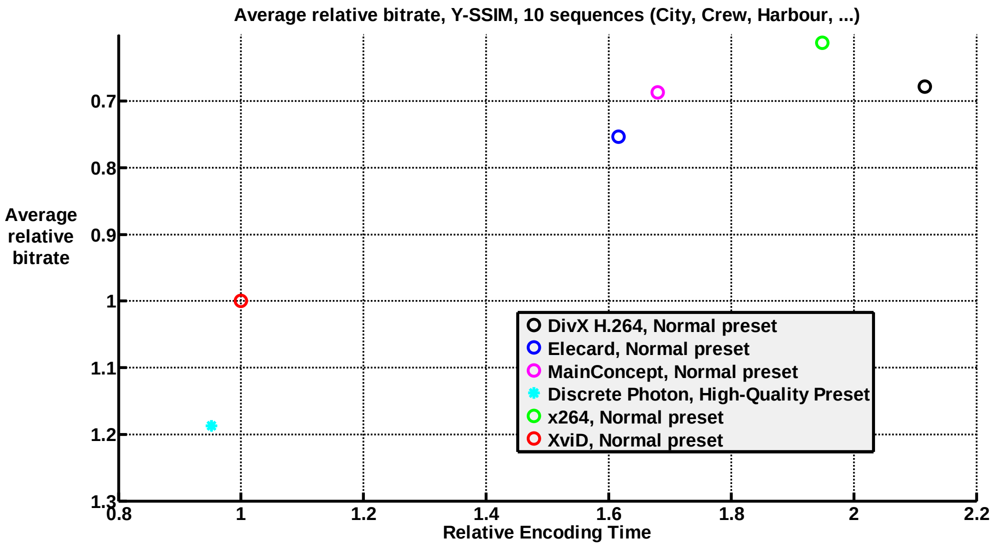


Figure 54. Speed/quality trade-off—usage area "Movies," All "Movie" sequences, Normal preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

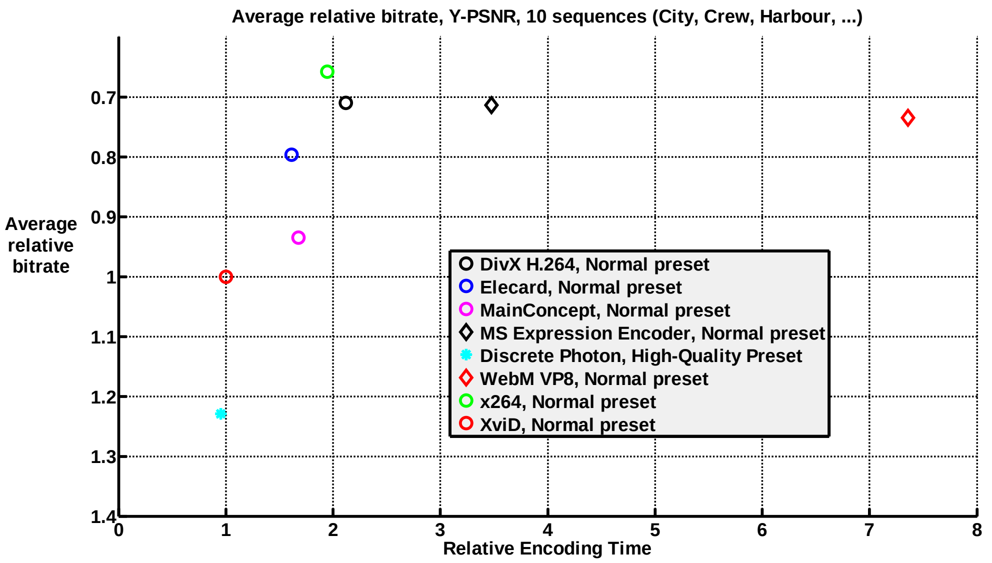


Figure 55. Speed/quality trade-off—usage area "Movies," All "Movie" sequences, Normal preset, Y-PSNR metric

4.2.3.3 High Quality Preset

Figure 56 through Figure 62 show results for the High Quality preset. The results depend on the chosen metric.

The four best codecs (no codec performs faster with higher quality) in terms of speed/quality are DiscretePhoton, Elecard and x264 at average. But there are sequences where results differ to average. PSNR metric usage changes the result: XviD became one of the best codecs too.

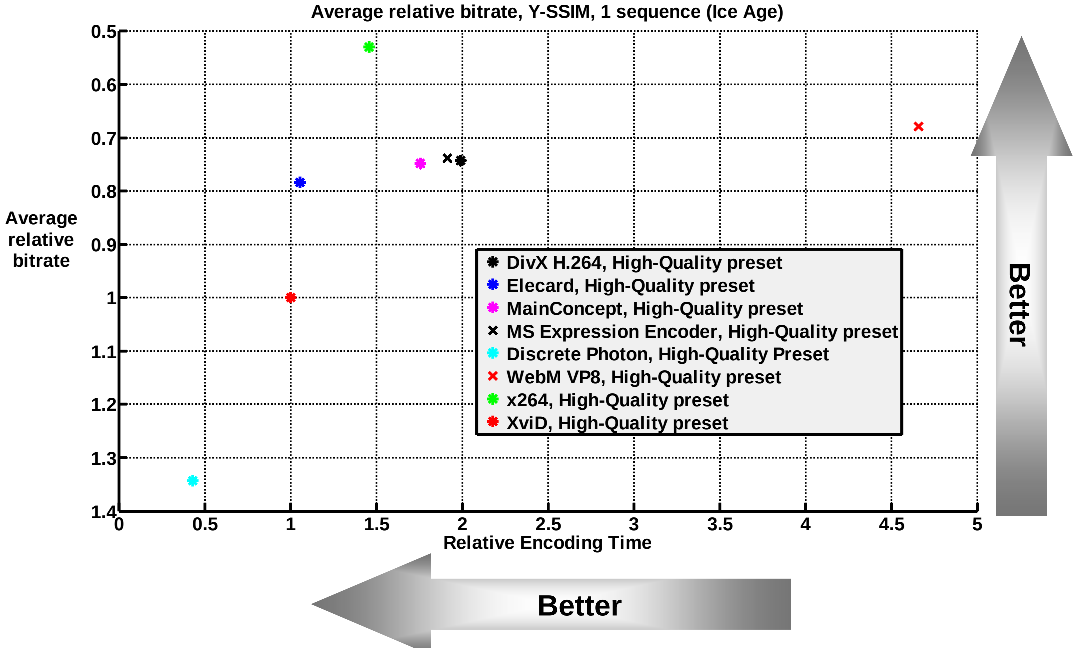


Figure 56. Speed/quality trade-off—usage area "Movies," "Ice Age" sequence, High Quality preset, Y-SSIM metric

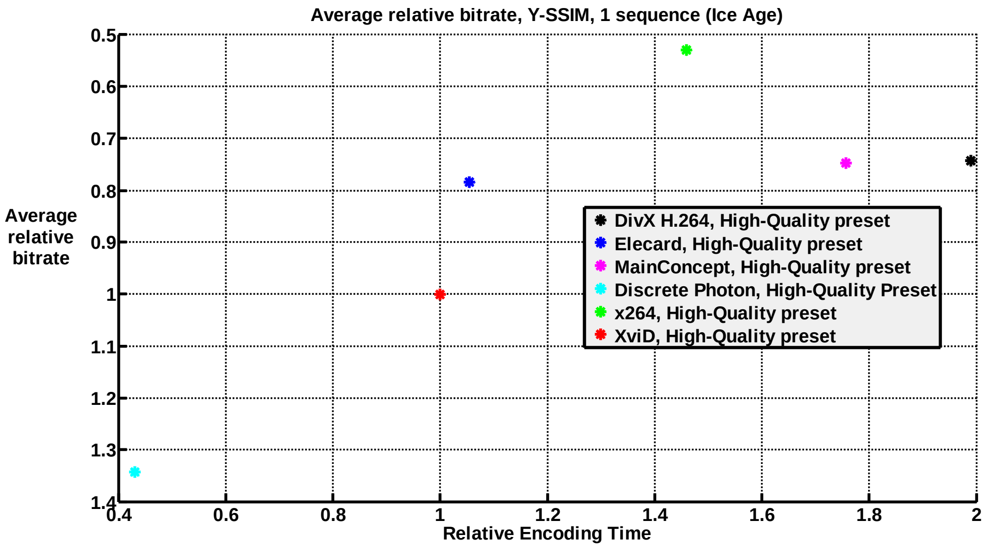


Figure 57. Speed/quality trade-off—usage area "Movies," "Ice Age" sequence, High Quality preset, Y-SSIM metric. Encoders that fit encoding speed requirements

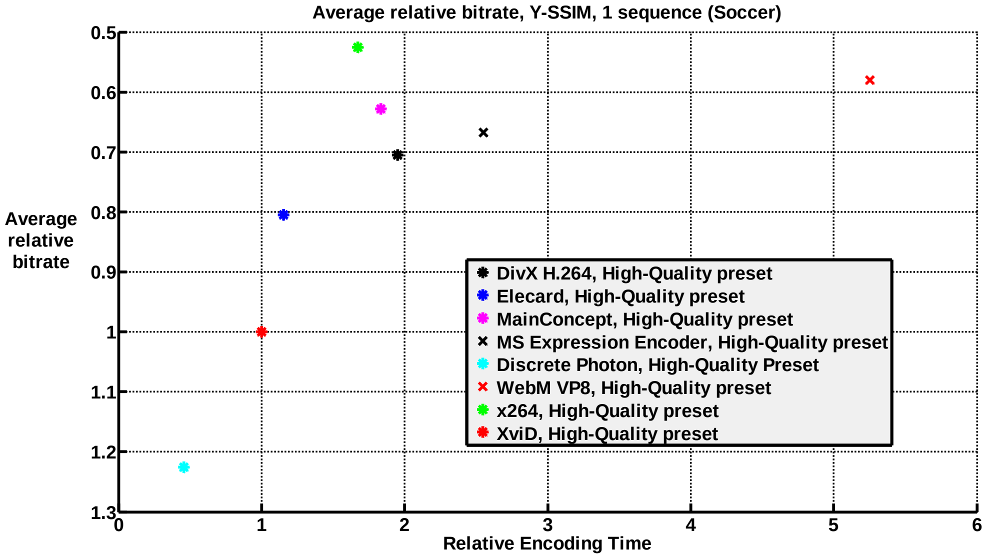


Figure 58. Speed/quality trade-off—usage area “Movies,” “Soccer” sequence, High Quality preset, Y-SSIM metric

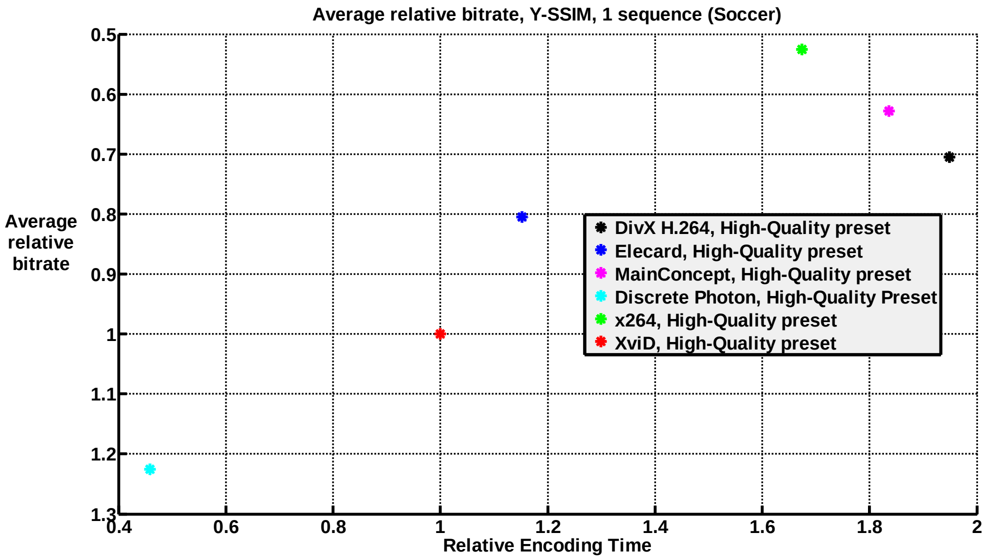


Figure 59. Speed/quality trade-off—usage area “Movies,” “Soccer” sequence, High Quality preset, Y-SSIM metric. Encoders that fit encoding speed requirements

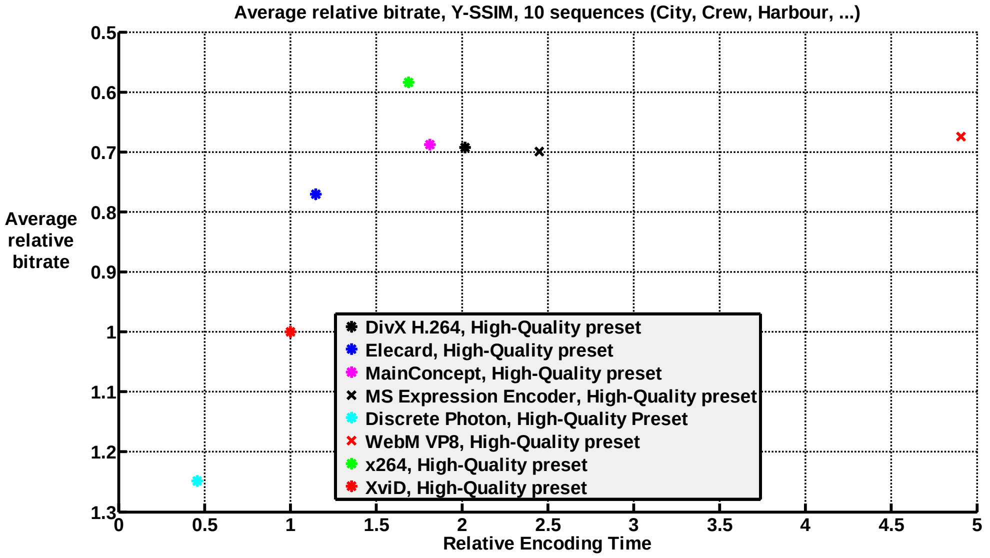


Figure 60. Speed/quality trade-off—usage area "Movies," All "Movie" sequences, High Quality preset, Y-SSIM metric

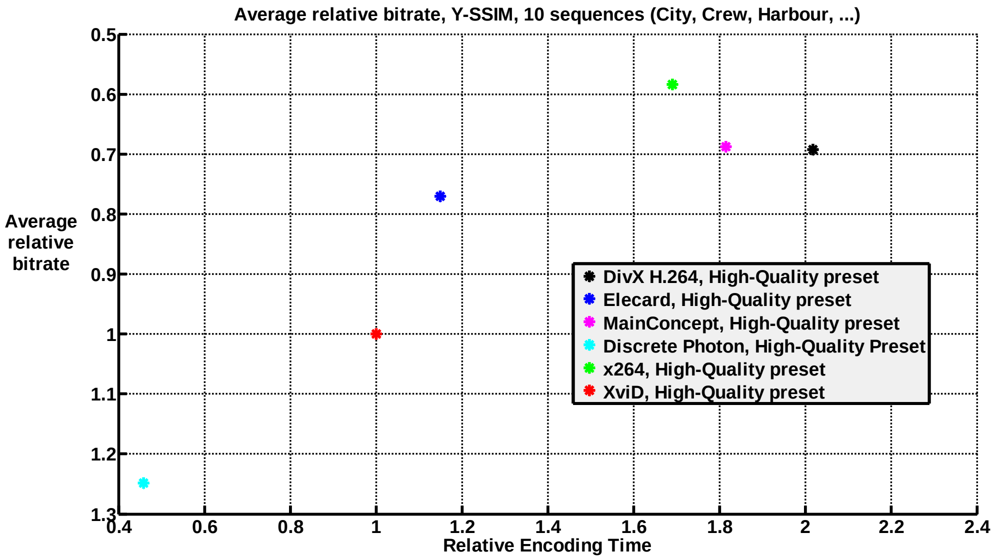


Figure 61. Speed/quality trade-off—usage area "Movies," All "Movie" sequences, High Quality preset, Y-SSIM metric. Encoders that fit encoding speed requirements

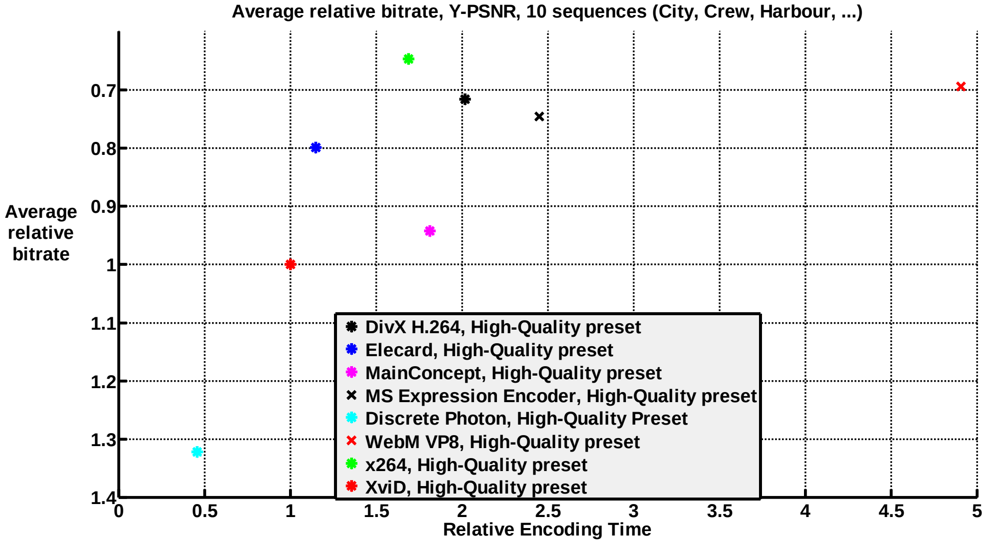


Figure 62. Speed/quality trade-off—usage area "Movies," All "Movie" sequences, High Quality preset, Y-PSNR metric

4.2.4 Bitrate Handling

4.2.4.1 High Speed Preset

Encoders with High Speed presets, except the XviD encoder, demonstrate good bitrate handling for all sequences. There are some issues with bitrate handling for DiscretePhoton encoder for some sequences (for example City and Race Horses sequences).

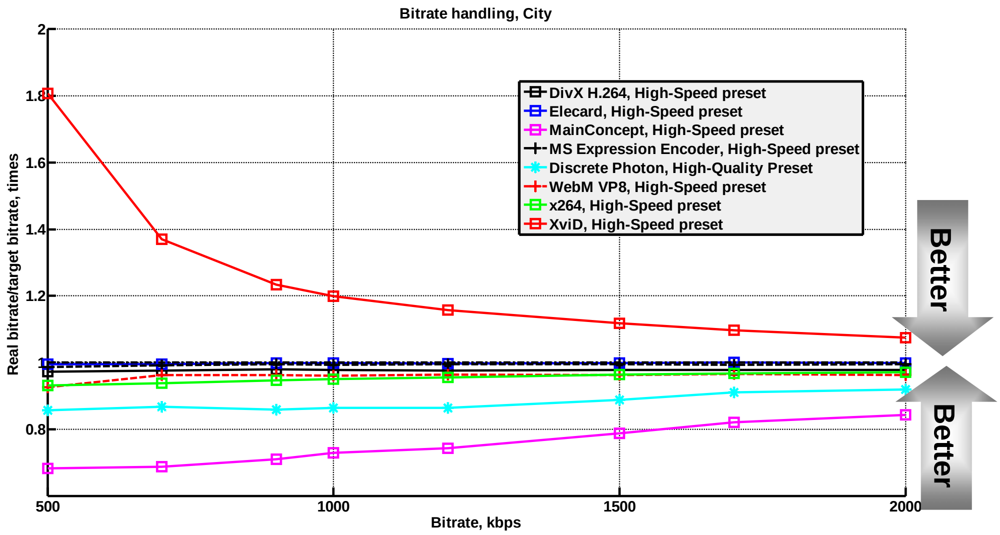


Figure 63. Bitrate handling—usage area "Movies," "City" sequence, High Speed preset

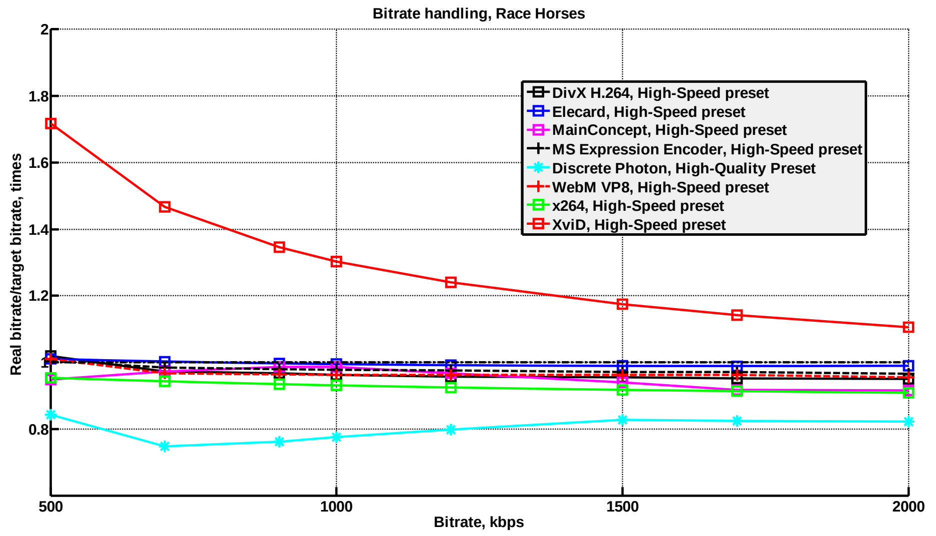


Figure 64. Bitrate handling—usage area "Movies," "Race Horses" sequence, High Speed preset

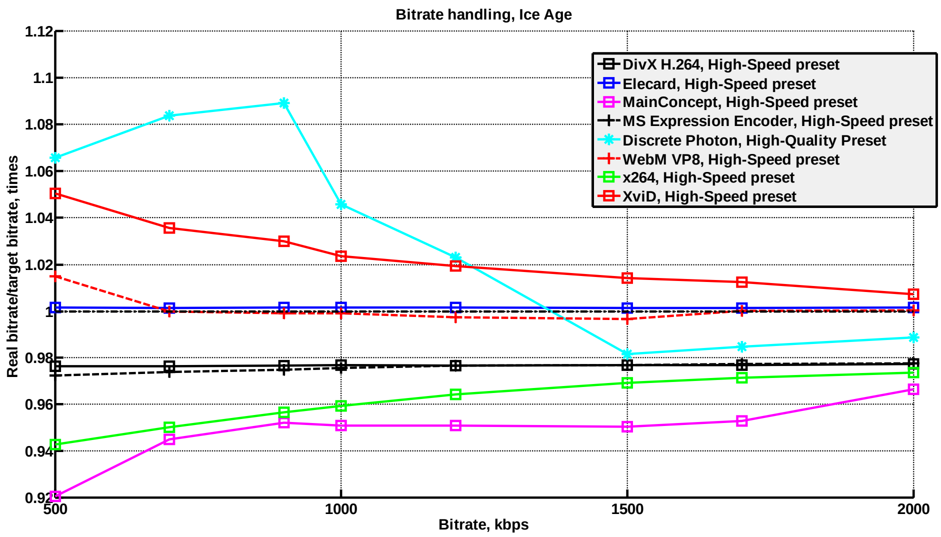


Figure 65. Bitrate handling—usage area "Movies," "Ice Age" sequence, High Speed preset

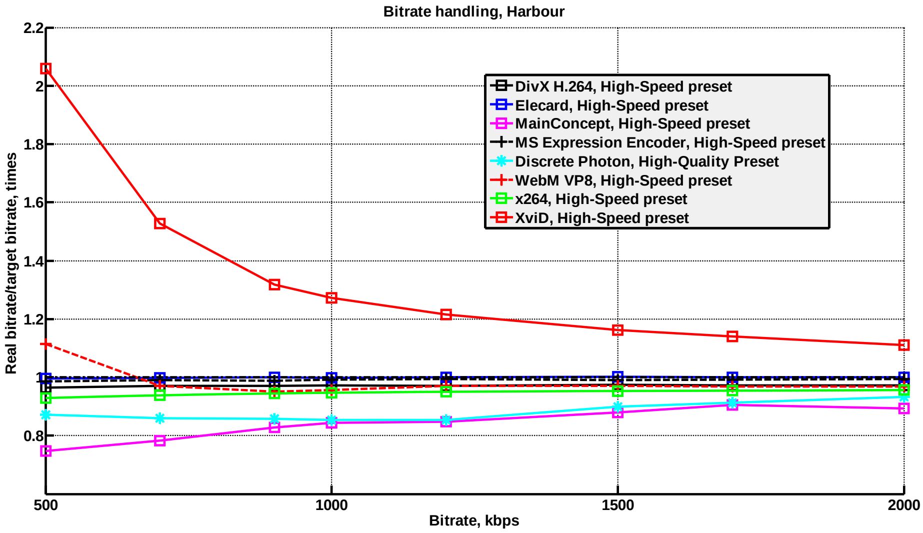


Figure 66. Bitrate handling—usage area "Movies," "Harbour" sequence, High Speed preset

4.2.4.2 Normal Preset

Results are close to HighSpeed results: encoders with High Speed presets, except the XviD encoder, demonstrate good bitrate handling for all sequences. There are some issues with bitrate handling for DiscretePhoton encoder for some sequences (for example, Race Horses sequence).

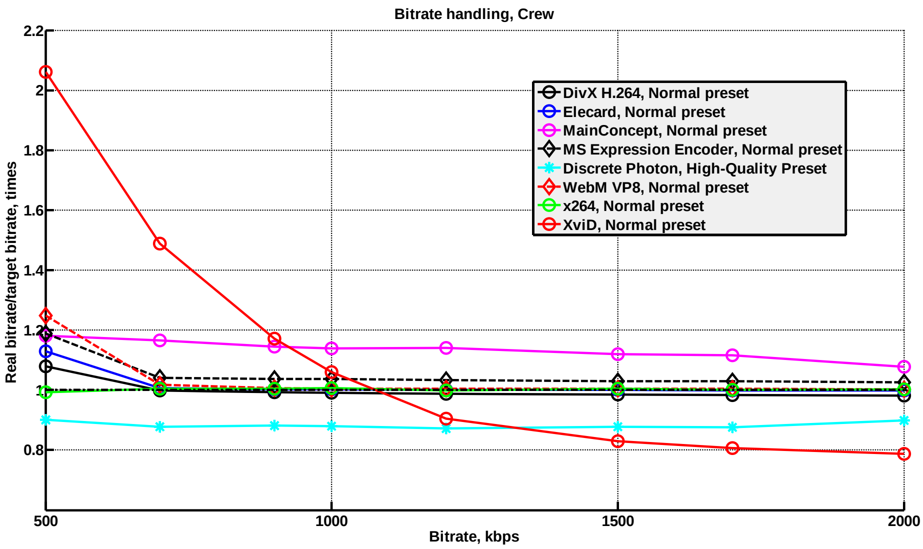


Figure 67. Bitrate handling—usage area "Movies," "Crew" sequence, Normal preset

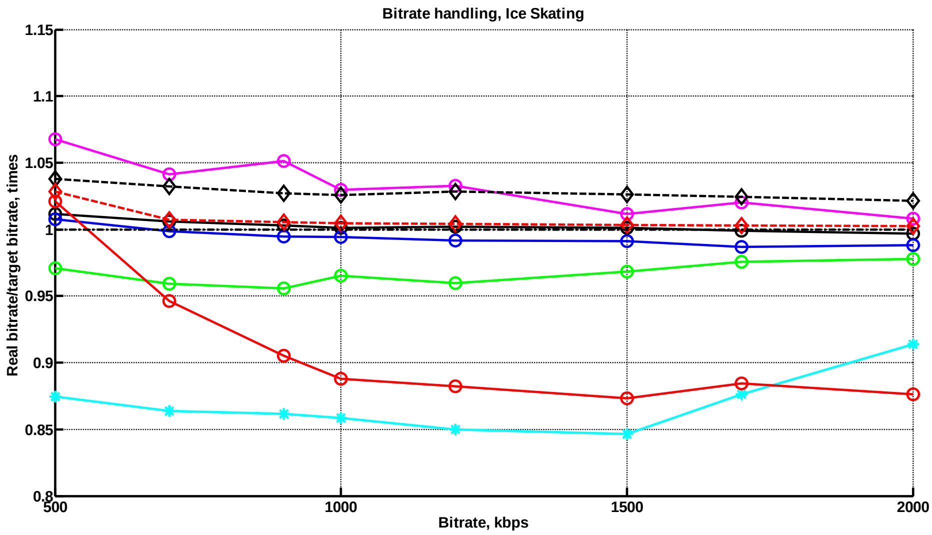


Figure 68. Bitrate handling—usage area "Movies," "Ice Skating" sequence, Normal preset

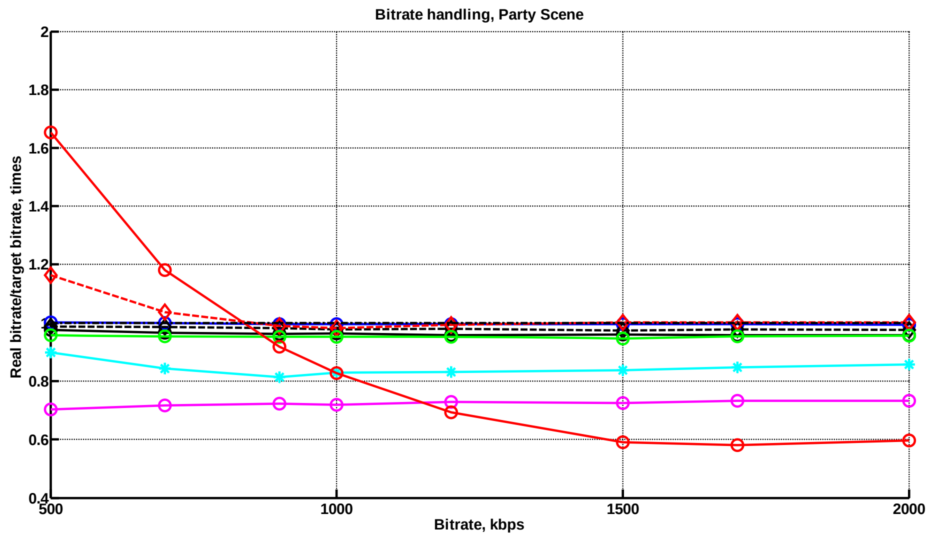


Figure 69. Bitrate handling—usage area "Movies," "Party Scene" sequence, Normal preset

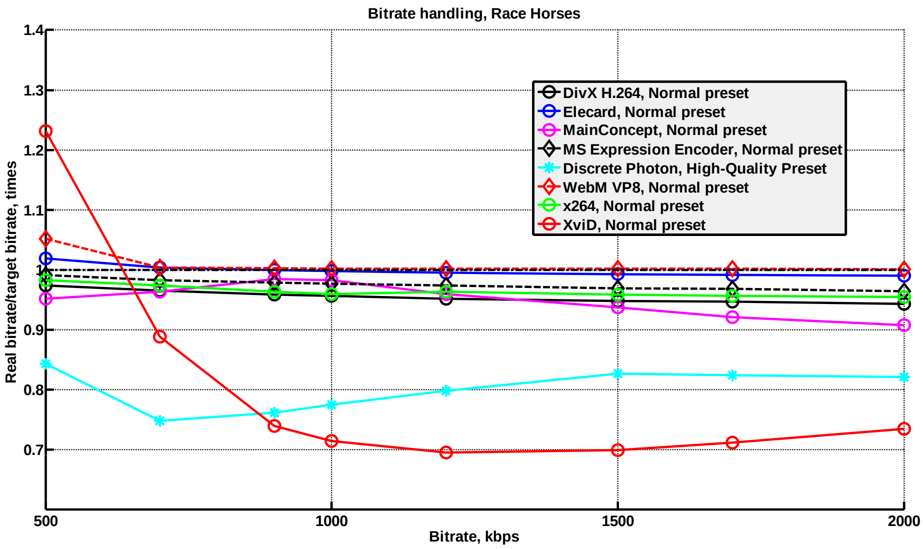


Figure 70. Bitrate handling—usage area "Movies," "Race Horses" sequence, Normal preset

4.2.4.3 High Quality Preset

The results are quite close to HighSpeed and Normal presets: all encoders show good bitrate handling mechanisms except XviD, with some issues for MainConcept and DiscretePhoton.

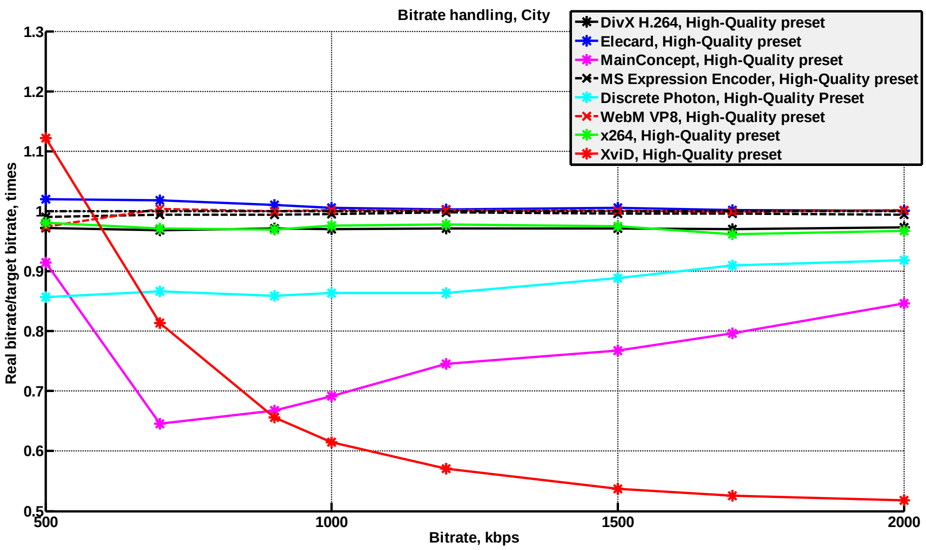


Figure 71. Bitrate handling—usage area "Movies," "City" sequence, High Quality preset

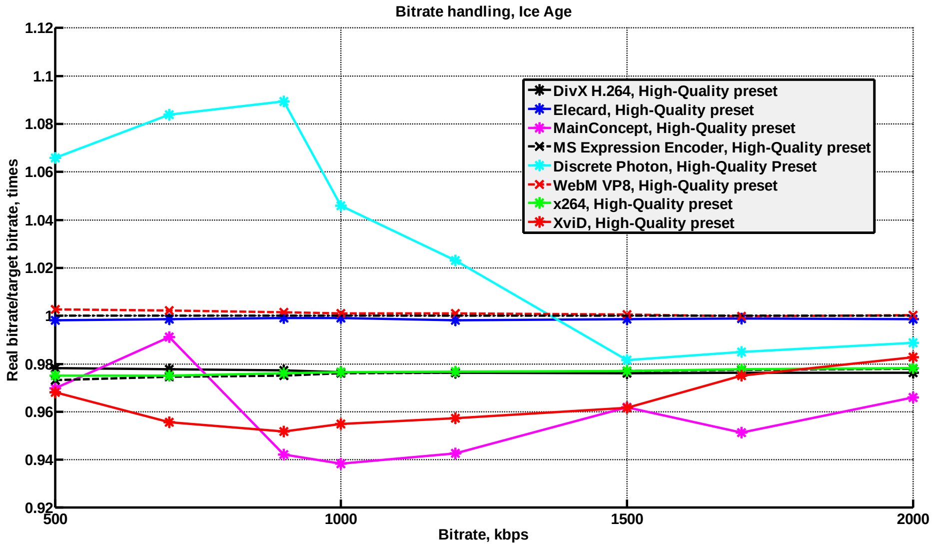


Figure 72. Bitrate handling—usage area "Movies," "Ice Age" sequence, High Quality preset

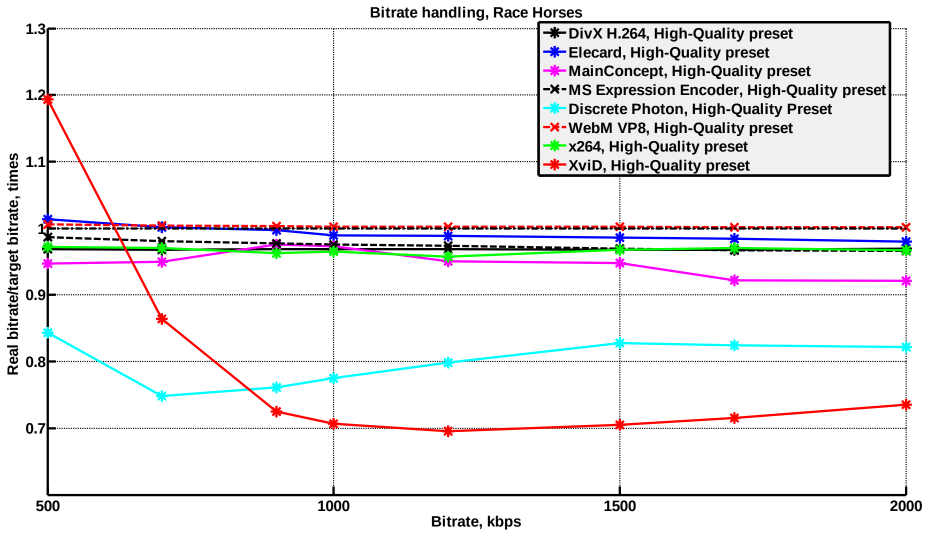


Figure 73. Bitrate handling—usage area "Movies," "Race Horses" sequence, High Quality preset

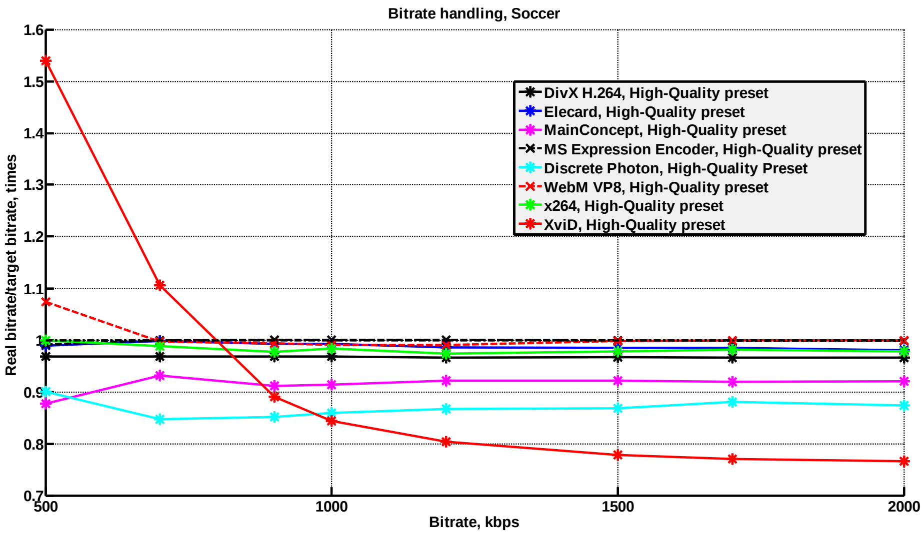


Figure 74. Bitrate handling—usage area "Movies," "Soccer" sequence, High Quality preset

4.2.5 Relative Quality Analysis

Table 5 through Table 10 show relative bitrates for a fixed quality output for all codecs and presets. Note that these tables do not include information about the speed of the encoder.

Note that each number in the tables below corresponds to some range of bitrates (see Appendix 7. Figures Explanation for more details). 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.

Consider the High Speed preset (Y-SSIM results in Table 5 and Y-PSNR results in Table 6). On average, the leader is the x264 encoder and MainConcept with DivX H.264 encoders are second (MainConcept is better than DivX H.264).

Table 7 and Table 8 present the Normal preset results for the Y-SSIM and Y-PSNR quality metrics, respectively. The results are similar to those of the High Speed preset: the leader is the x264 encoder and MainConcept with DivX H.264 encoders are second (DivX H.264 is better than MainConcept).

Table 9 and Table 10 present the High Quality preset results for the Y-SSIM and Y-PSNR quality metrics, respectively. The results are very similar to those of the Normal preset: the leader is the x264 encoder and MainConcept with DivX H.264 encoders are second (DivX H.264 is better than MainConcept).

Table 5. Average bitrate ratio for the same quality. Usage area "Movie". "High Speed" preset, Y-SSIM.

	DivX H.264	Elecard	MainConcept	Discrete Photon	x264	XviD
DivX H.264	100%	88%	78%	139%	70%	142%
Elecard	114%	100%	89%	161%	80%	161%
MainConcept	128%	112%	100%	178%	90%	181%
Discrete Photon	72%	62%	56%	100%	50%	102%
x264	142%	125%	111%	198%	100%	198%
XviD	70%	62%	55%	98%	50%	100%

Table 6. Average bitrate ratio for the same quality. Usage area "Movie". "High Speed" preset, Y-PSNR.

	DivX H.264	Elecard	MainConcept	Discrete Photon	x264	XviD
DivX H.264	100%	91%	105%	141%	84%	141%
Elecard	110%	100%	118%	157%	92%	155%
MainConcept	95%	85%	100%	140%	76%	136%
Discrete Photon	71%	64%	72%	100%	59%	101%
x264	119%	109%	131%	170%	100%	167%
XviD	71%	65%	73%	99%	60%	100%

Table 7. Average bitrate ratio for the same quality. Usage area "Movie". "Normal" preset, Y-SSIM.

	DivX H.264	Elecard	MainConcept	Discrete Photon	x264	XviD
DivX H.264	100%	111%	98%	180%	82%	147%
Elecard	90%	100%	89%	161%	74%	133%
MainConcept	102%	113%	100%	181%	84%	150%
Discrete Photon	56%	62%	55%	100%	45%	84%
x264	122%	135%	119%	220%	100%	163%
XviD	68%	75%	67%	119%	61%	100%

Table 8. Average bitrate ratio for the same quality. Usage area “Movie”. “Normal” preset, Y-PSNR.

	DivX H.264	Elecard	MainConcept	Discrete Photon	x264	XviD
DivX H.264	100%	113%	134%	177%	94%	141%
Elecard	88%	100%	118%	157%	83%	126%
MainConcept	75%	85%	100%	142%	77%	112%
Discrete Photon	56%	64%	71%	100%	52%	81%
x264	106%	120%	130%	192%	100%	152%
XviD	71%	80%	89%	123%	66%	100%

Table 9. Average bitrate ratio for the same quality. Usage area “Movie”. “High Quality” preset, Y-SSIM.

	DivX H.264	Elecard	MainConcept	Discrete Photon	x264	XviD
DivX H.264	100%	111%	95%	189%	76%	144%
Elecard	90%	100%	86%	168%	69%	130%
MainConcept	105%	116%	100%	197%	81%	151%
Discrete Photon	53%	60%	51%	100%	42%	80%
x264	131%	144%	124%	237%	100%	171%
XviD	69%	77%	66%	125%	58%	100%

Table 10. Average bitrate ratio for the same quality. Usage area “Movie”. “High Quality” preset, Y-PSNR.

	DivX H.264	Elecard	MainConcept	Discrete Photon	x264	XviD
DivX H.264	100%	112%	136%	191%	91%	140%
Elecard	89%	100%	120%	169%	81%	125%
MainConcept	73%	83%	100%	152%	73%	110%
Discrete Photon	52%	59%	66%	100%	47%	76%
x264	110%	123%	136%	214%	100%	155%
XviD	72%	80%	91%	132%	65%	100%

Figure 75 through Figure 80 depict the data from the tables above. Each line in the figures 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.

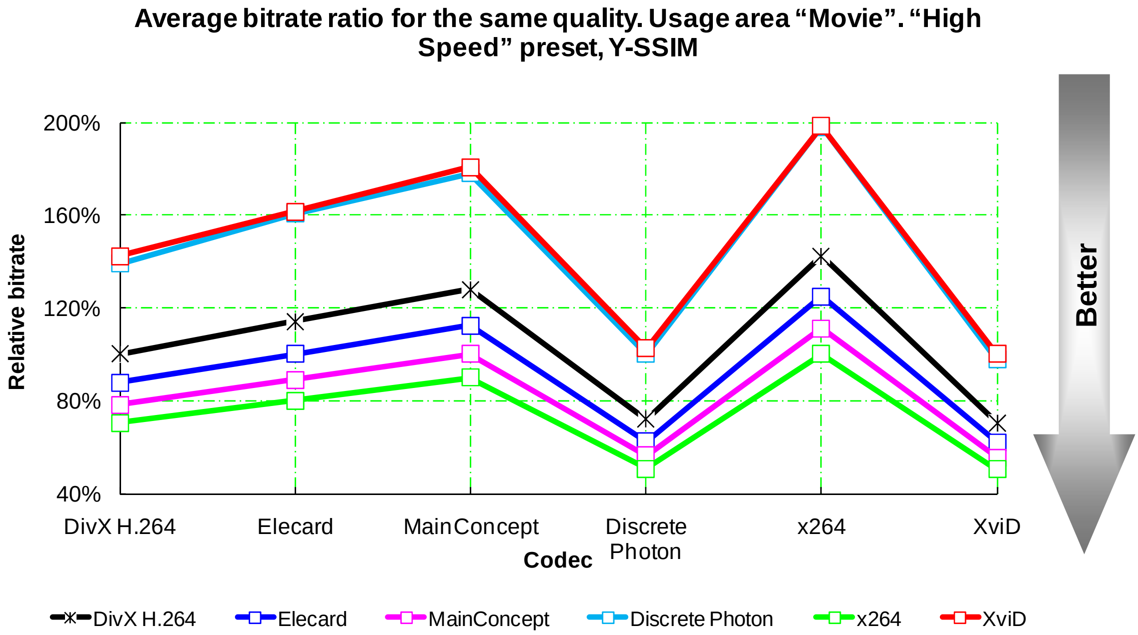


Figure 75. **Average bitrate ratio for a fixed quality—usage area “Movies,” High Speed preset, Y-SSIM metric**

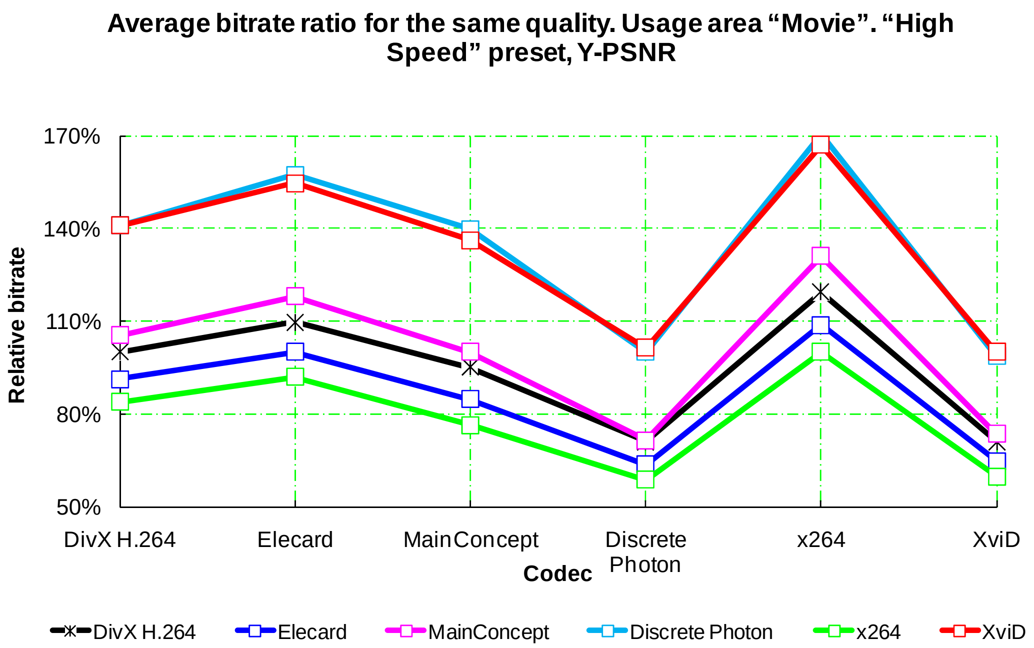


Figure 76. **Average bitrate ratio for a fixed quality—usage area “Movies”. High Speed preset, Y-PSNR metric.**

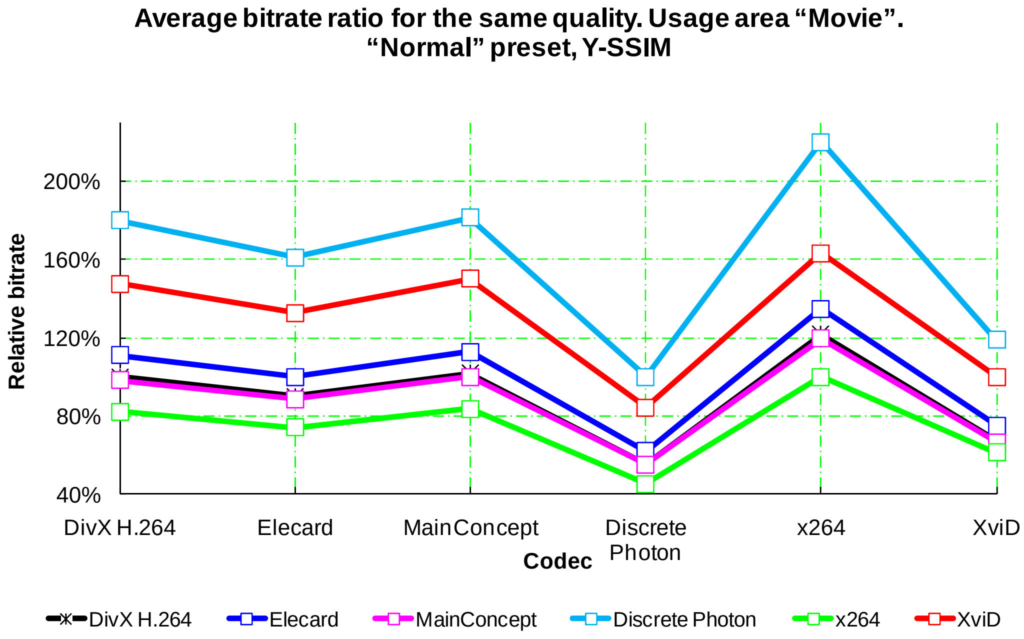


Figure 77. Average bitrate ratio for a fixed quality—usage area "Movies". Normal preset, Y-SSIM metric.

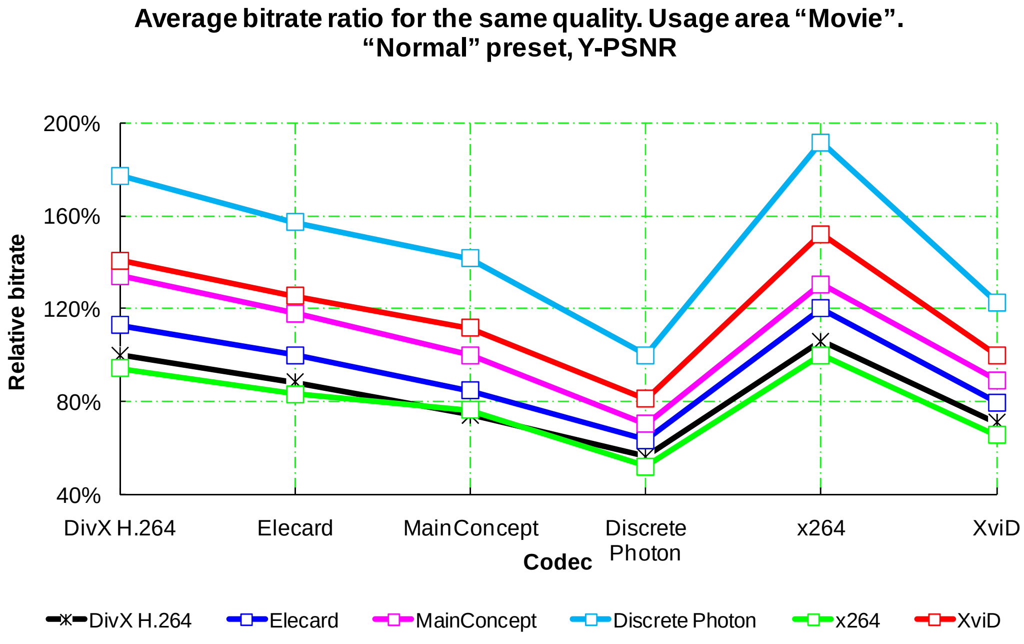


Figure 78. Average bitrate ratio for a fixed quality—usage area "Movies". Normal preset, Y-PSNR metric.

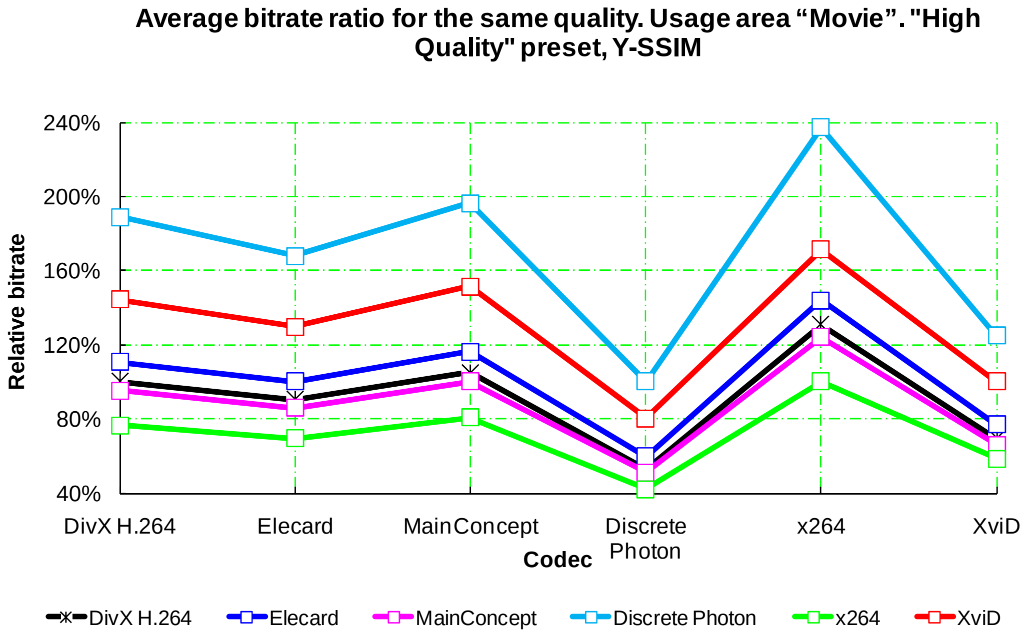


Figure 79. **Average bitrate ratio for a fixed quality—usage area “Movies”. High Speed preset, Y-SSIM metric.**

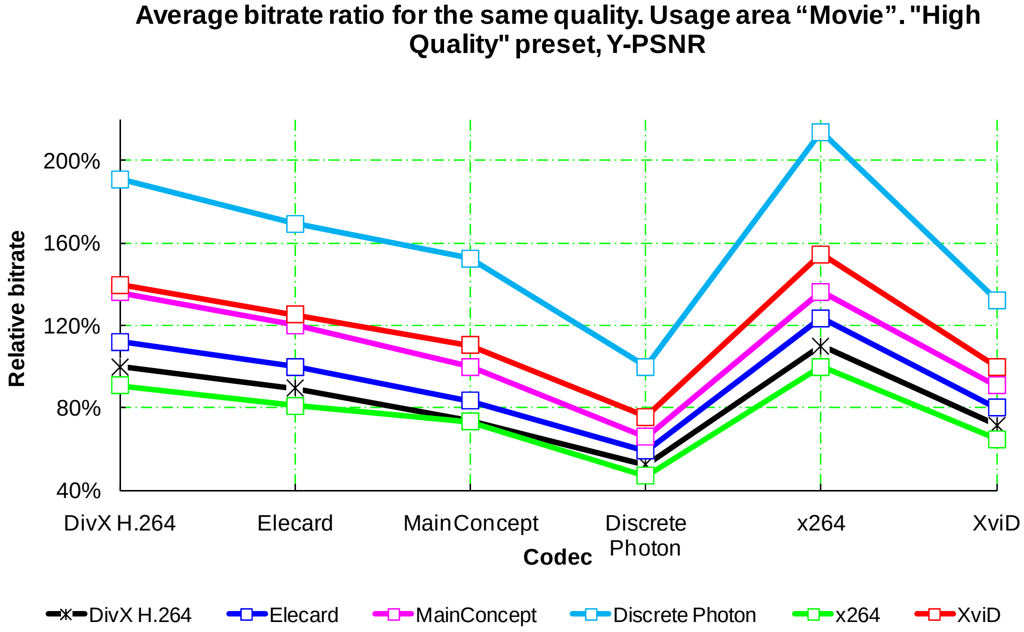


Figure 80. **Average bitrate ratio for a fixed quality—usage area “Movies”. High Quality preset, Y-PSNR metric.**

4.3 HDTV

4.3.1 RD Curves

4.3.1.1 High Speed Preset

The High Speed preset results for each sequence are presented in Figure 81 through Figure 87. The first six figures show the Y-SSIM results, and the last figure shows the Y-PSNR results. The leader is x264 followed by DivX H.264 and Elecard encoders at average but there are some video sequences (for example, Riverbed), where the situation changes strongly – the leader is Elecard and x264 shows lower quality than even XviD. PSNR metric usage changes the results: for some sequences Elecard shows better quality and x264 shows lower quality.

MSE and WebM High Speed presets do not fit the encoding speed requirements.

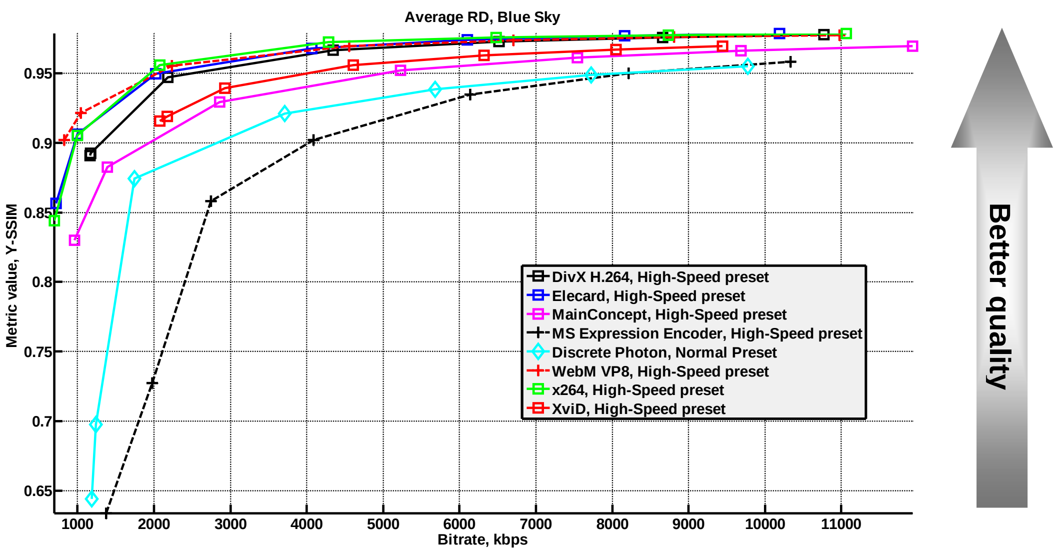


Figure 81. Bitrate/quality—usage area "HDTV," "Blue sky" sequence, High Speed preset, Y-SSIM metric

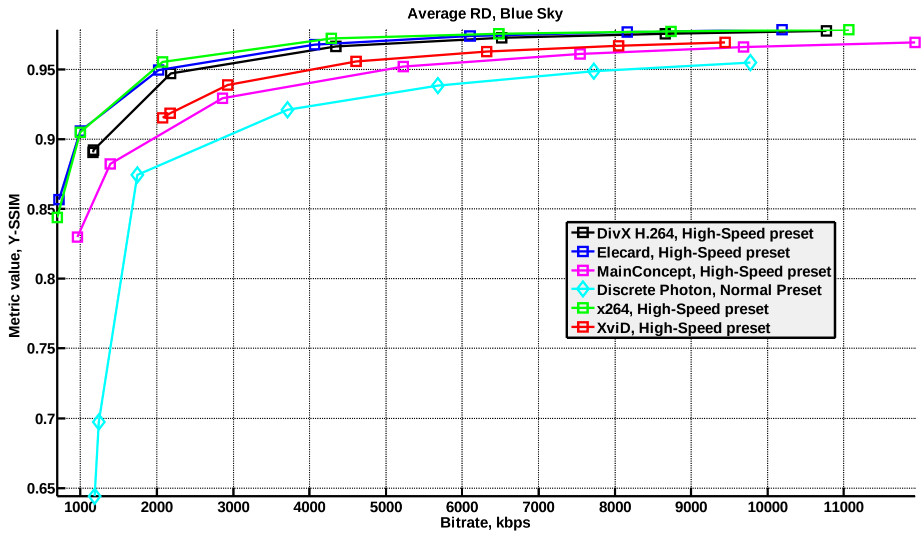


Figure 82. Bitrate/quality—usage area "HDTV," "Blue Sky" sequence, High Speed preset, Y-SSIM metric. Encoders that fit encoding speed requirements

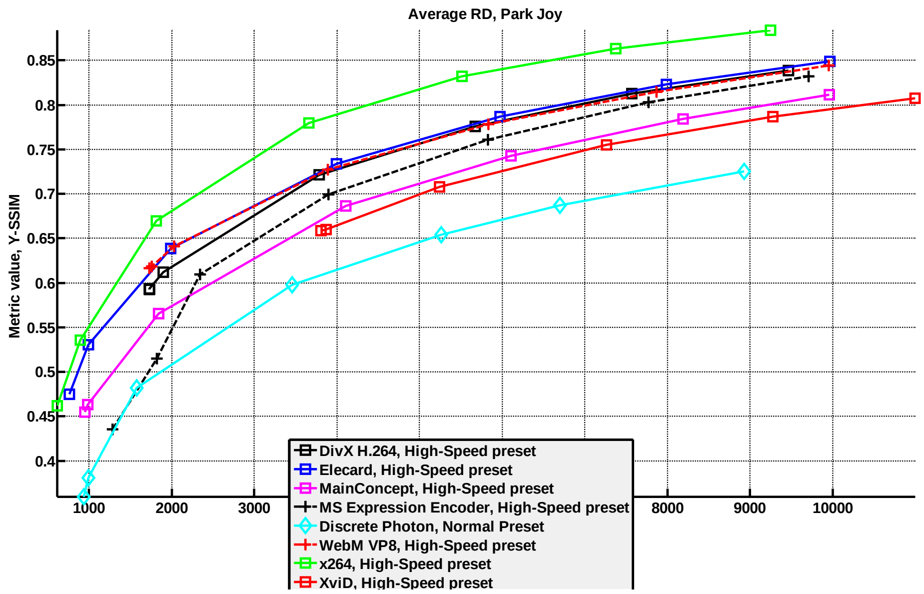


Figure 83. Bitrate/quality—usage area "HDTV," "Park Joy" sequence, High Speed preset, Y-SSIM metric

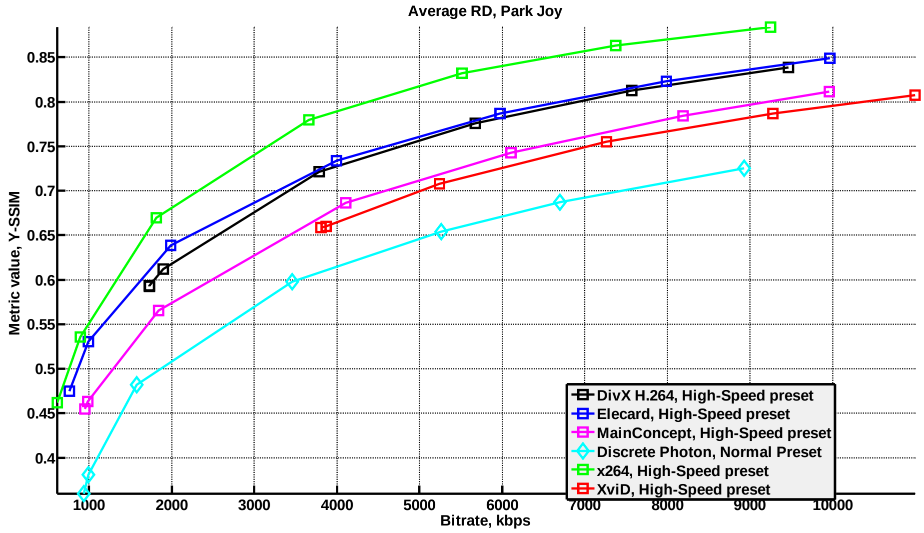


Figure 84. Bitrate/quality—usage area "HDTV," "Park Joy" sequence, High Speed preset, Y-SSIM metric. Encoders that fit encoding speed requirements

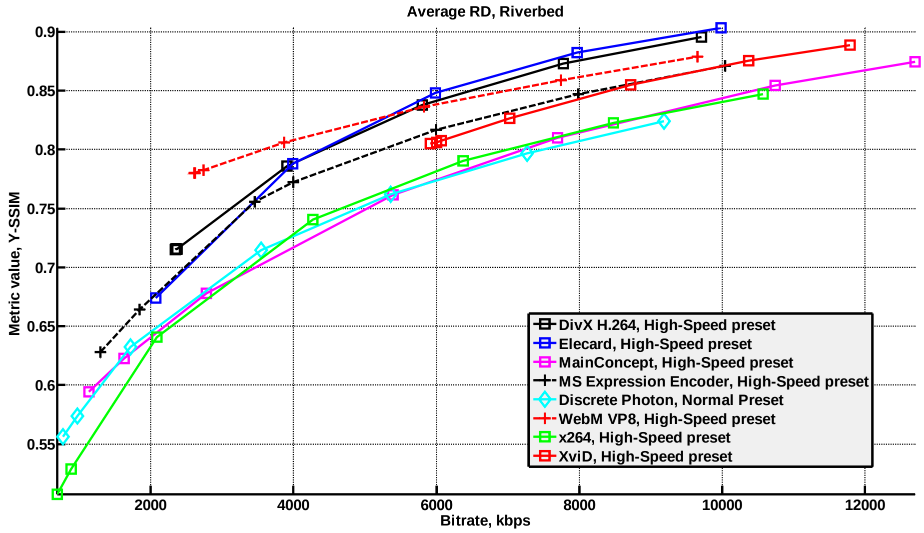


Figure 85. Bitrate/quality—usage area "HDTV," "Riverbed" sequence, High Speed preset, Y-SSIM metric

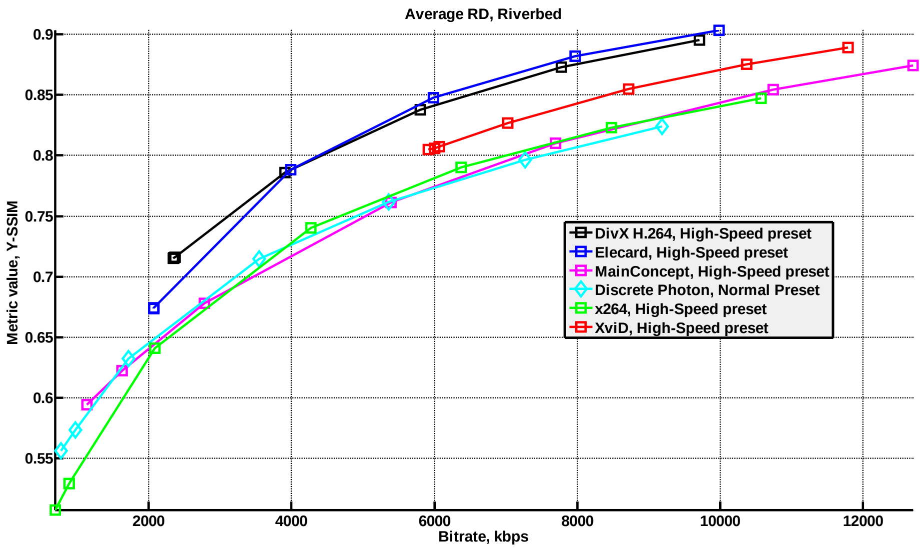


Figure 86. Bitrate/quality—usage area "HDTV," "Riverbed" sequence, High Speed preset, Y-SSIM metric. Encoders that fit encoding speed requirements

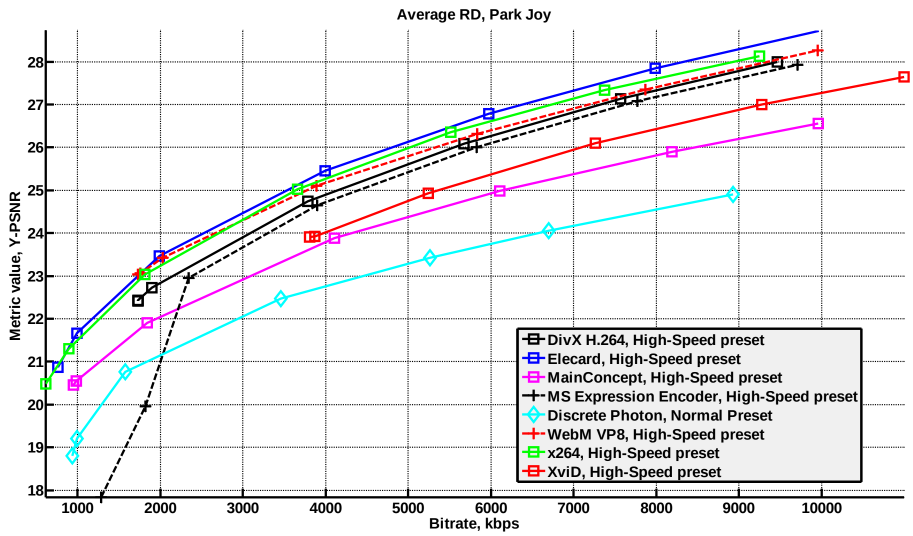


Figure 87. Bitrate/quality—usage area "HDTV," "Park Joy" sequence, High Speed preset, Y-PSNR metric

4.3.1.2 Normal Preset

The Normal preset results for each sequence are presented in Figure 88 through Figure 94. The first six figures show the Y-SSIM results, and last figure show the Y-PSNR results.

The situation is close to High Speed preset, but MainConcept shows good results and is among the leaders for this preset. x264 is the leader by quality at average.

MSE and WebM Normal presets do not fit the encoding speed requirements.

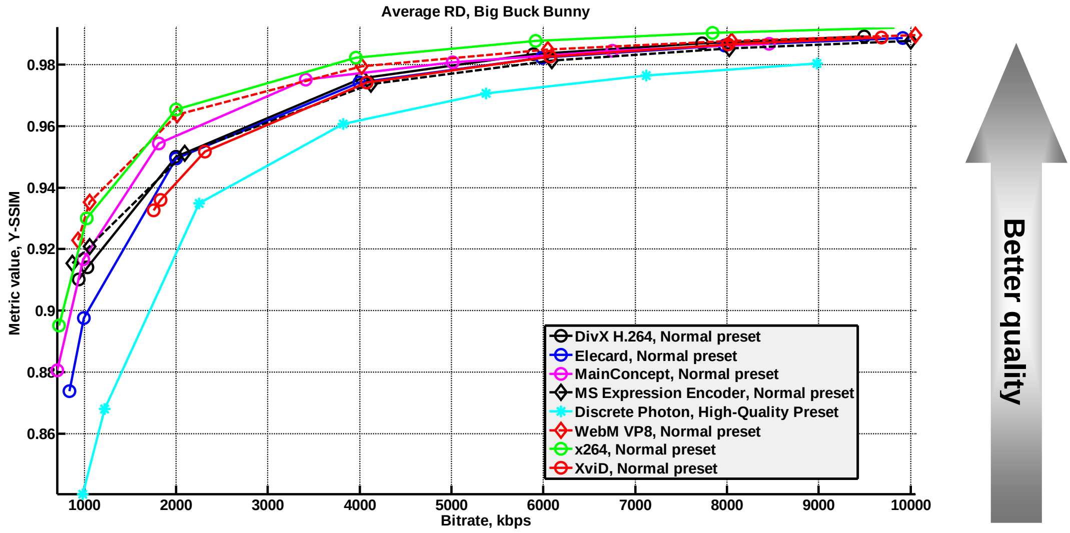


Figure 88. Bitrate/quality—usage area "HDTV," "Big Buck Bunny" sequence, Normal preset, Y-SSIM metric

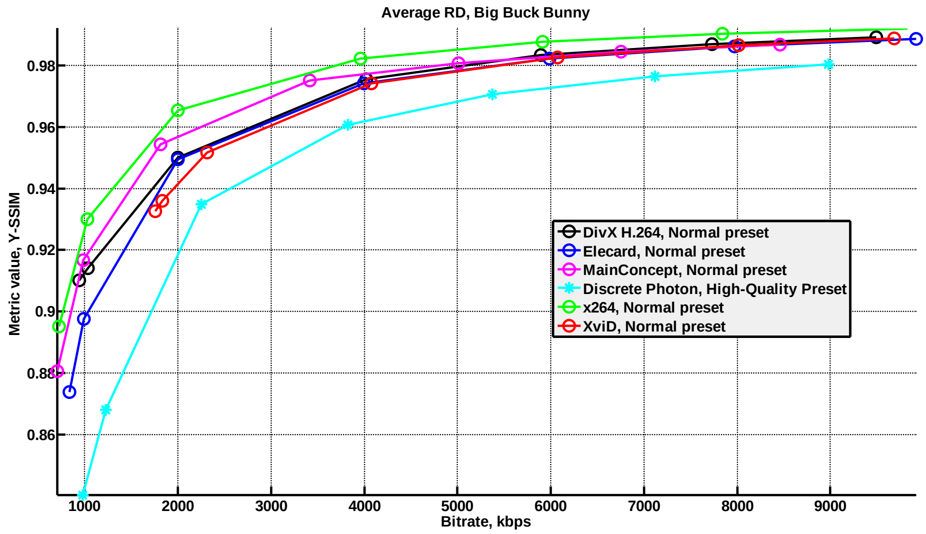


Figure 89. Bitrate/quality—usage area "HDTV," "Big Buck Bunny" sequence, Normal preset, Y-SSIM metric. Encoders that fit encoding speed requirements

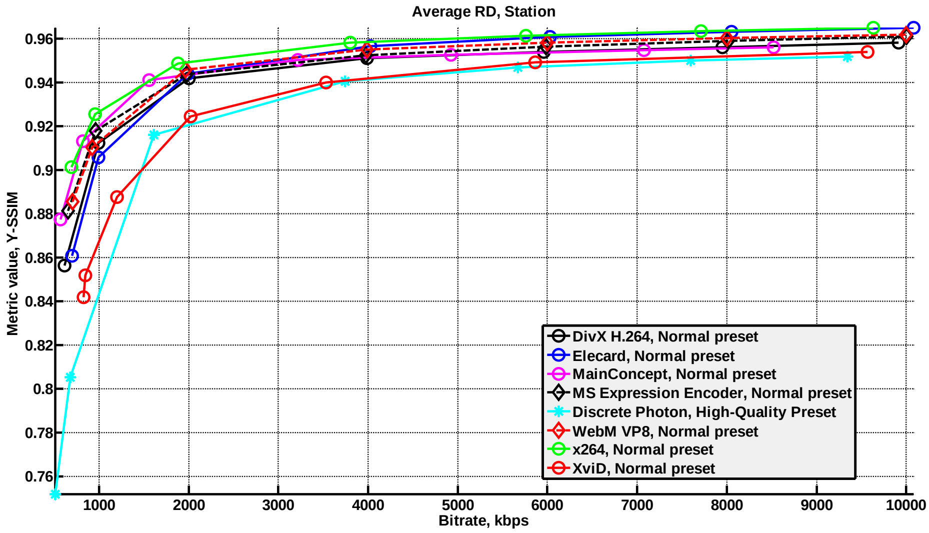


Figure 90. Bitrate/quality—usage area "HDTV," "Station" sequence, Normal preset, Y-SSIM metric

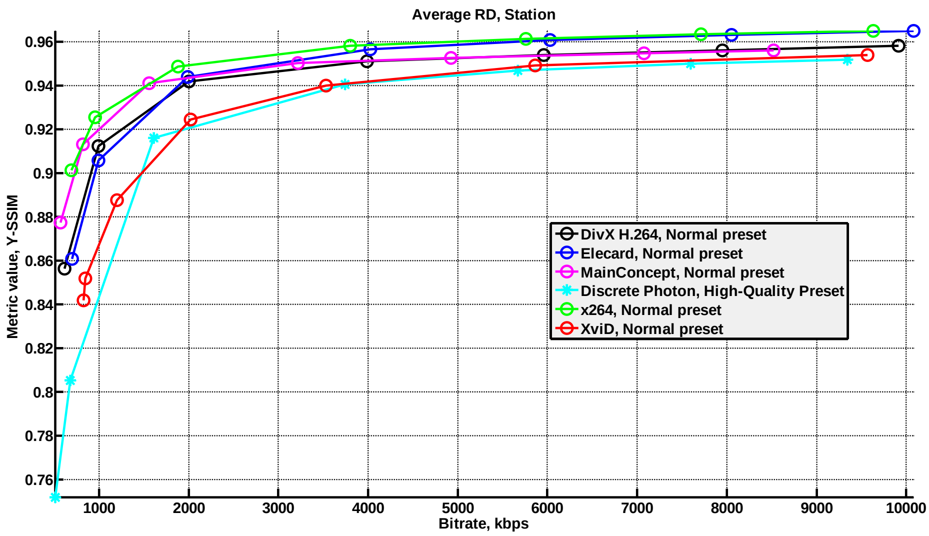


Figure 91. Bitrate/quality—usage area "HDTV," "Station" sequence, Normal preset, Y-SSIM metric. Encoders that fit encoding speed requirements

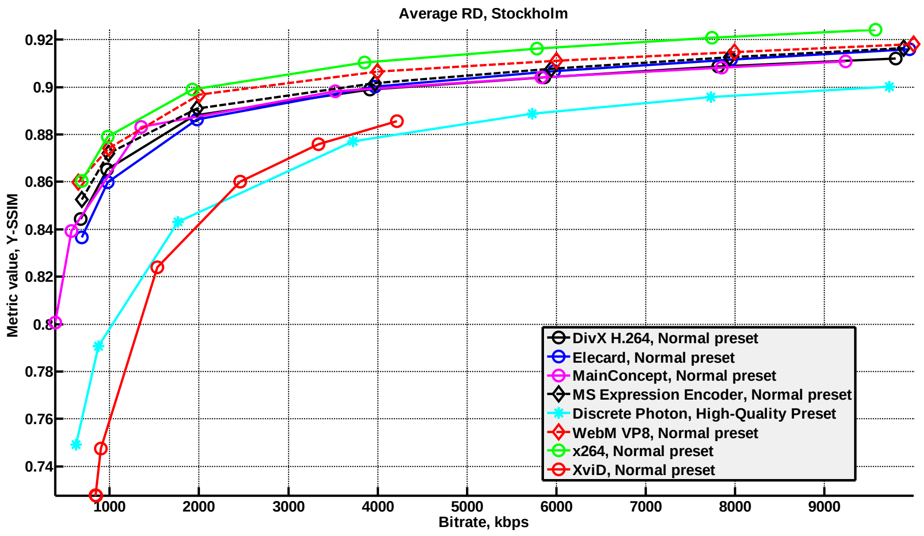


Figure 92. Bitrate/quality—usage area "HDTV," "Stockholm" sequence, Normal preset, Y-SSIM metric

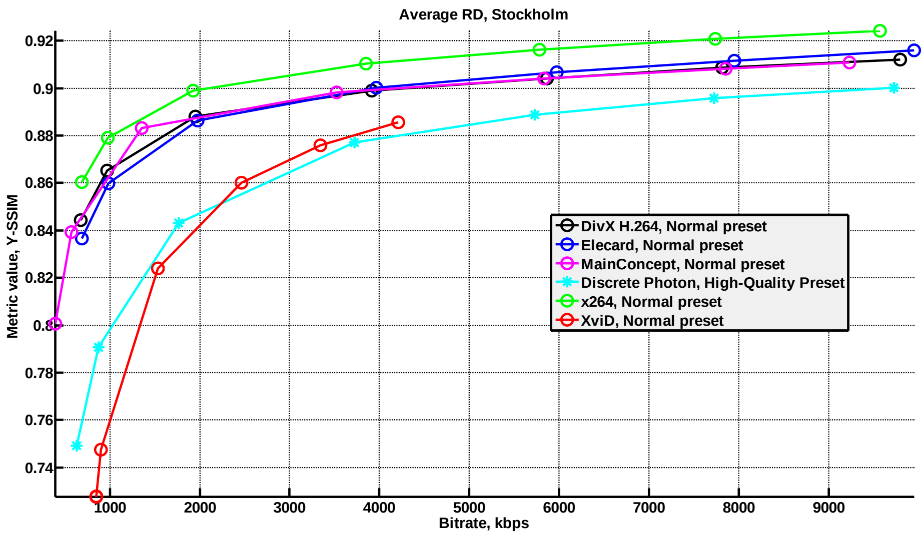


Figure 93. Bitrate/quality—usage area "HDTV," "Stockholm" sequence, Normal preset, Y-SSIM metric. Encoders that fit encoding speed requirements

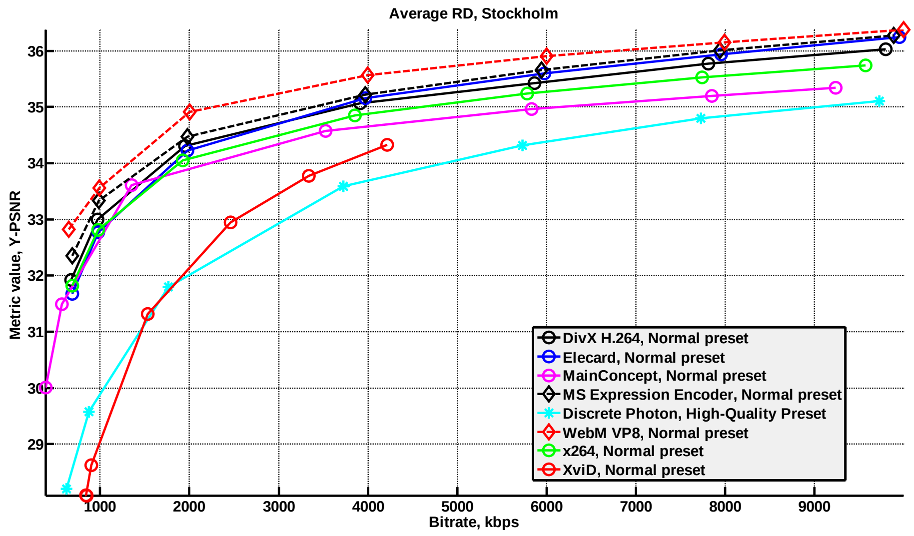


Figure 94. Bitrate/quality—usage area "HDTV," "Stockholm" sequence, Normal preset, Y-PSNR metric

4.3.1.3 High Quality Preset

The High Quality preset results for each sequence are presented in Figure 95 through Figure 101. The leader in this use case is x264. Three encoders are the second: MainConcept, DivX H.264 and Elecard. The DiscretePhoton encoder demonstrates the poorest results as for High Speed and Normal presets.

WebM HighQuality preset does not fit encoding speed requirements.

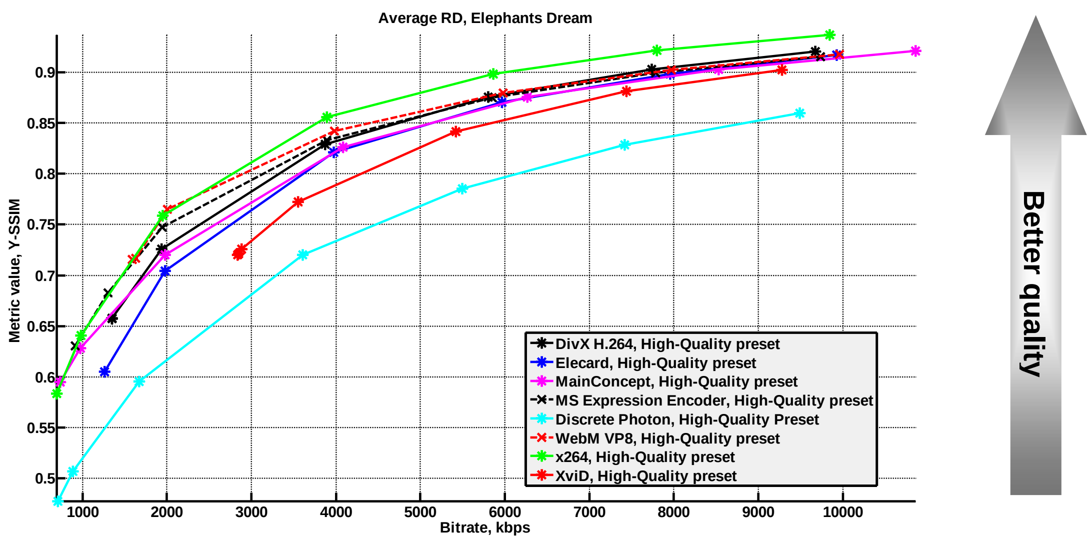


Figure 95. Bitrate/quality—usage area "HDTV," "Elephants Dream" sequence, High Quality preset, Y-SSIM metric

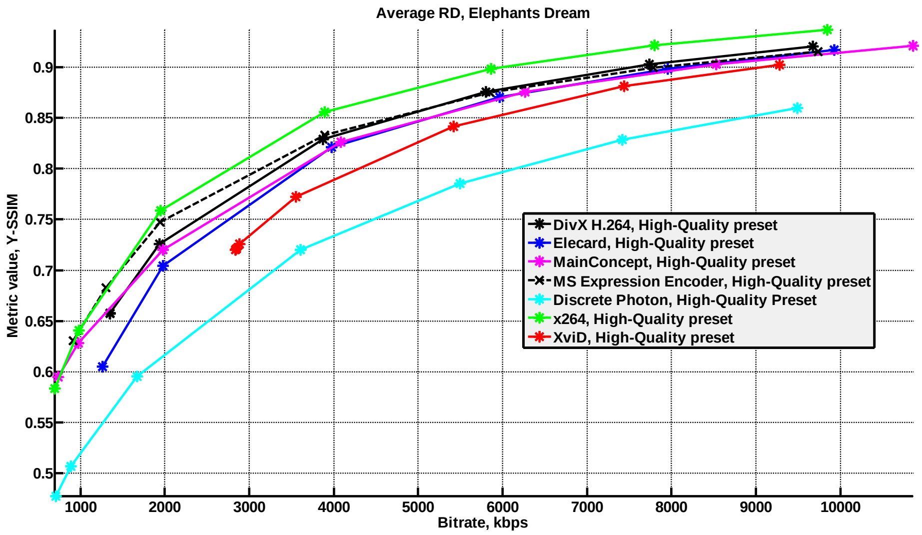


Figure 96. Bitrate/quality—usage area "HDTV," "Elephants Dream" sequence, High Quality preset, Y-SSIM metric. Encoders that fit encoding speed requirements

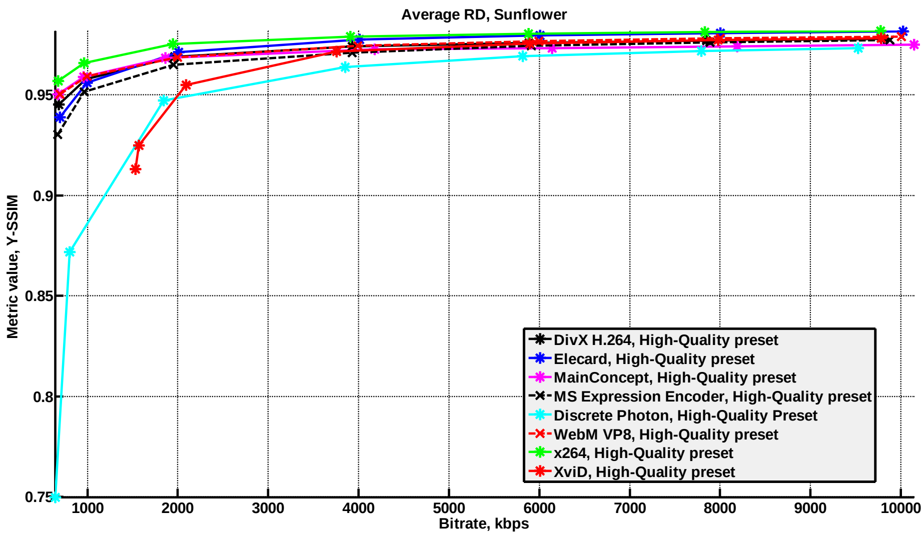


Figure 97. Bitrate/quality—usage area "HDTV," "Sunflower" sequence, High Quality preset, Y-SSIM metric.

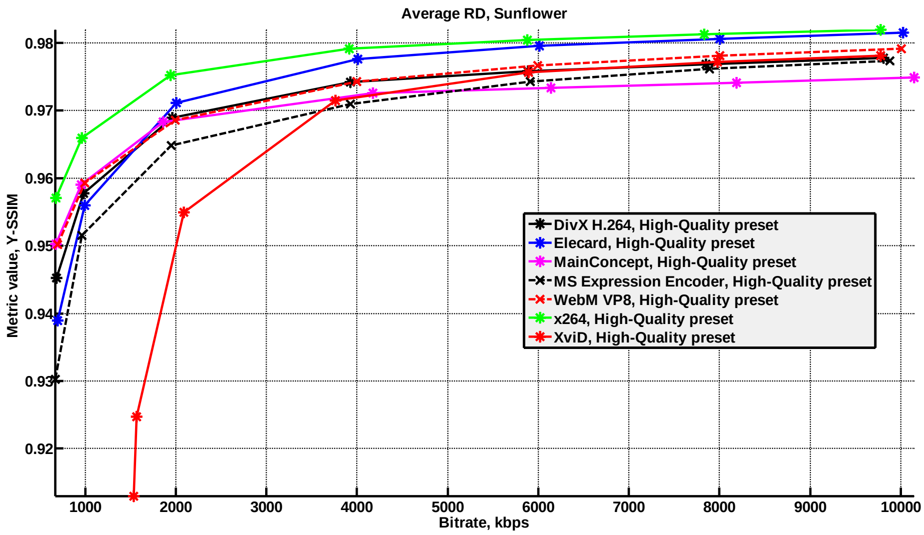


Figure 98. Bitrate/quality—usage area "HDTV," "Sunflower" sequence, High Quality preset, Y-SSIM metric. All encoders except Discrete Photon (to improve readability of the graph)

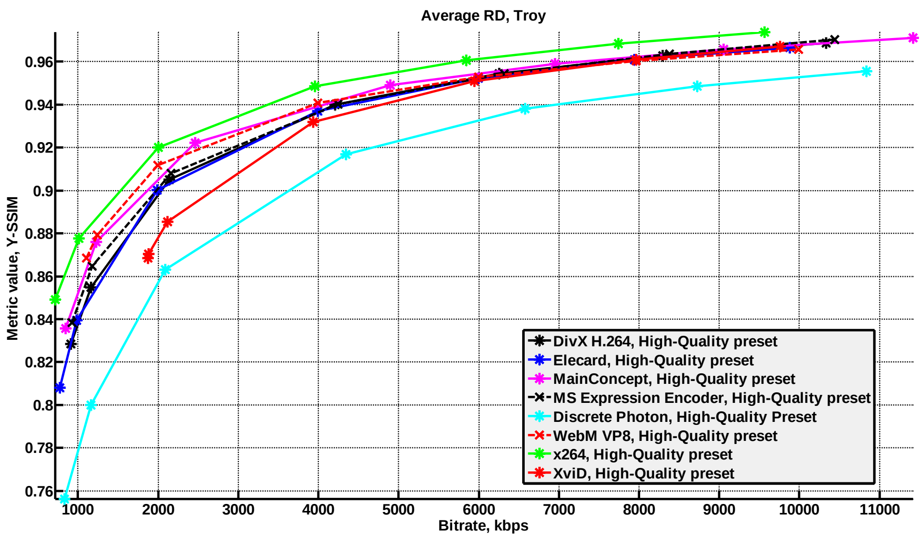


Figure 99. Bitrate/quality—usage area "HDTV," "Troy" sequence, High Quality preset, Y-SSIM metric

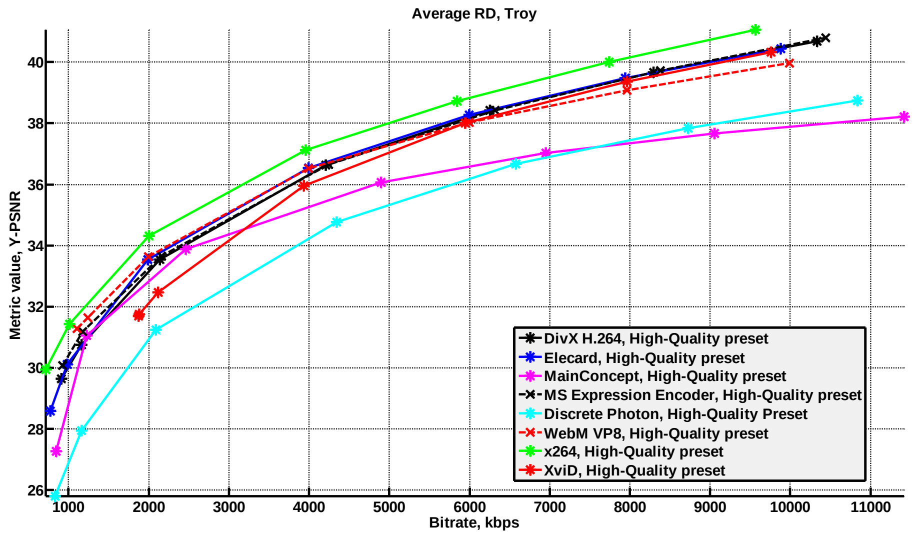


Figure 100. Bitrate/quality—usage area "HDTV," "Troy" sequence, High Quality preset, Y-PSNR metric

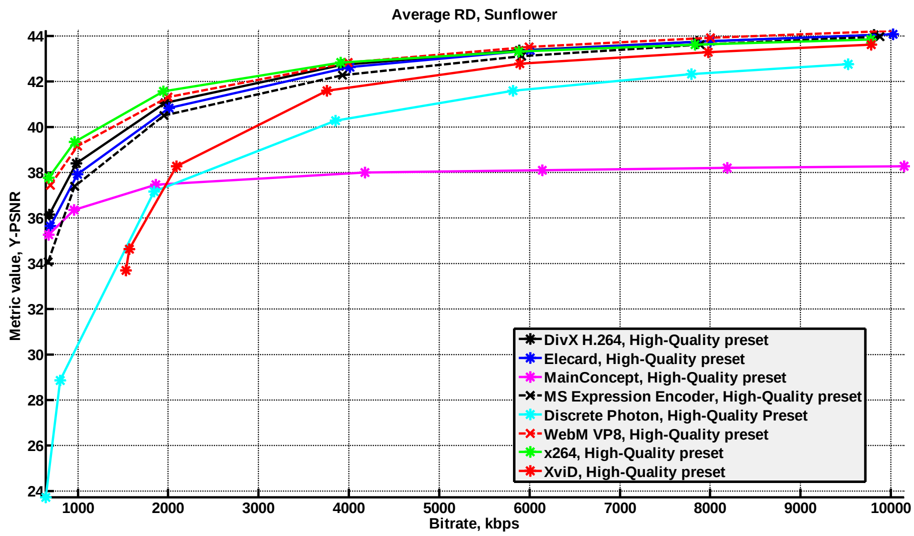


Figure 101. Bitrate/quality—usage area "HDTV," "Sunflower" sequence, High Quality preset, Y-PSNR metric

4.3.2 Encoding Speed

4.3.2.1 High Speed Preset

Absolute speed results are presented in Figure 102 through Figure 104. All the encoders, except Microsoft Expression for all sequences and Elecard for Riverbed sequence, have a similar growth rate for encoding time versus increasing bitrate.

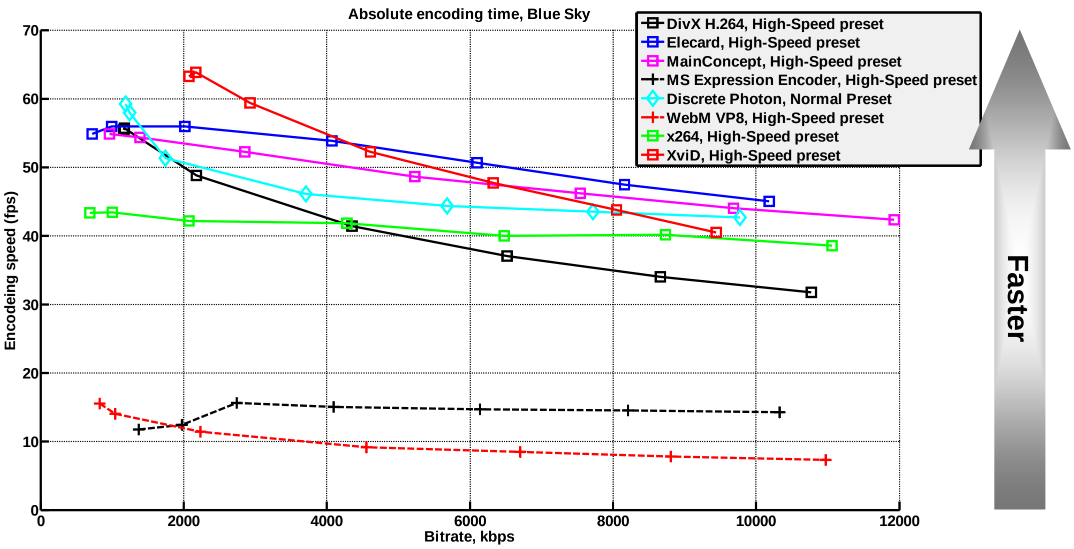


Figure 102. Encoding speed—usage area "HDTV," "Blue Sky" sequence, High Speed preset

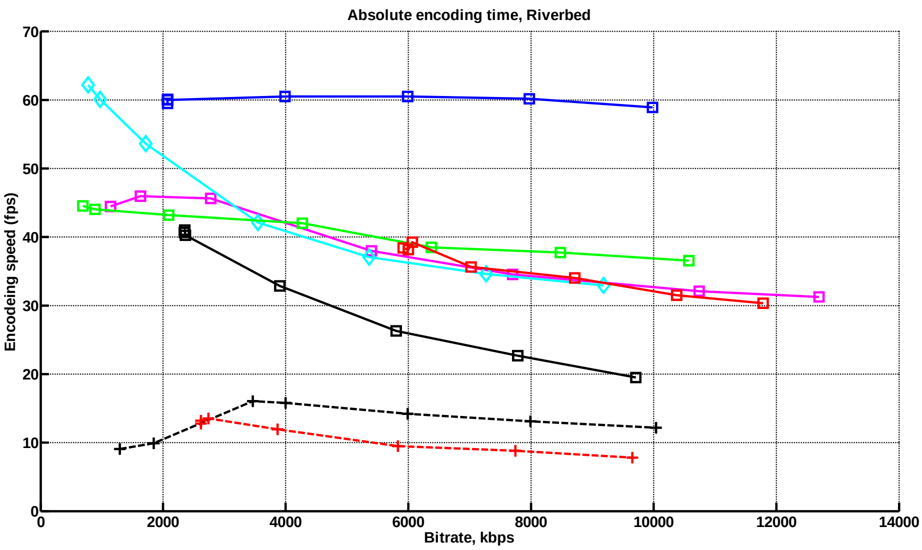


Figure 103. Encoding speed—usage area "HDTV," "Riverbed" sequence, High Speed preset

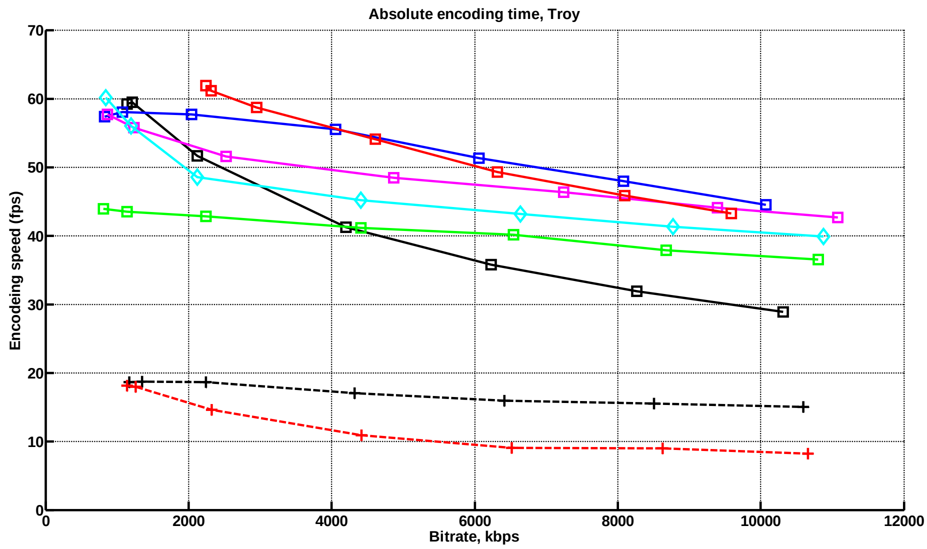


Figure 104. Encoding speed—usage area "HDTV," "Troy" sequence, High Speed preset

4.3.2.2 Normal Preset

Absolute speed results are presented in Figure 105 through Figure 108. All the encoders, have similar growth rate for encoding time versus increasing bitrate. Elecard and DiscretePhoton are the fastest.

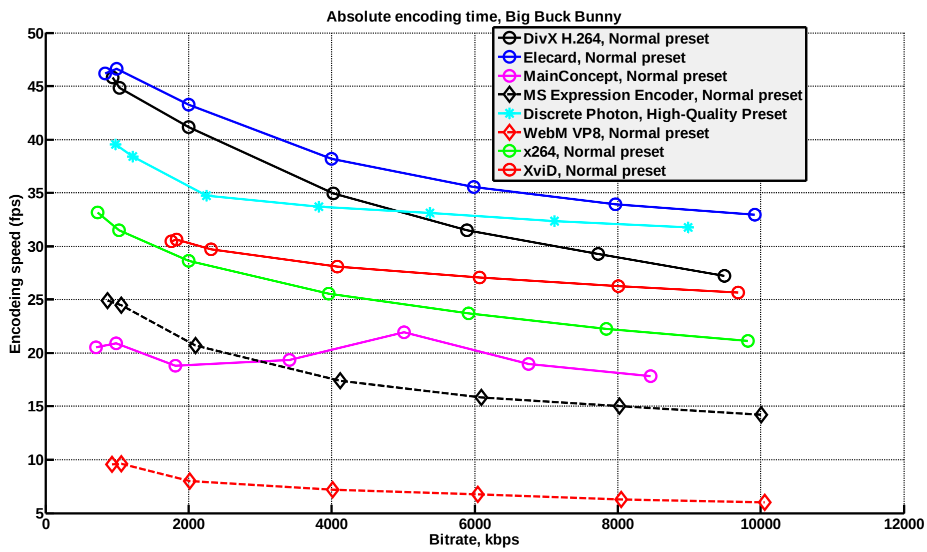


Figure 105. Encoding speed—usage area "HDTV," "Bick Buck Bunny" sequence, Normal preset

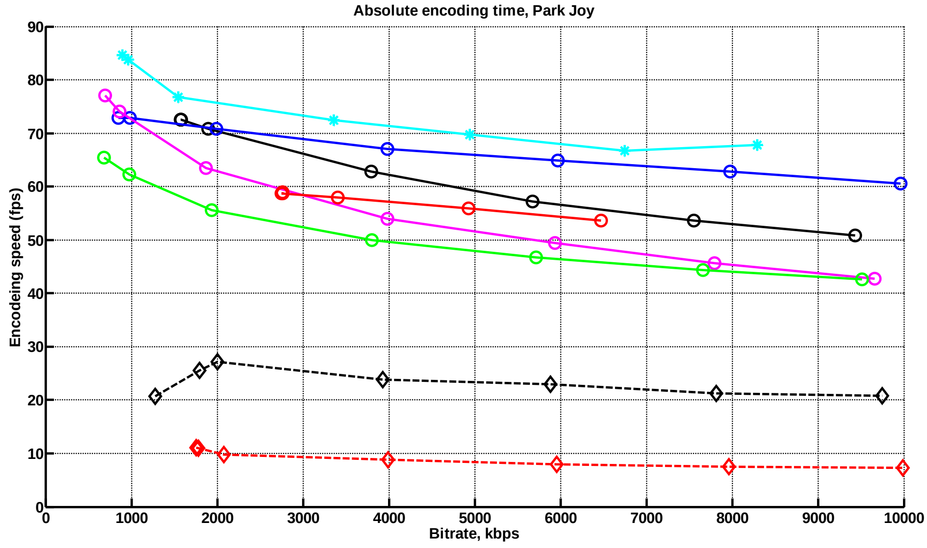


Figure 106. Encoding speed—usage area “HDTV,” “Park Joy” sequence, Normal preset

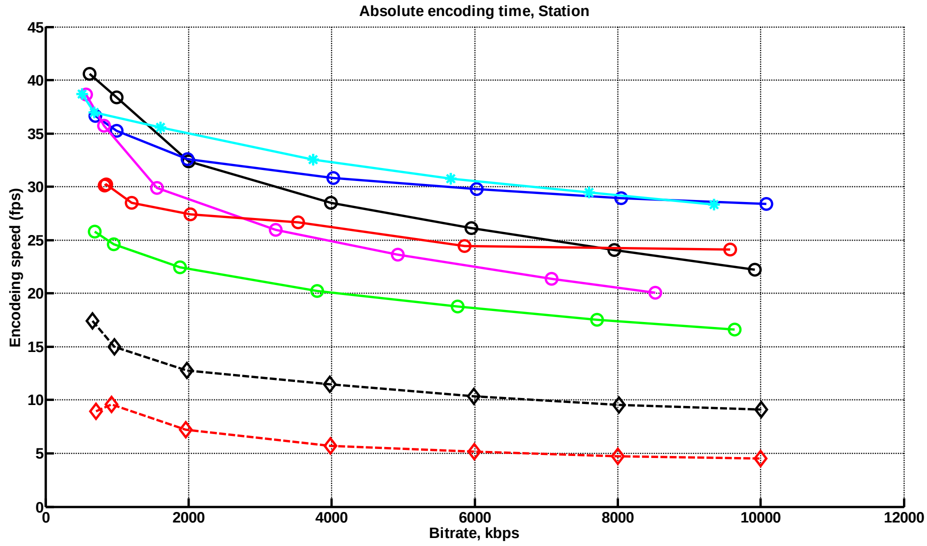


Figure 107. Encoding speed—usage area “HDTV,” “Station” sequence, Normal preset

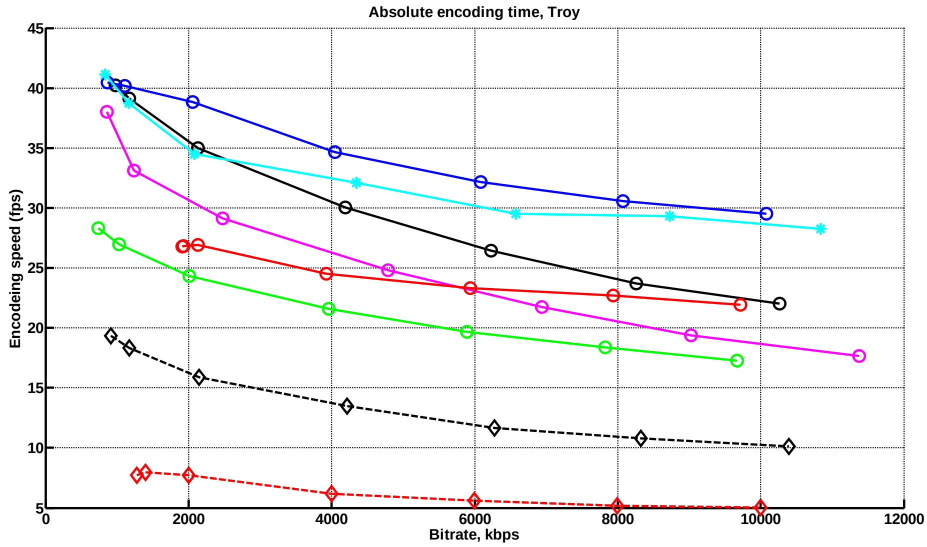


Figure 108. Encoding speed—usage area “HDTV,” “Troy” sequence, Normal preset

4.3.2.3 High Quality Preset

Absolute speed results are presented in Figure 109 through Figure 112. All the encoders, except Elecard at Riverbed sequence, have a similar growth rate for encoding time versus increasing bitrate. DiscretePhoton is the fastest.

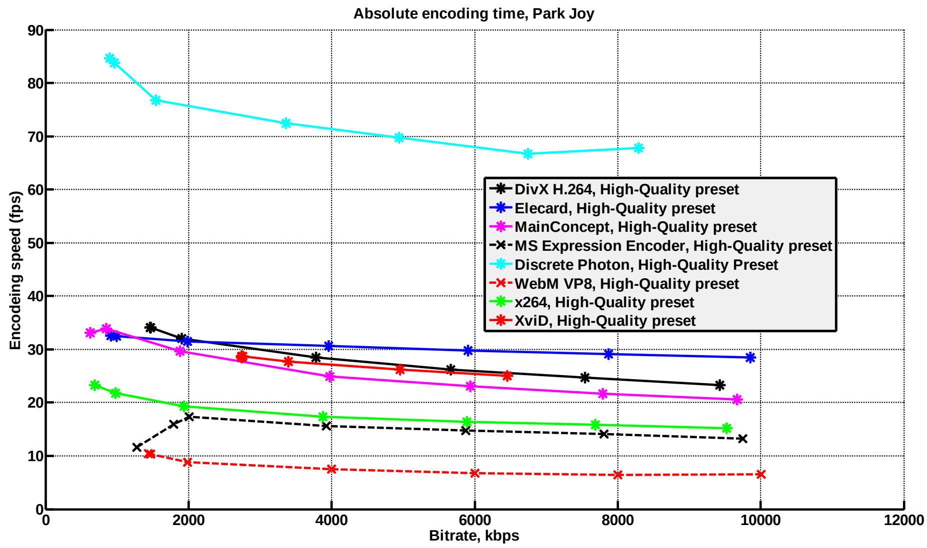


Figure 109. Encoding speed—usage area “HDTV,” “Park Joy” sequence, High Quality preset

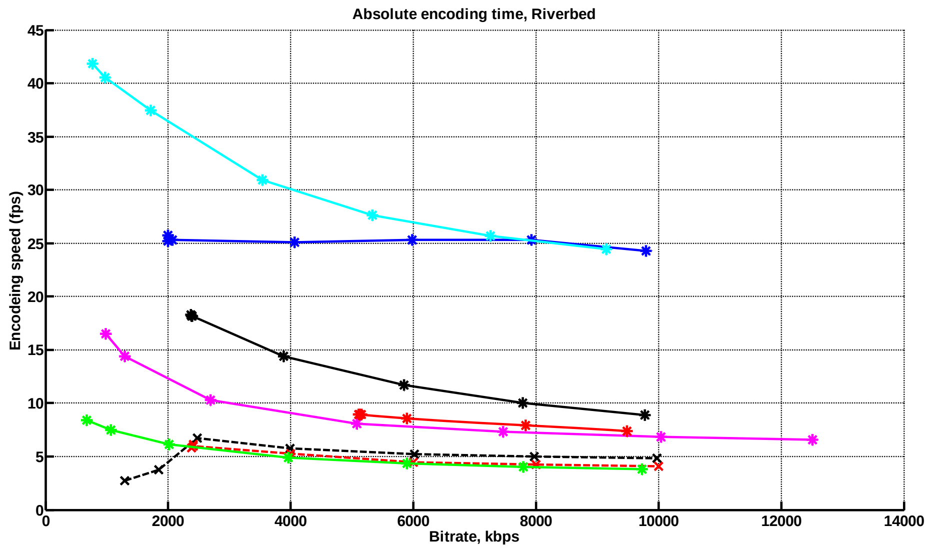


Figure 110. Encoding speed—usage area "HDTV," "Riverbed" sequence, High Quality preset

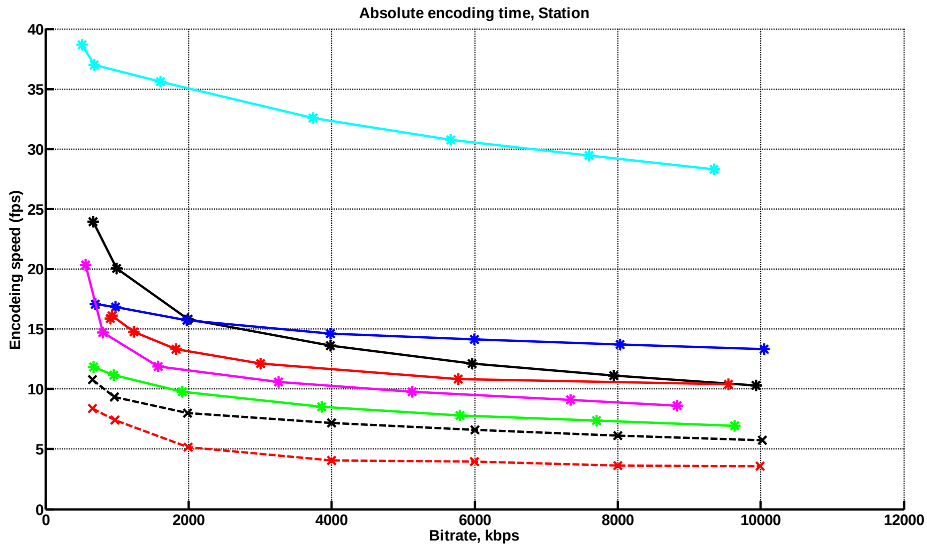


Figure 111. Encoding speed—usage area "HDTV," "Station" sequence, High Quality preset

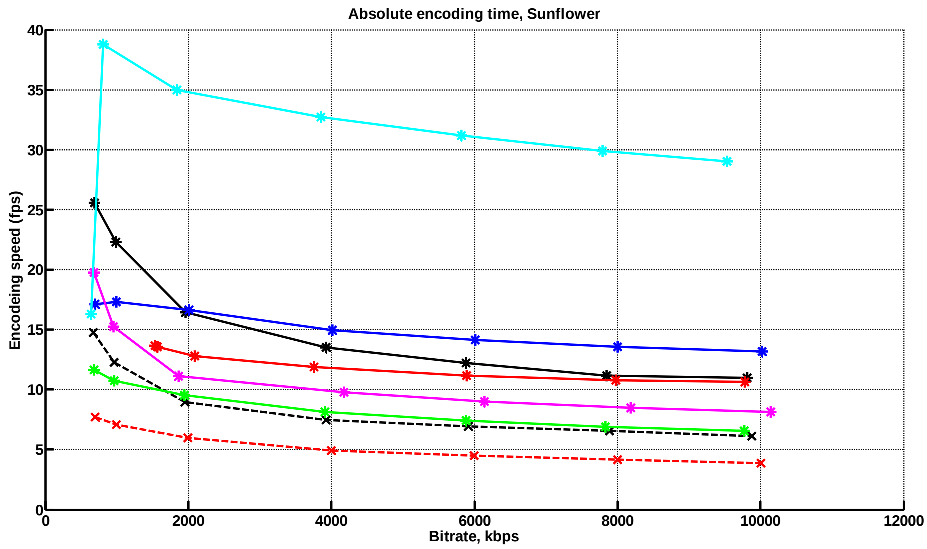


Figure 112. Encoding speed—usage area “HDTV,” “Sunflower” sequence, High Quality preset

4.3.3 Speed/Quality Trade-Off

Detailed descriptions of the speed/quality trade-off graphs can be found in Appendix 7. Figures Explanation. 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. XviD 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.

Please note that the method of averaging among all sequences assumes that all codecs produced results for each sequence. When this is not the case, only existing results are taken into account.

4.3.3.1 High Speed Preset

Figure 113 through Figure 119 show results for the High Speed preset. For the speed/quality trade-off using fast presets, the leaders are the x264 and Elecard encoders. For Y-PSNR metric only Elecard remains a leader.

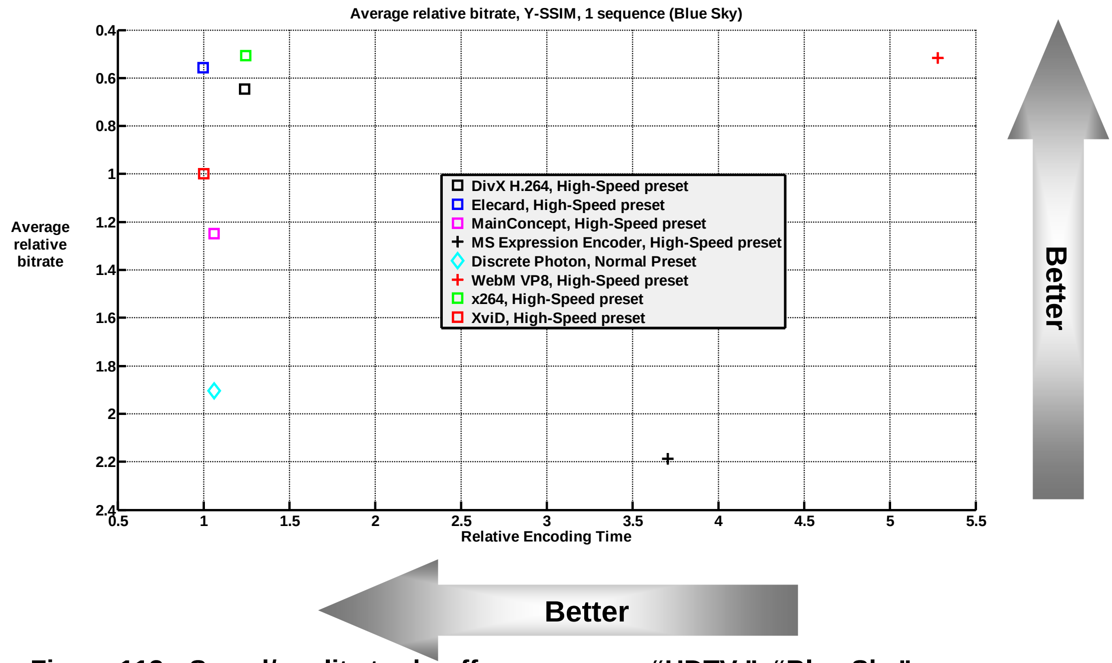


Figure 113. Speed/quality trade-off—usage area "HDTV," "Blue Sky" sequence, High Speed preset, Y-SSIM metric

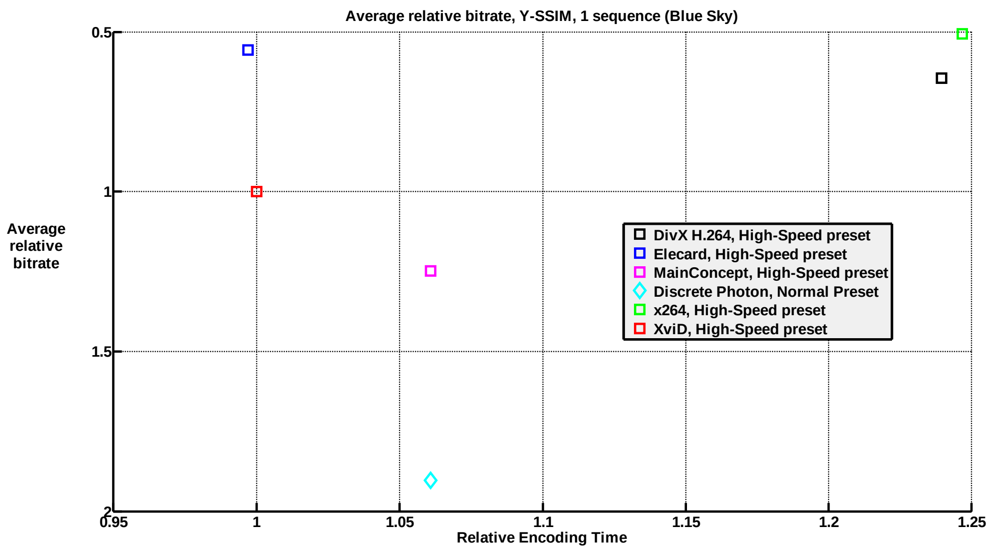


Figure 114. Speed/quality trade-off—usage area "HDTV," "Blue Sky" sequence, High Speed preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

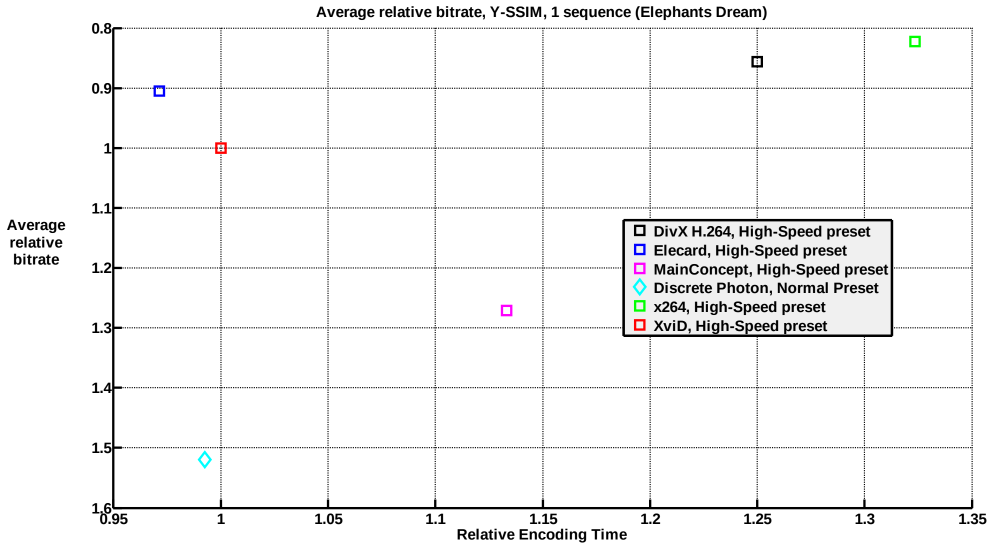


Figure 115. Speed/quality trade-off—usage area "HDTV," "Elephants Dream" sequence, High Speed preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

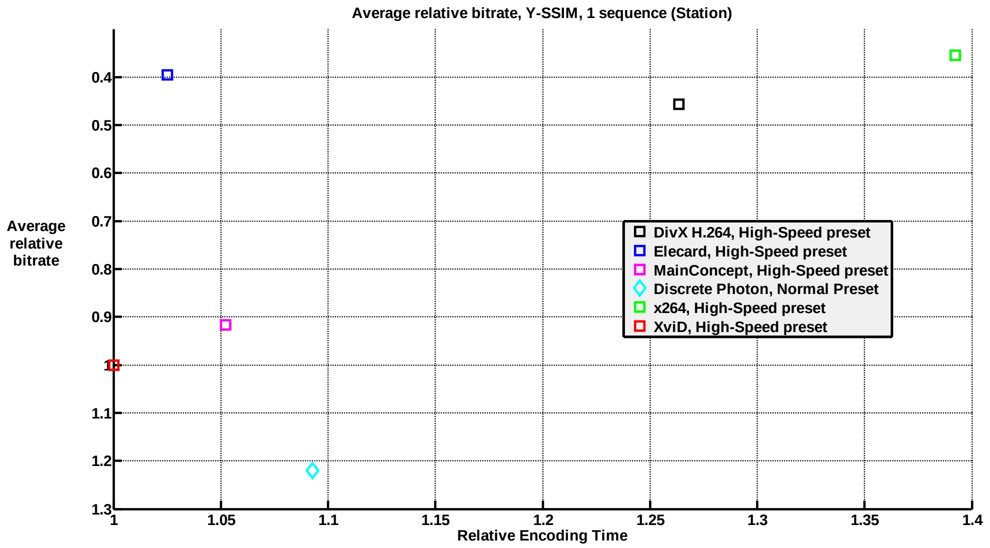


Figure 116. Speed/quality trade-off—usage area "HDTV," "Station" sequence, High Speed preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

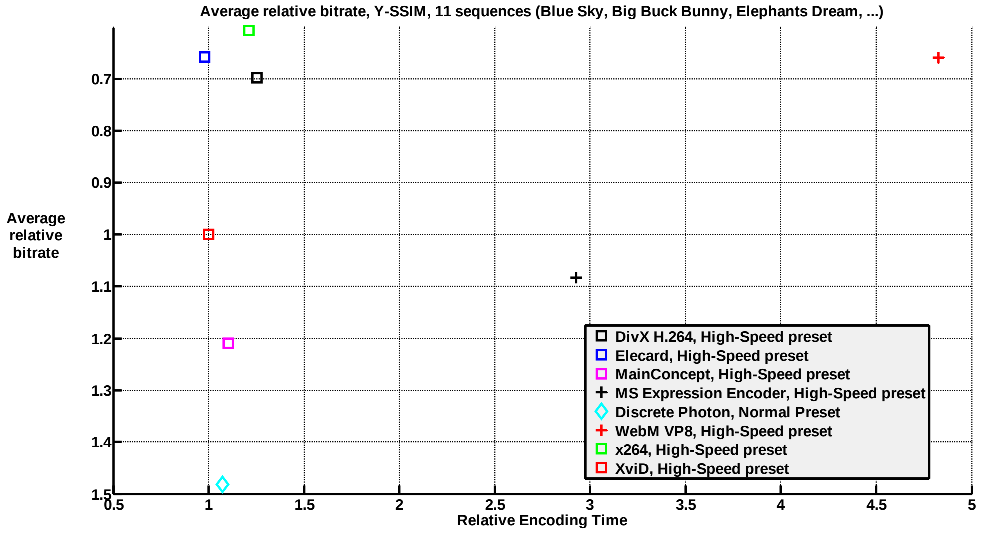


Figure 117. Speed/quality trade-off—usage area “HDTV,” all sequences, High Speed preset, Y-SSIM metric

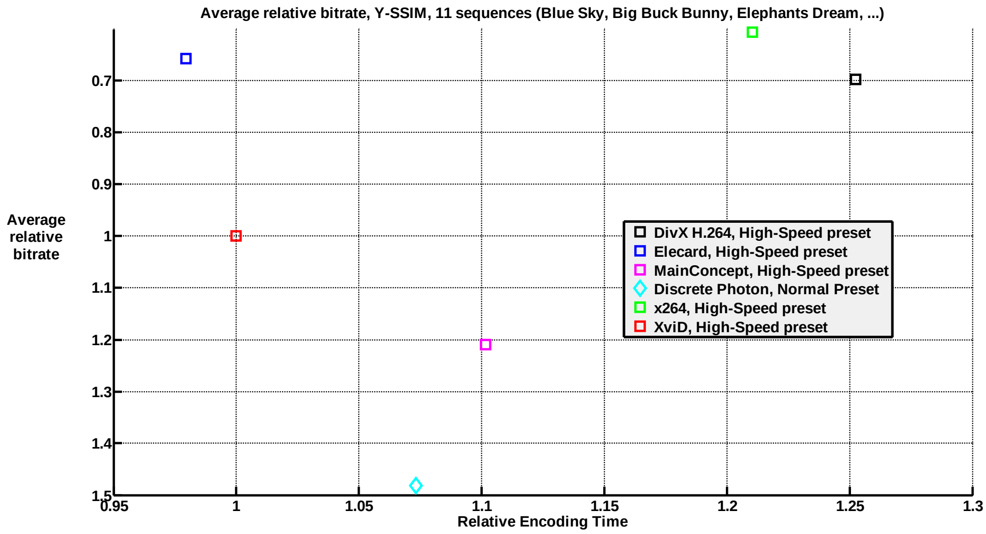


Figure 118. Speed/quality trade-off—usage area “HDTV,” all sequences, High Speed preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

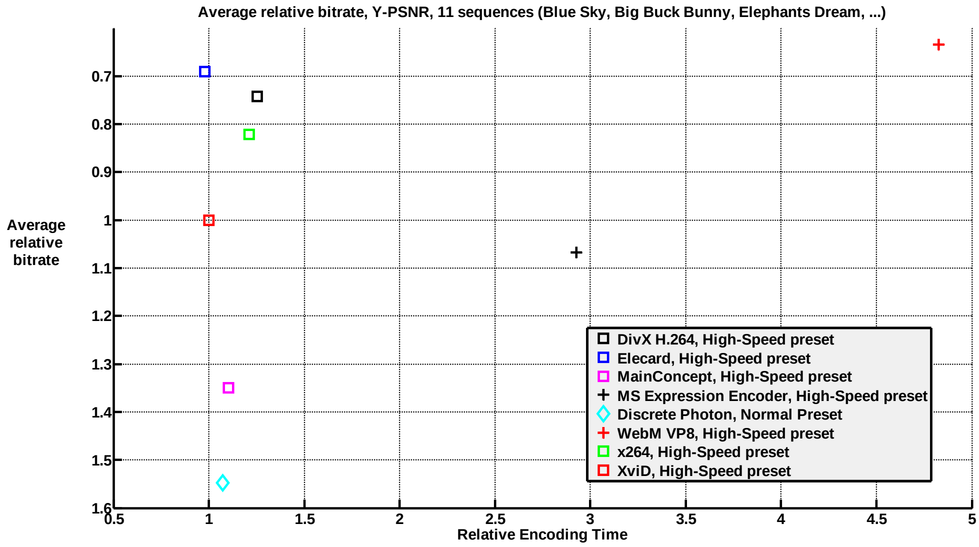


Figure 119. Speed/quality trade-off—usage area "HDTV," all sequences, High Speed preset, Y-PSNR metric

4.3.3.2 Normal Preset

Figure 120 through Figure 126 show results for the Normal preset. On average, the MainConcept, DivX H.264, Elecard and x264 codecs demonstrate best speed-quality trade-off. PSNR metric usage excludes MainConcept form leaders list.

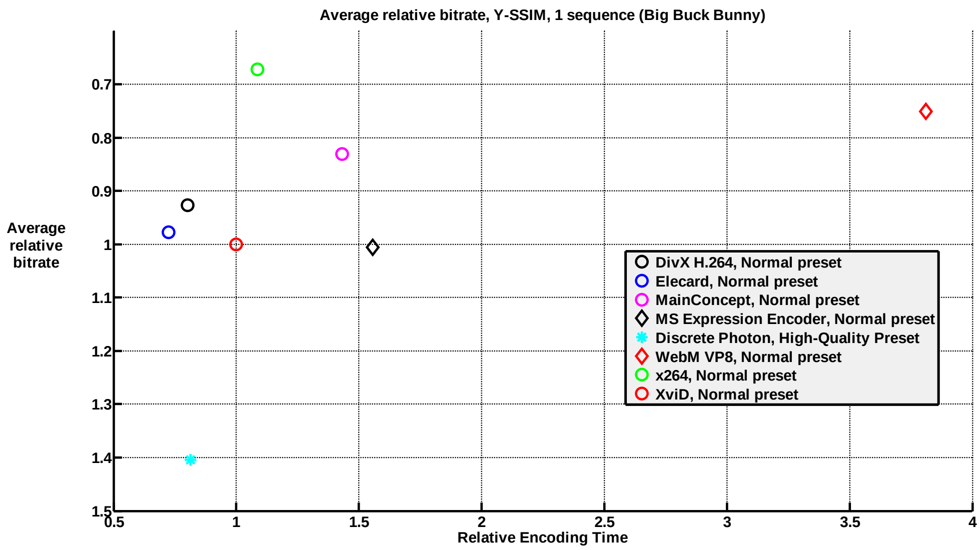


Figure 120. Speed/quality trade-off—usage area "HDTV," "Big Buck Bunny" sequence, Normal preset, Y-SSIM metric

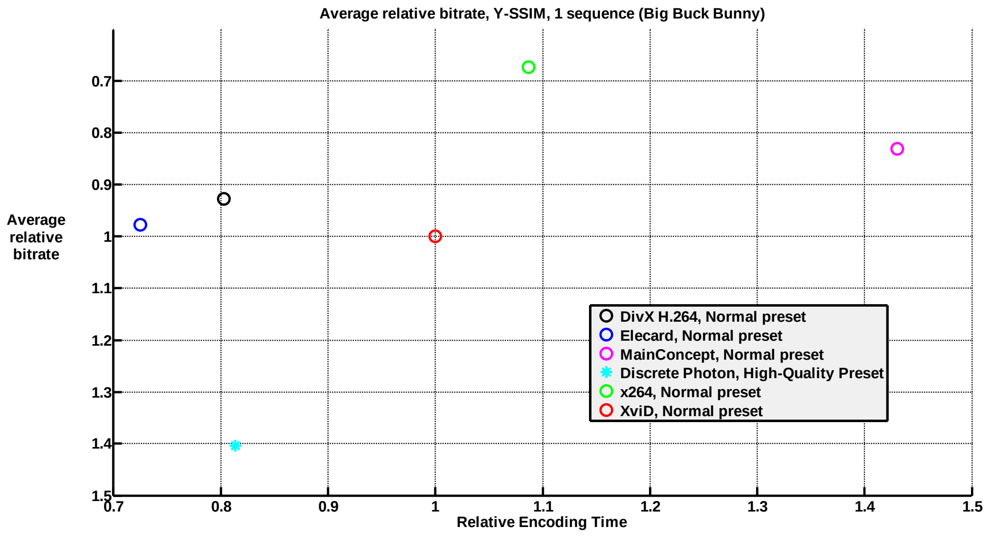


Figure 121. Speed/quality trade-off—usage area "HDTV," "Big Buck Bunny" sequence, Normal preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

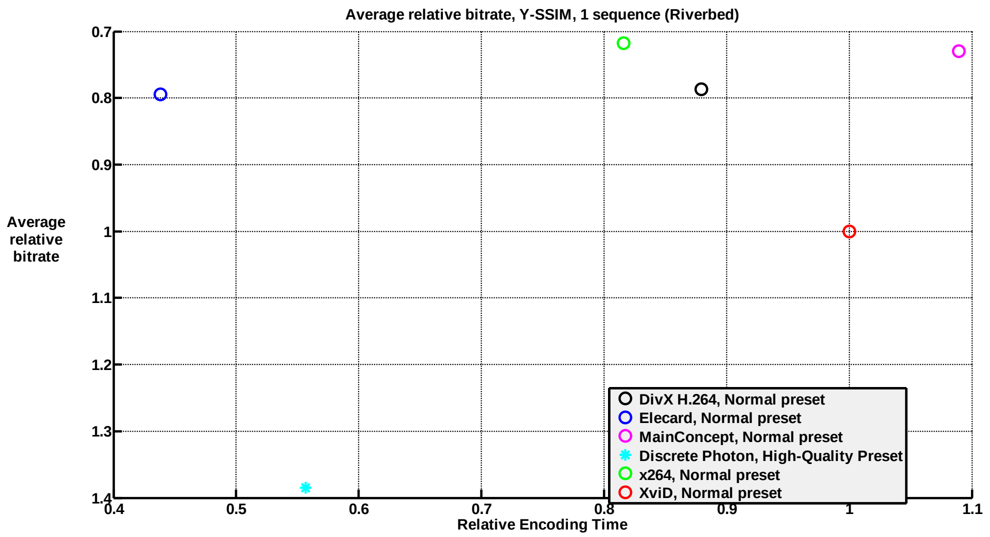


Figure 122. Speed/quality trade-off—usage area "HDTV," "Riverbed" sequence, Normal preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

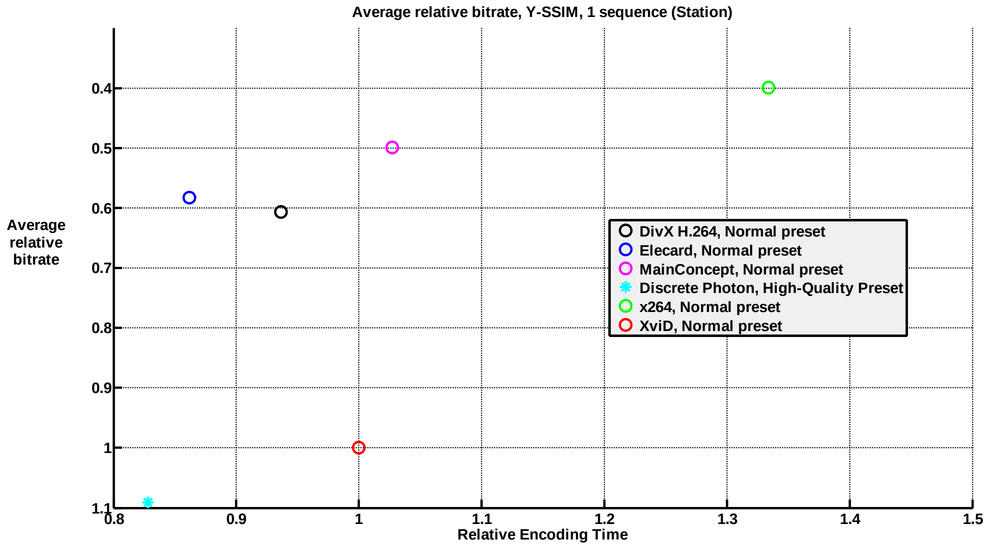


Figure 123. Speed/quality trade-off—usage area "HDTV," "Station" sequence, Normal preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

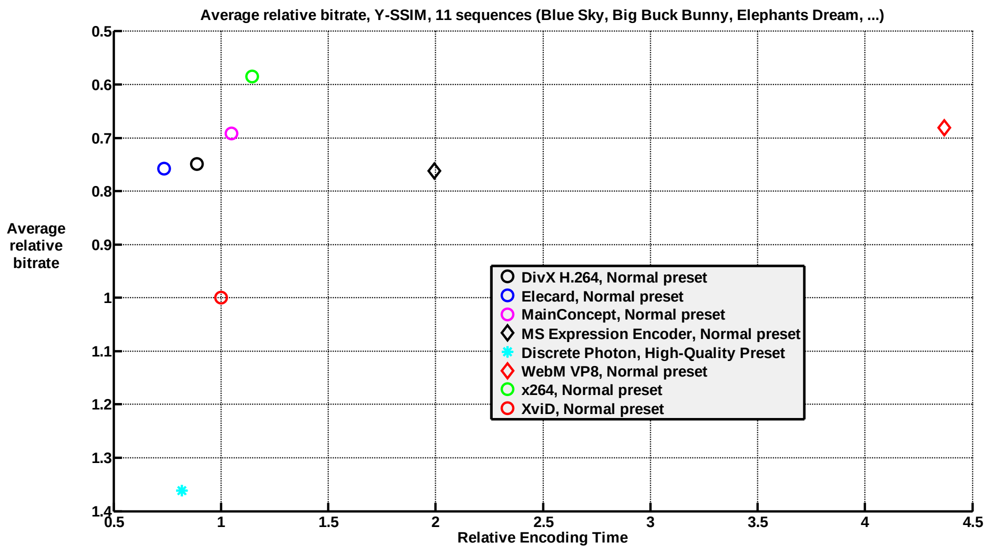


Figure 124. Speed/quality trade-off—usage area "HDTV," all sequences, Normal preset, Y-SSIM metric

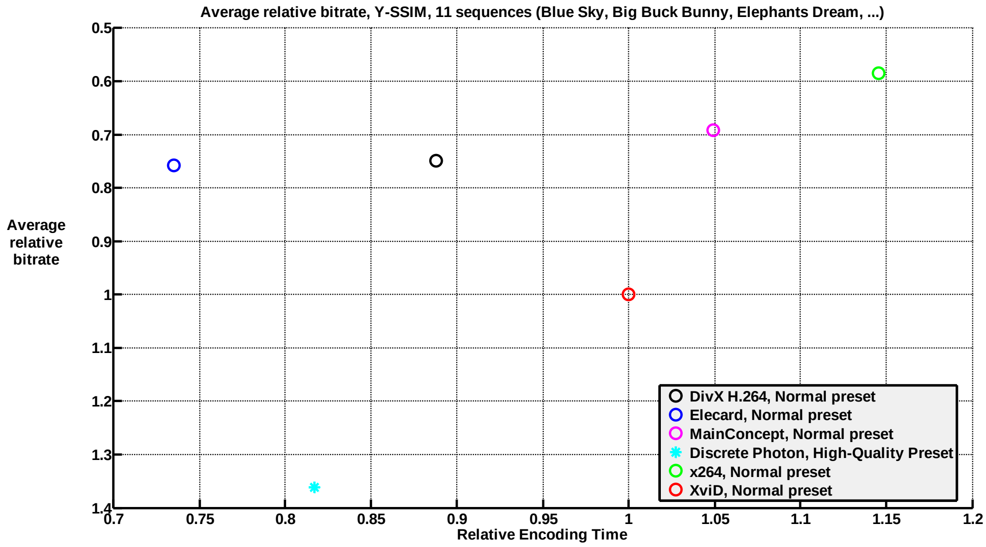


Figure 125. Speed/quality trade-off—usage area "HDTV," all sequences, Normal preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

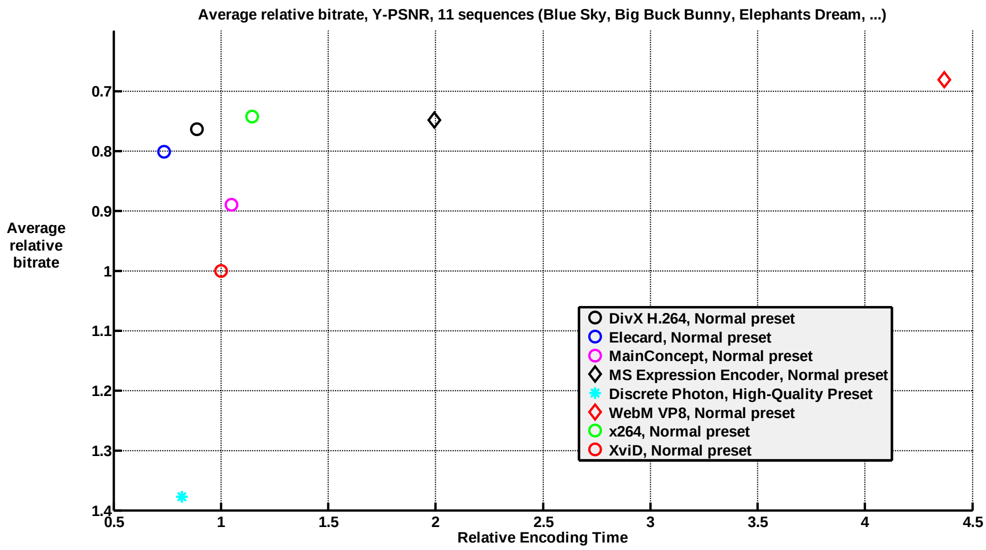


Figure 126. Speed/quality trade-off—usage area "HDTV," all sequences, Normal preset, Y-PSNR metric

4.3.3.3 High Quality Preset

Figure 127 through Figure 131 show results for the High Quality preset. All encoders except MSE and XviD are Pareto-optimal for this preset. The fastest codec is DiscretePhoton; x264 demonstrates the best quality. Y-PSNR usage removes MainConcept from Pareto-optimal codecs list.

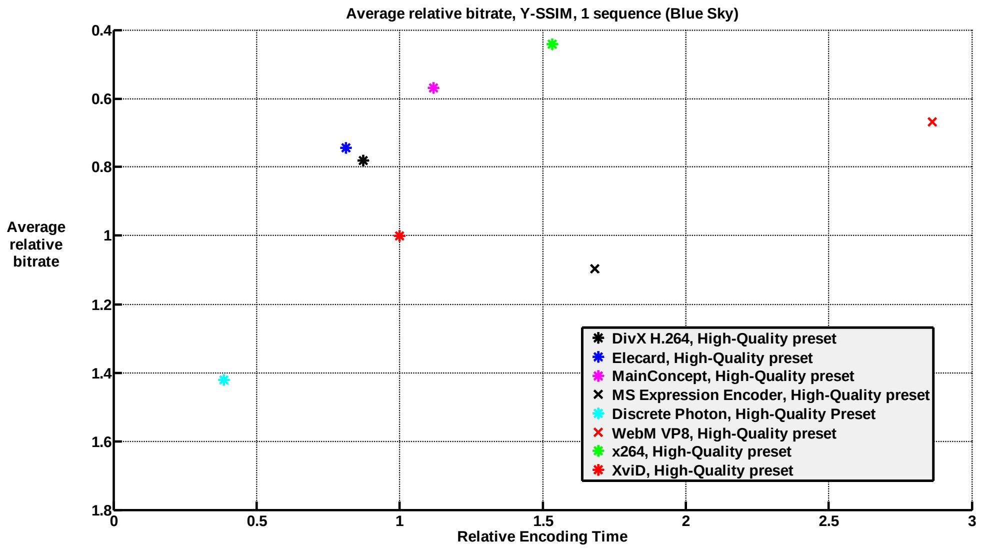


Figure 127. Speed/quality trade-off—usage area "HDTV," "Blue Sky" sequence, High Quality preset, Y-SSIM metric

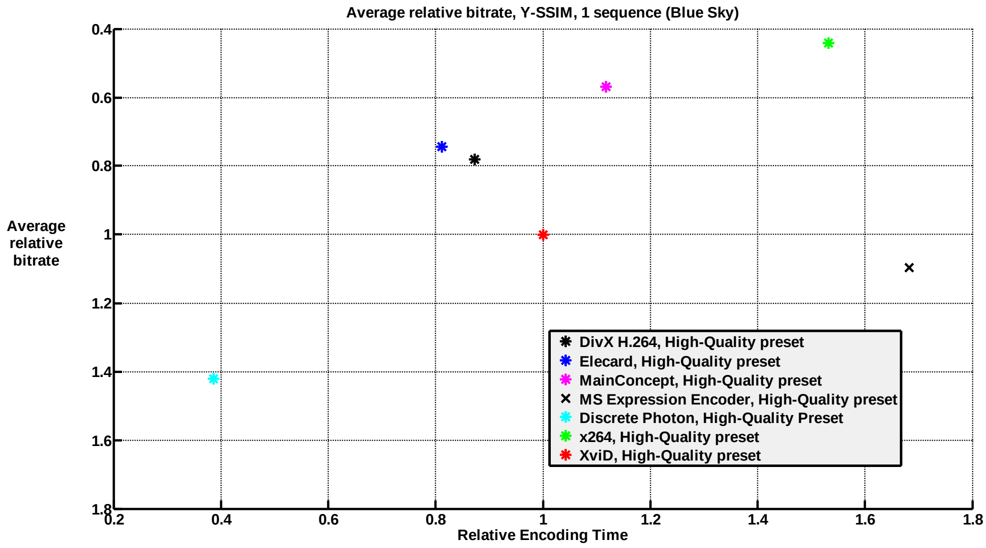


Figure 128. Speed/quality trade-off—usage area "HDTV," "Blue Sky" sequence, High Quality preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

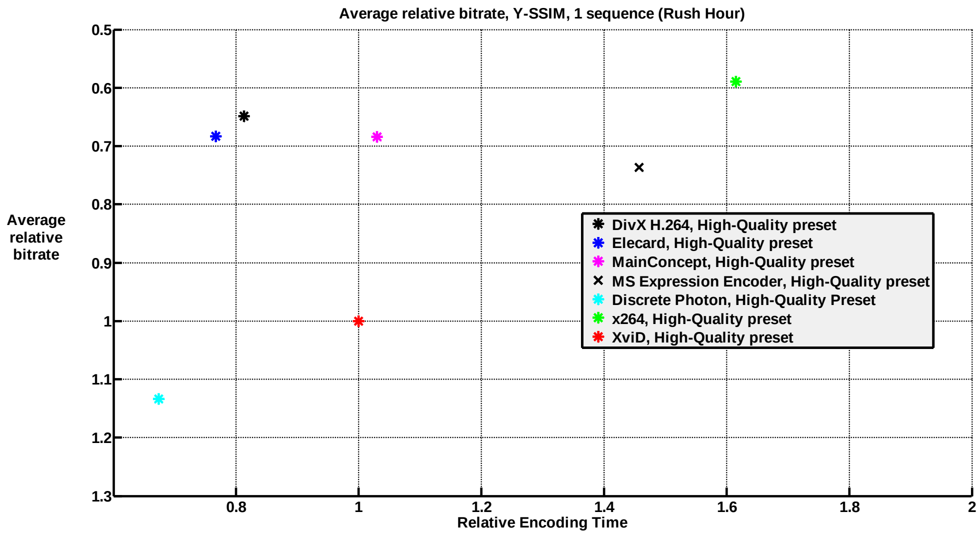


Figure 129. Speed/quality trade-off—usage area "HDTV," "Rush Hour" sequence, High Quality preset, Y-SSIM metric. Encoders that fit encoding speed requirements.

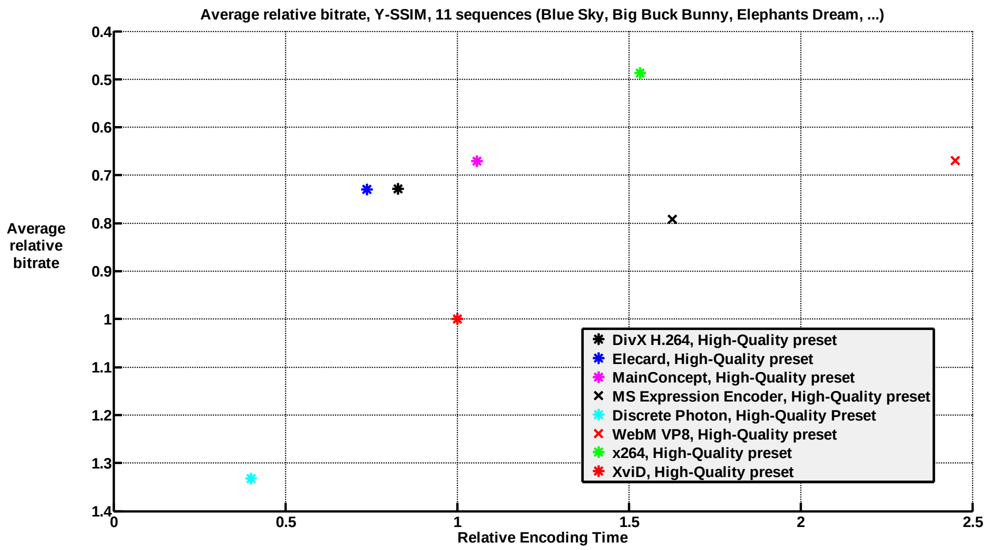


Figure 130. Speed/quality trade-off—usage area "HDTV," all sequences, High Quality preset, Y-SSIM metric

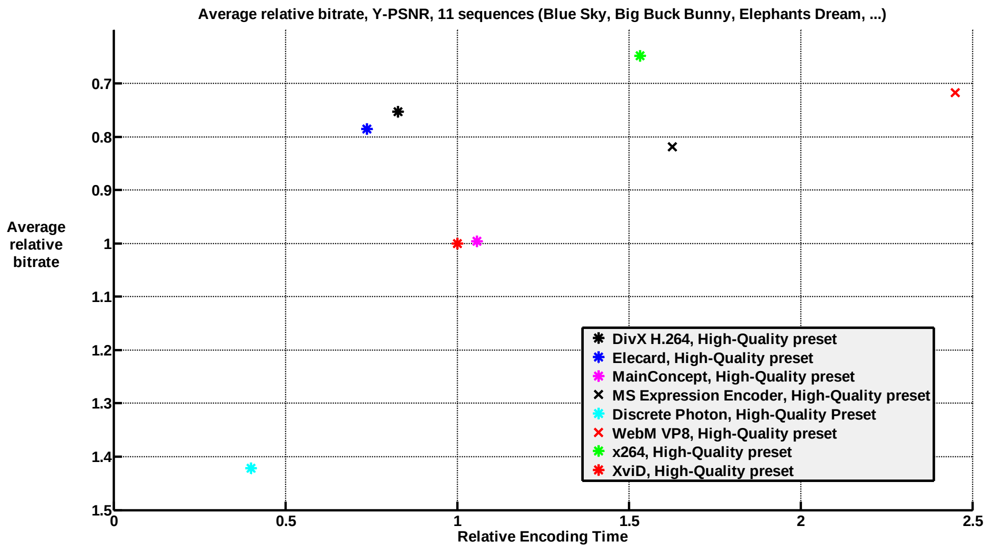


Figure 131. Speed/quality trade-off—usage area "HDTV," all sequences, High Quality preset, Y-PSNR metric

4.3.4 Bitrate Handling

4.3.4.1 High Speed Preset

Most codecs demonstrate problems maintaining steady low bitrate using fast presets. The XviD encoder provides the largest increase in bitrate—by more than five times.

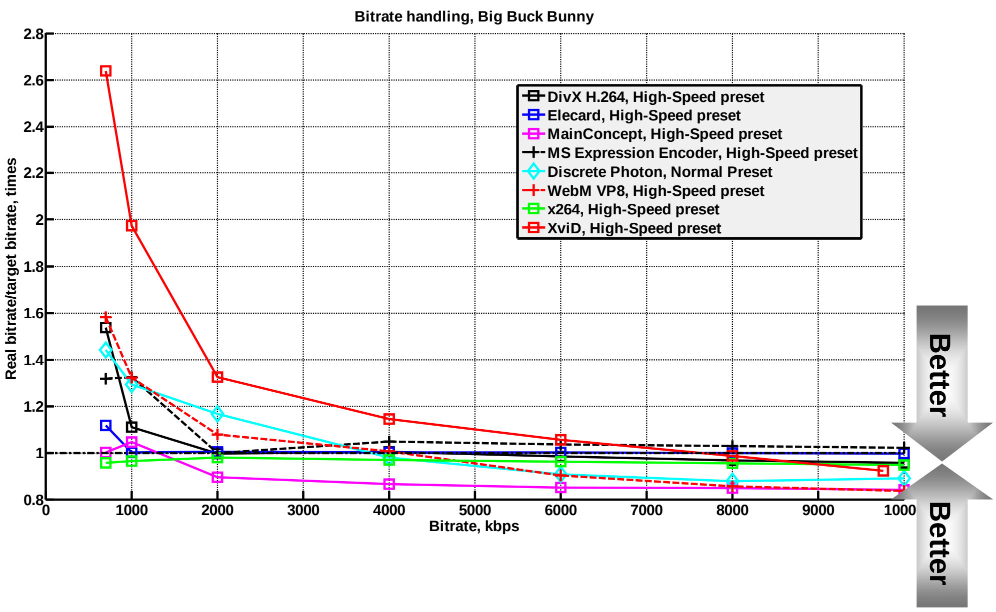


Figure 132. Bitrate handling—usage area "HDTV," "Big Buck Bunny" sequence, High Speed preset

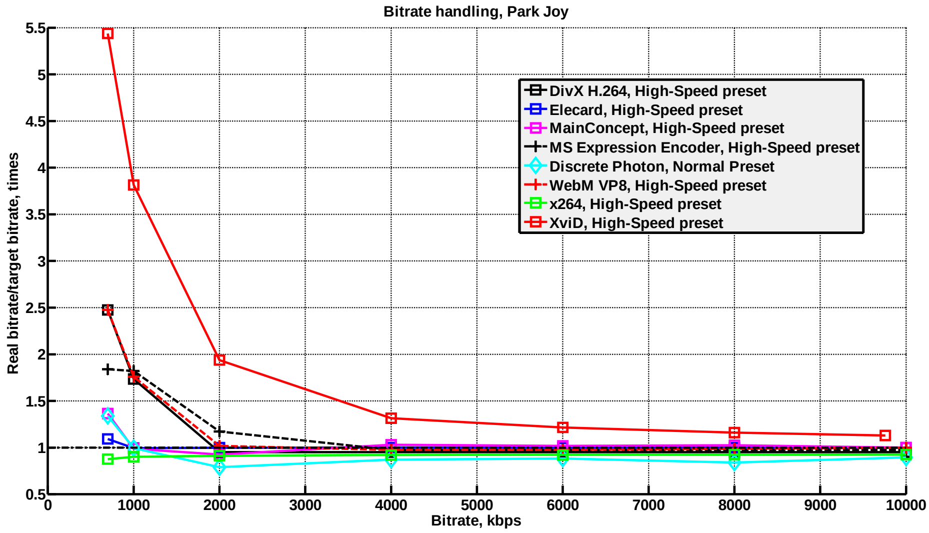


Figure 133. Bitrate handling—usage area "HDTV," "Park Joy" sequence, High Speed preset

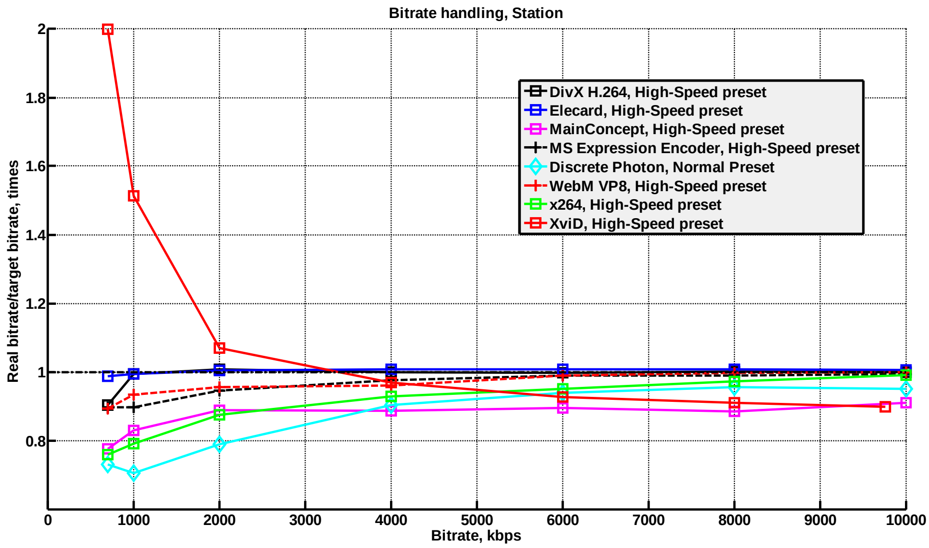


Figure 134. Bitrate handling—usage area "HDTV," "Station" sequence, High Speed preset

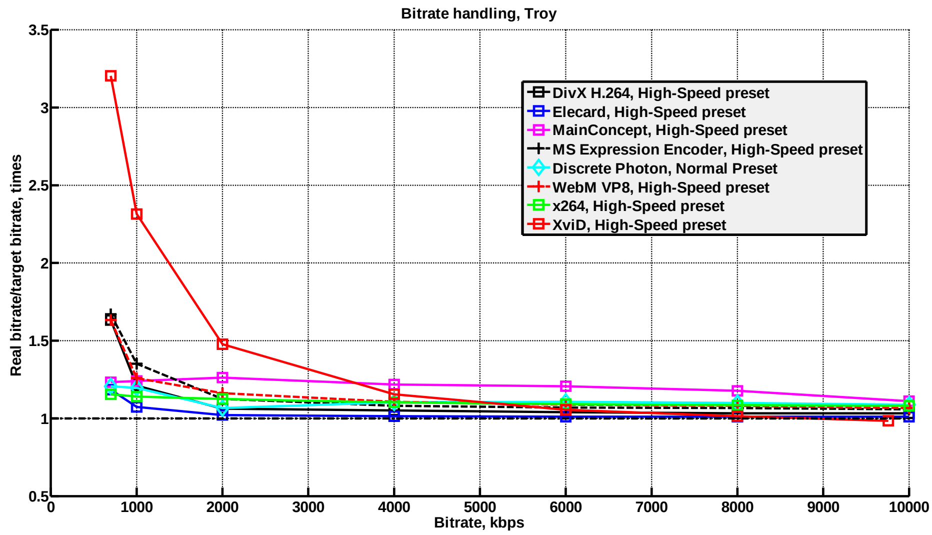
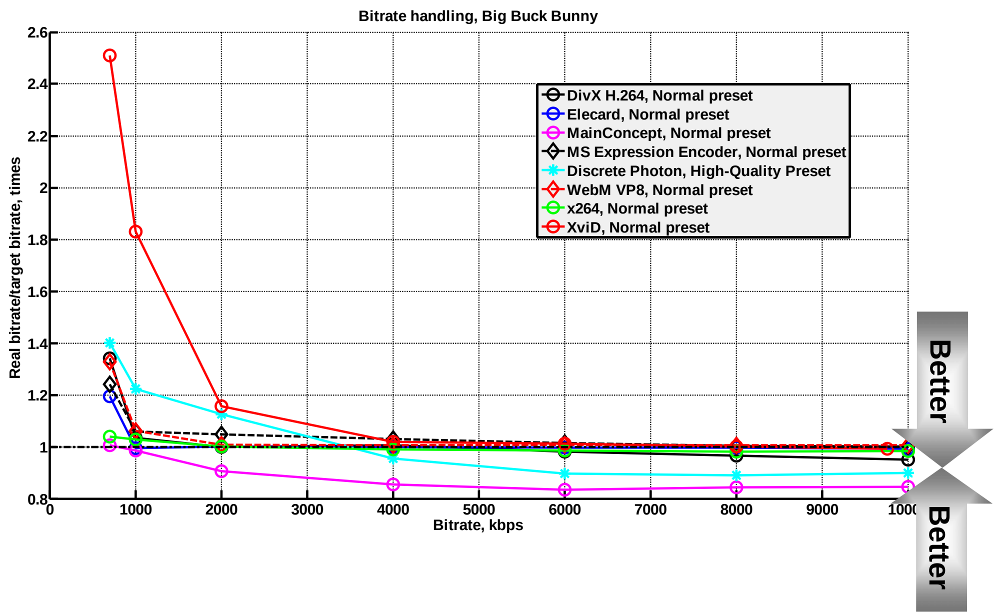


Figure 135. Bitrate handling—usage area "HDTV," "Troy" sequence, High Speed preset

4.3.4.2 Normal Preset

The codecs' behavior for the Normal preset is similar to that for the Fast preset. The XviD exhibits the worst bitrate handling. Interestingly, the MainConcept and XviD shows very bad bitrate handling mechanism in the case of the "Station" sequence. The leaders demonstrated good bitrate handling.



.

Figure 136. Bitrate handling—usage area "HDTV," "Big Buck Bunny" sequence, Normal preset

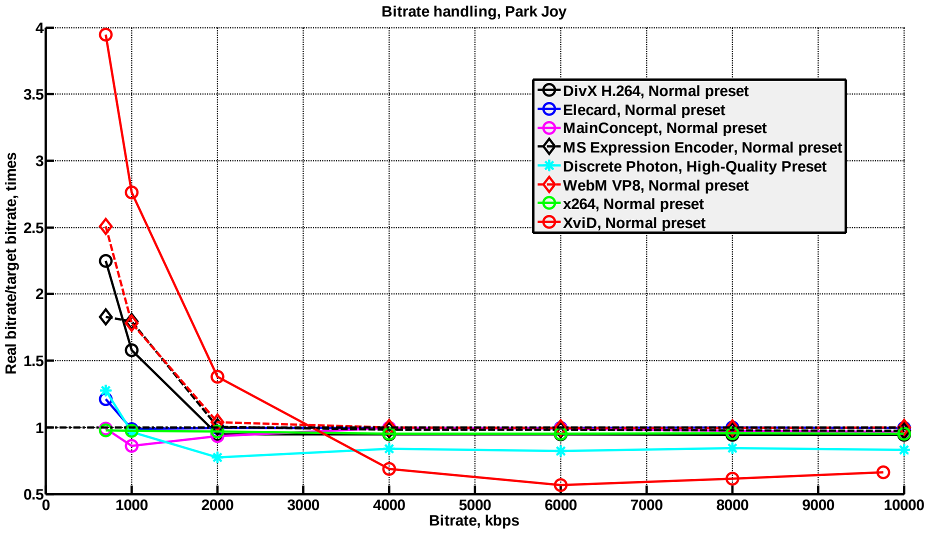


Figure 137. Bitrate handling—usage area "HDTV," "Park Joy" sequence, Normal preset

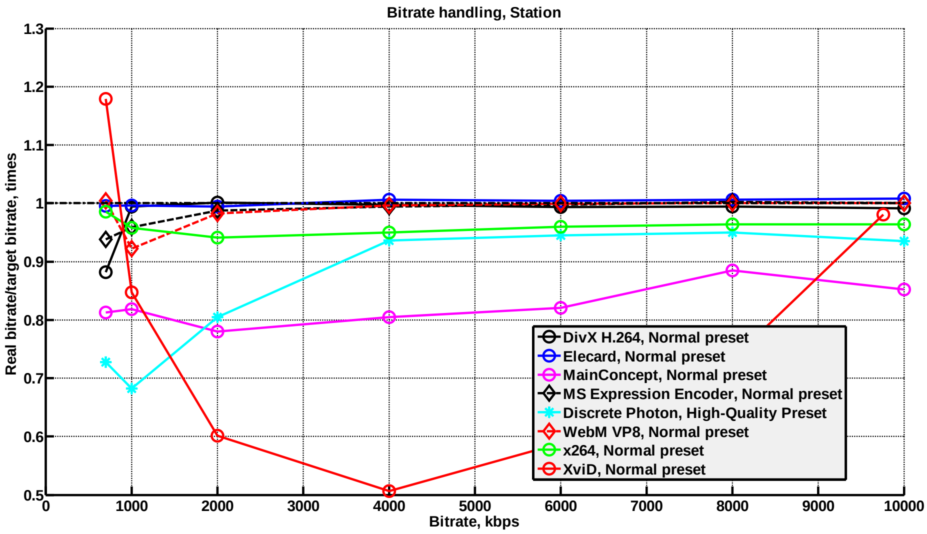


Figure 138. Bitrate handling—usage area "HDTV," "Station" sequence, Normal preset

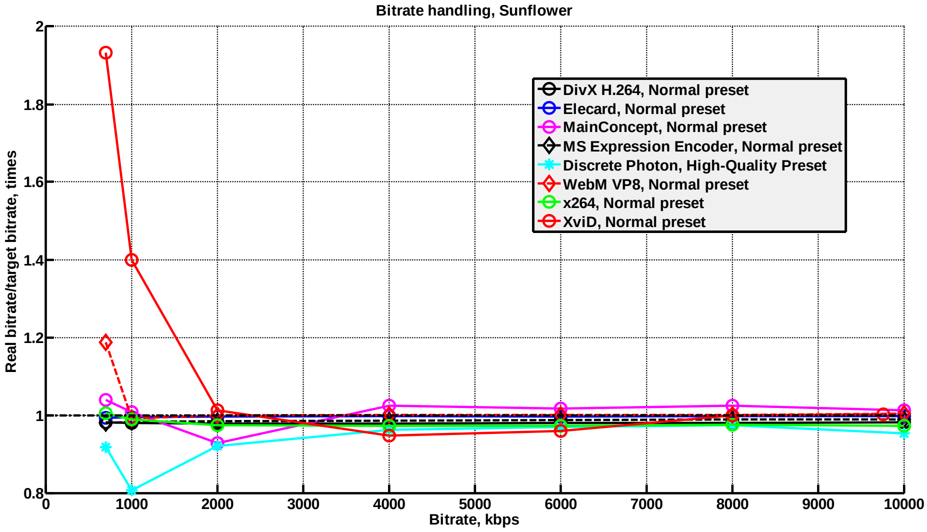


Figure 139. Bitrate handling—usage area "HDTV," "Sunflower" sequence, Normal preset

4.3.4.3 High Quality Preset

Most codecs, except XviD, maintain bitrate rather well. At some sequences MainConcept and DiscretePhoton decrease target bitrate.

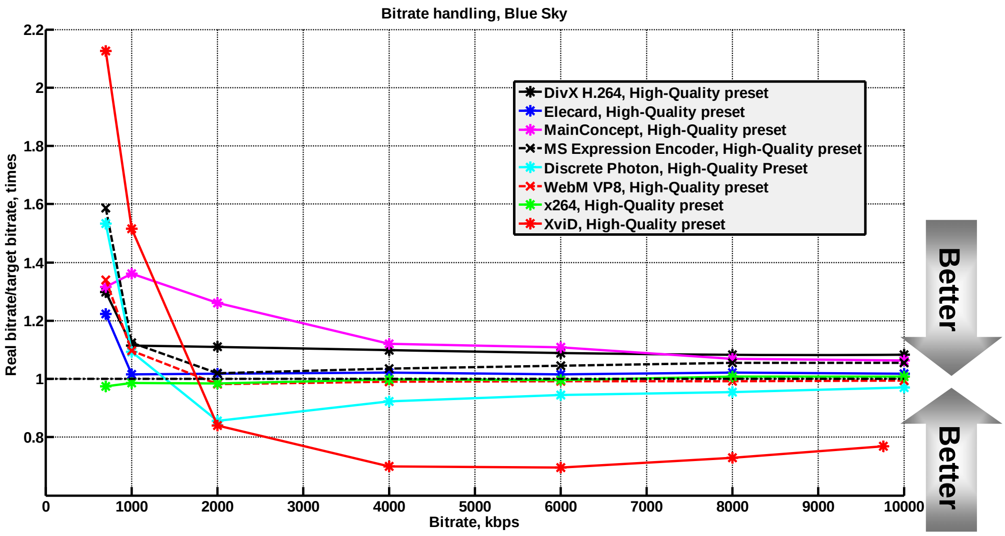


Figure 140. Bitrate handling—usage area "HDTV," "Blue Sky" sequence, High Quality preset

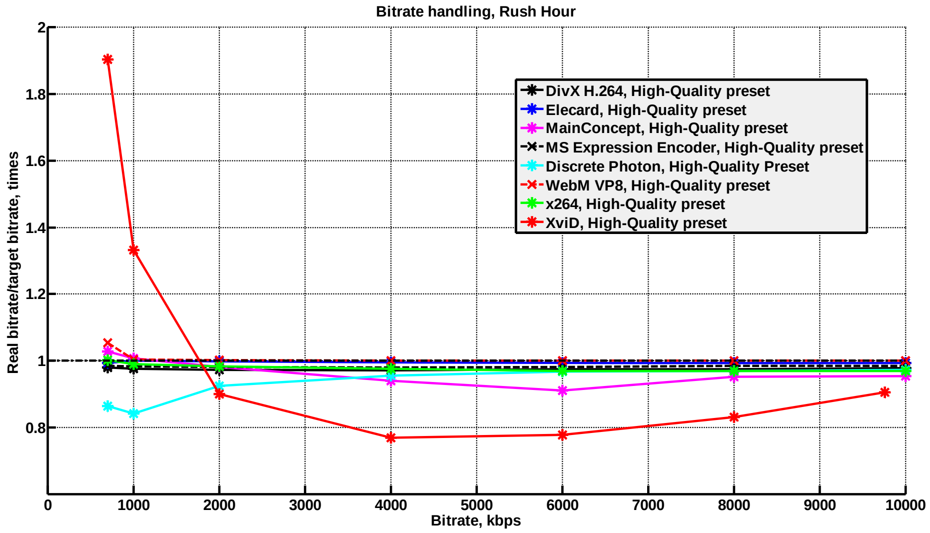


Figure 141. Bitrate handling—usage area “HDTV,” “Rush Hour” sequence, High Quality preset

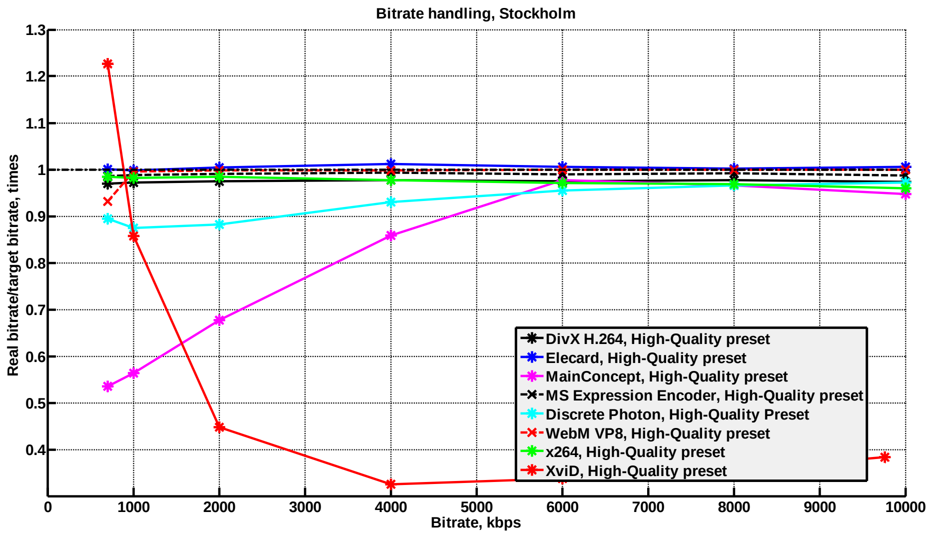


Figure 142. Bitrate handling—usage area “HDTV,” “Stockholm” sequence, High Quality preset

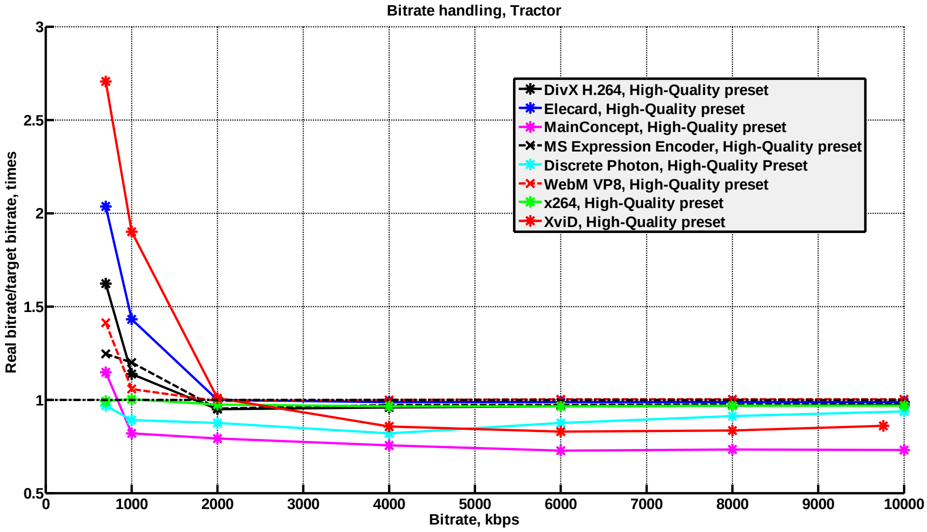


Figure 143. Bitrate handling—usage area "HDTV," "Tractor" sequence, High Quality preset

4.3.5 Relative Quality Analysis

Table 11 through Table 16 show relative bitrates for a fixed-quality output for all codecs and presets. Note that these tables do not include information about encoder speed.

Note that each number in the tables below corresponds to some range of bitrates (see Appendix 7. Figures Explanation for more details). Unfortunately, these ranges can differ significantly because of differences in the quality produced by the encoders under comparison. This situation can lead to some inadequate results when comparing three or more codecs.

Table 11. Average bitrate ratio for a fixed quality—usage area "HDTV". High Speed preset, Y-SSIM metric.

	DivX H.264	Elecard	MainConcept	Discrete Photon	x264	XviD
DivX H.264	100%	95%	166%	204%	88%	144%
Elecard	105%	100%	187%	230%	92%	152%
MainConcept	60%	53%	100%	127%	49%	83%
Discrete Photon	49%	43%	79%	100%	41%	67%
x264	114%	109%	204%	246%	100%	165%
XviD	70%	66%	121%	148%	61%	100%

Table 12. Average bitrate ratio for a fixed quality—usage area "HDTV". High Speed preset, Y-PSNR metric.

	DivX H.264	Elecard	MainConcept	Discrete Photon	x264	XviD
DivX H.264	100%	93%	196%	218%	113%	135%
Elecard	107%	100%	226%	245%	122%	145%
MainConcept	51%	44%	100%	120%	51%	74%
Discrete Photon	46%	41%	84%	100%	46%	65%
x264	88%	82%	195%	215%	100%	122%
XviD	74%	69%	135%	155%	82%	100%

Table 13. Average bitrate ratio for a fixed quality—usage area "HDTV". Normal preset, Y-SSIM metric.

	DivX H.264	Elecard	MainConcept	Discrete Photon	x264	XviD
DivX H.264	100%	99%	94%	187%	75%	133%
Elecard	101%	100%	93%	179%	76%	132%
MainConcept	106%	107%	100%	208%	81%	144%
Discrete Photon	53%	56%	48%	100%	43%	73%
x264	134%	131%	124%	232%	100%	171%
XviD	75%	76%	69%	136%	58%	100%

Table 14. Average bitrate ratio for a fixed quality—usage area "HDTV". Normal preset, Y-PSNR metric.

	DivX H.264	Elecard	MainConcept	Discrete Photon	x264	XviD
DivX H.264	100%	104%	139%	187%	98%	131%
Elecard	97%	100%	129%	178%	94%	125%
MainConcept	72%	78%	100%	150%	70%	112%
Discrete Photon	53%	56%	67%	100%	51%	73%
x264	103%	106%	144%	195%	100%	135%
XviD	76%	80%	89%	138%	74%	100%

Table 15. Average bitrate ratio for a fixed quality—usage area "HDTV". High Quality preset, Y-SSIM metric.

	DivX H.264	Elecard	MainConcept	Discrete Photon	x264	XviD
DivX H.264	100%	99%	95%	197%	64%	137%
Elecard	101%	100%	93%	185%	66%	137%
MainConcept	106%	108%	100%	220%	68%	149%
Discrete Photon	51%	54%	45%	100%	34%	75%
x264	157%	151%	146%	294%	100%	205%
XviD	73%	73%	67%	133%	49%	100%

Table 16. Average bitrate ratio for a fixed quality—usage area "HDTV". High Quality preset, Y-PSNR metric.

	DivX H.264	Elecard	MainConcept	Discrete Photon	x264	XviD
DivX H.264	100%	104%	139%	198%	88%	133%
Elecard	96%	100%	128%	188%	84%	127%
MainConcept	72%	78%	100%	159%	55%	100%
Discrete Photon	51%	53%	63%	100%	41%	70%
x264	114%	120%	180%	242%	100%	154%
XviD	75%	79%	100%	142%	65%	100%

Figure 144 through Figure 149 depict the data in the tables above. Each line in these figures corresponds to one codec. Values along the vertical axis are average relative bitrates as compared with the codecs along the horizontal axis. A lower bitrate indicates better relative results.

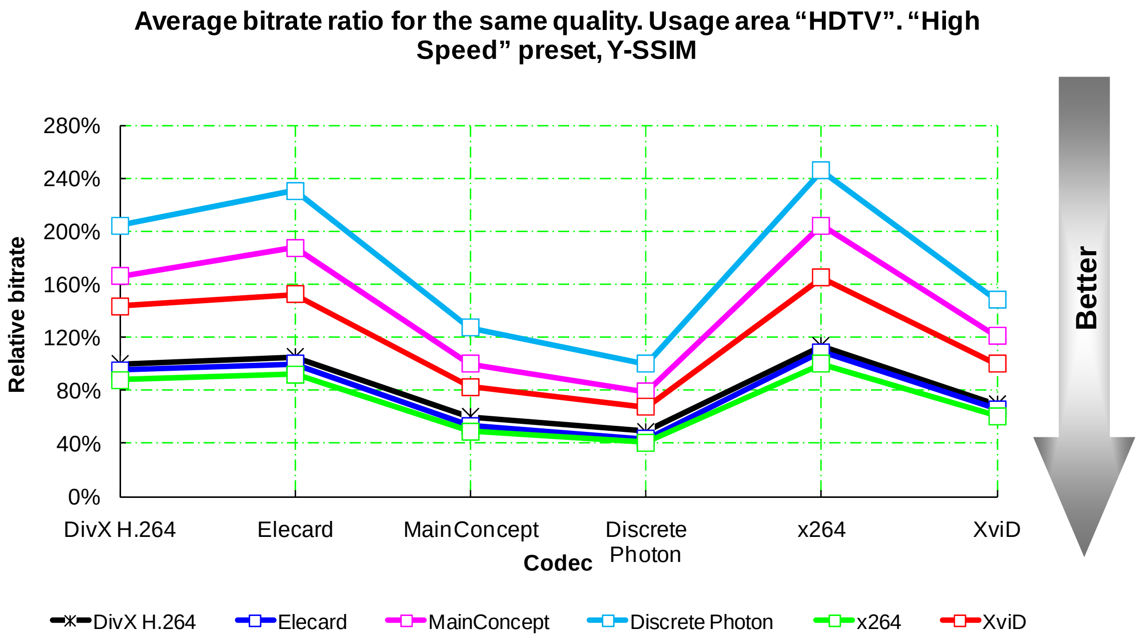


Figure 144. Average bitrate ratio for a fixed quality—usage area “HDTV”. High Speed preset, Y-SSIM metric.

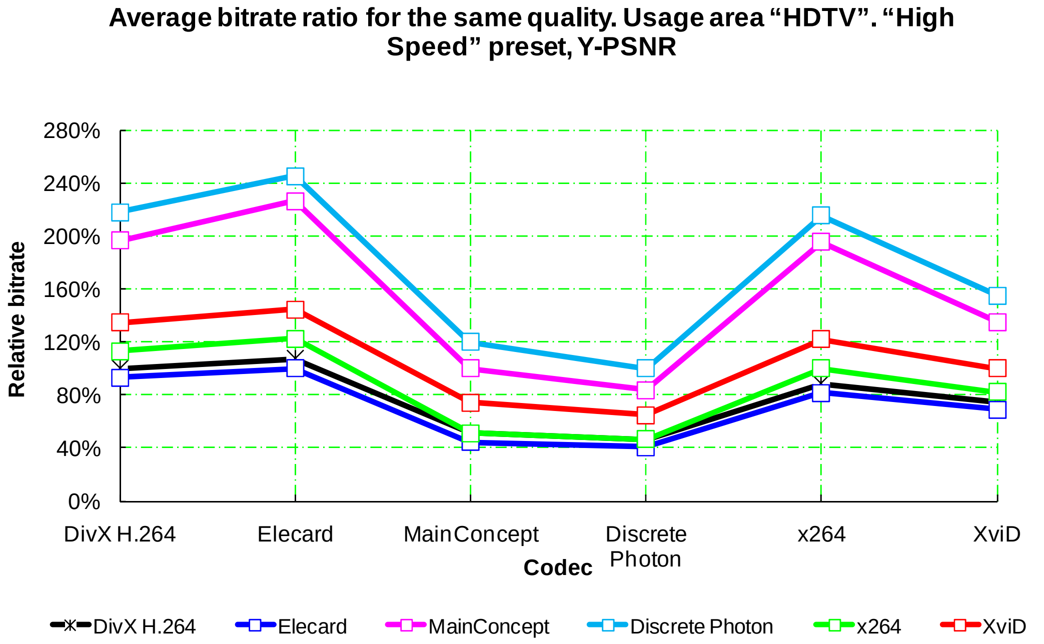


Figure 145. Average bitrate ratio for a fixed quality—usage area “HDTV”. High Speed preset, Y-PSNR metric.

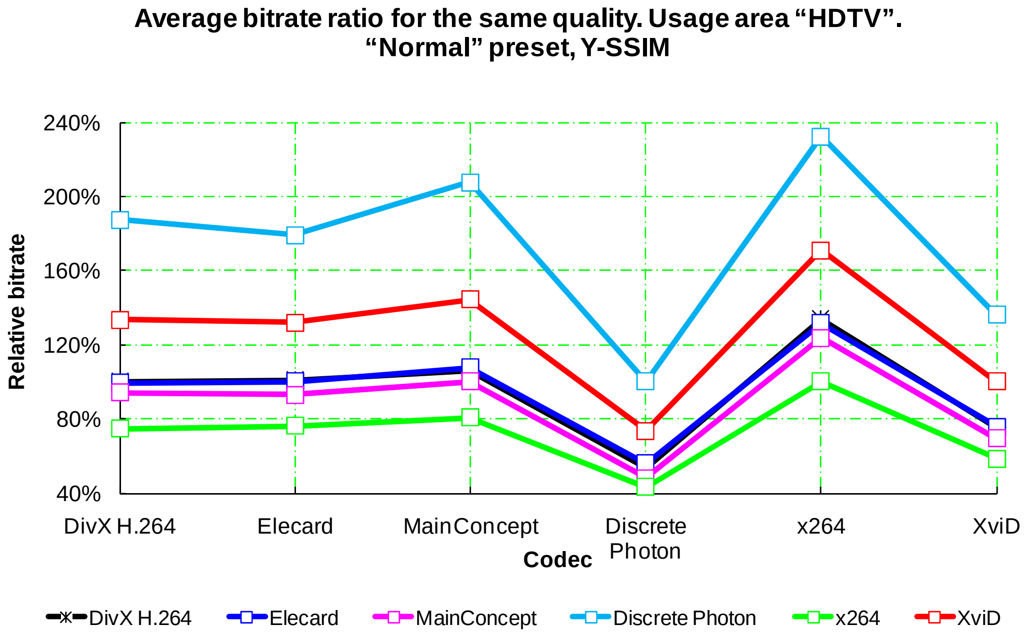


Figure 146. Average bitrate ratio for a fixed quality—usage area "HDTV". Normal preset, Y-SSIM metric.

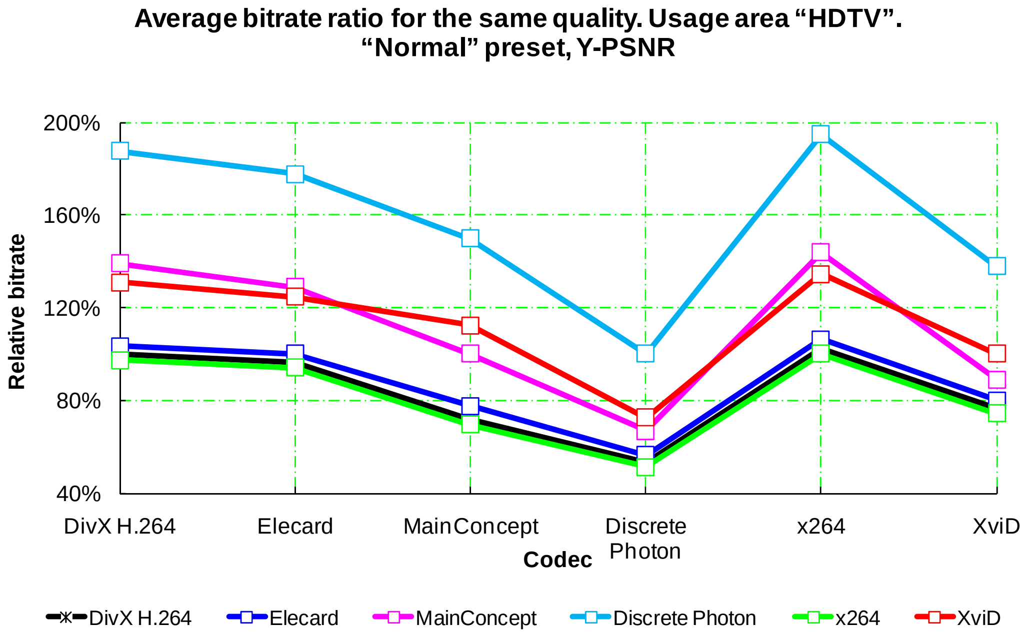


Figure 147. Average bitrate ratio for a fixed quality—usage area "HDTV". Normal preset, Y-PSNR metric.

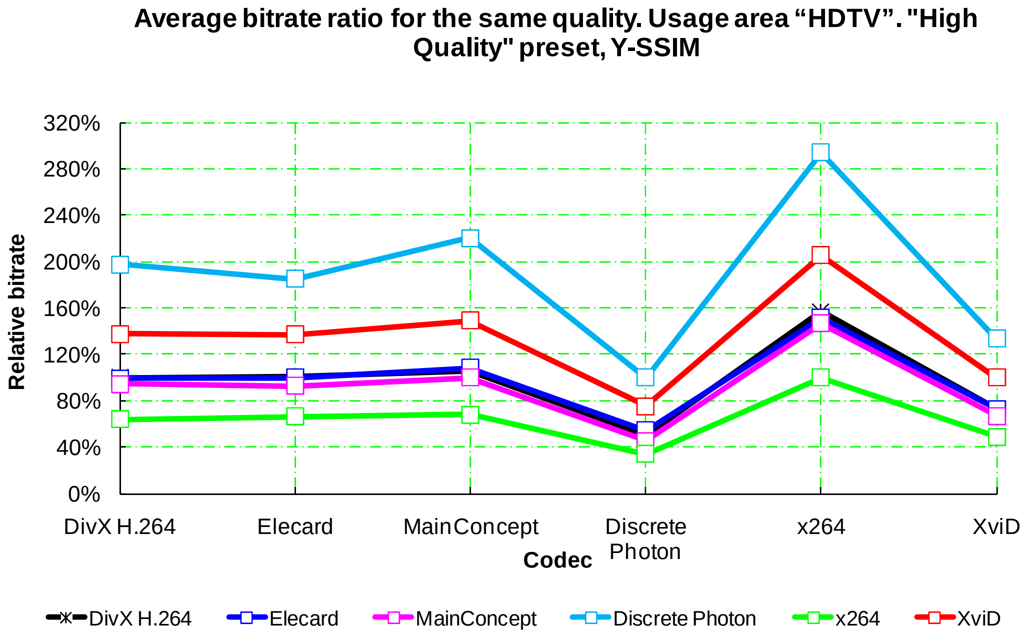


Figure 148. Average bitrate ratio for a fixed quality—usage area “HDTV”. High Quality preset, Y-SSIM metric.

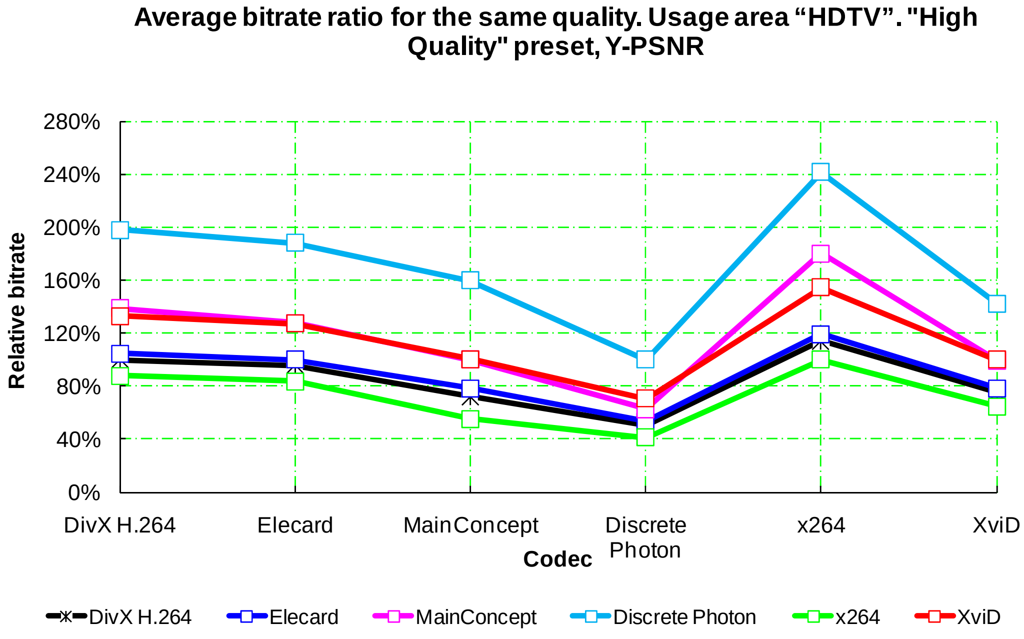


Figure 149. Average bitrate ratio for a fixed quality—usage area “HDTV”. High Quality preset, Y-PSNR metric.

4.4 Conclusions

4.4.1 Video Conference

The x264 encoder demonstrates better quality on average. The top three codecs for this preset are the following:

1. x264
2. DivX H.264
3. Elecard

But the x264 encoder demonstrates the lowest quality of local bitrate handling (see 4.1.5 Local Bitrate Handling), so analyzed preset could be hardly used for video conference encoding.

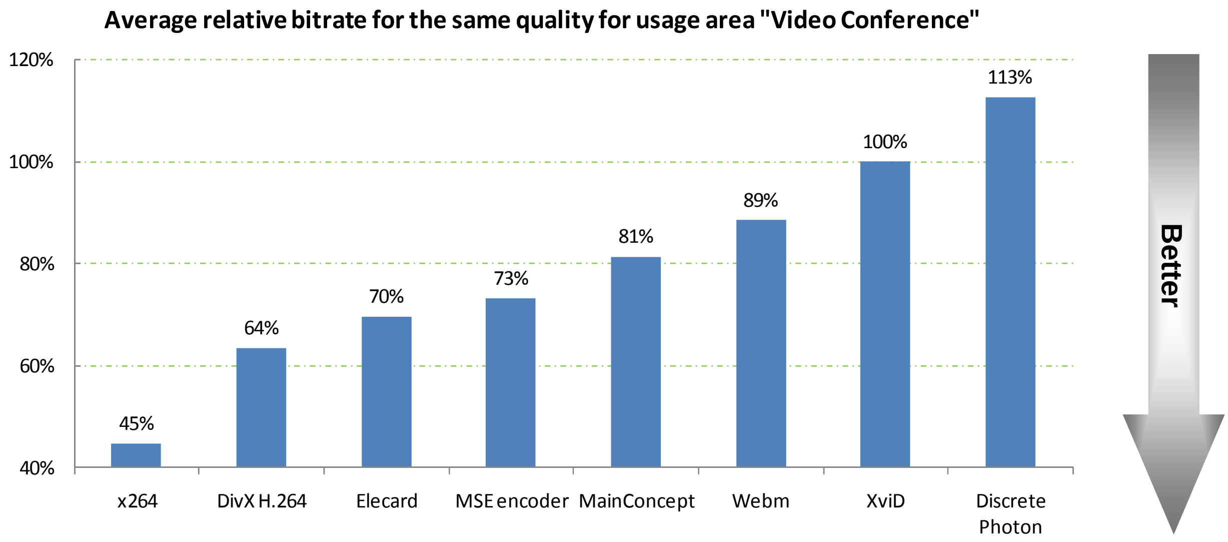


Figure 150. Average bitrate ratio for a fixed quality—usage area "Video Conference," Y-SSIM.

4.4.2 Movies

The leading encoder in this usage area is x264, followed by MainConcept, DivX H.264 and Elecard.

4.4.2.1 High Speed Preset

The x264 encoder demonstrates better quality on average, and MainConcept shows slightly lower quality. The top three codecs for this preset are the following:

1. x264
2. MainConcept
3. Elecard

4.4.2.2 Normal Preset

The results for the Normal preset differ from those for the High Speed presets in second and third places. The x264 encoder demonstrates better quality on average, and MainConcept and Divx H.264 show slightly lower quality. The Elecard encoder holds third place. The top four codecs for this preset are the following:

1. x264
2. MainConcept and DivX H.264
3. Elecard

4.4.2.3 High Quality Preset

The results for this preset are similar to those of the Normal preset. The leader is the x264, followed by three codecs. The top four codecs for this preset are the following:

1. x264
2. MainConcept and DivX H.264
3. Elecard

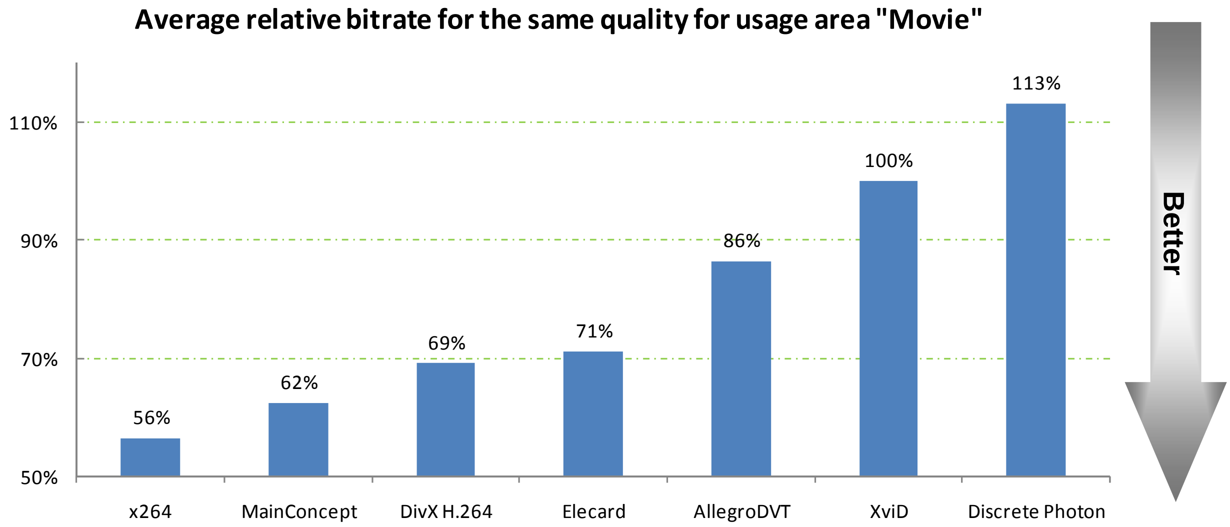


Figure 151. Average bitrate ratio for a fixed quality—usage area “Movies,” all presets, Y-SSIM.

4.4.3 HDTV

The leaders in the HDTV area are x264, DivX H.264, Elecard and MainConcept. The DiscretePhoton encoder trails all other H.264 encoders.

4.4.3.1 High Speed Preset

The x264 encoder demonstrates better quality on average, and Elecard and DivX H.264 show slightly lower quality. The top three codecs for this preset are the following:

1. x264
2. Elecard
3. DivX H.264

4.4.3.2 Normal Preset

The results for the Normal preset differ from those of the High Speed presets. x264 shows best result, MainConcept shows better results than for High Speed preset; DivX H.264 and Elecard share third place (their quality results are very similar).The top four codecs for this preset are the following:

1. x264

2. MainConcept
3. DivX H.264 and Elecard

4.4.3.3 High Quality Preset

The results for the High Quality preset are very close to Normal preset results: x264 held first place, MainConcept is in second place, and two codecs (DivX H.264 and Elecard) share third place (their quality results are very close). The top four codecs for this preset are the following:

1. x264
2. MainConcept
3. DivX H.264 and Elecard

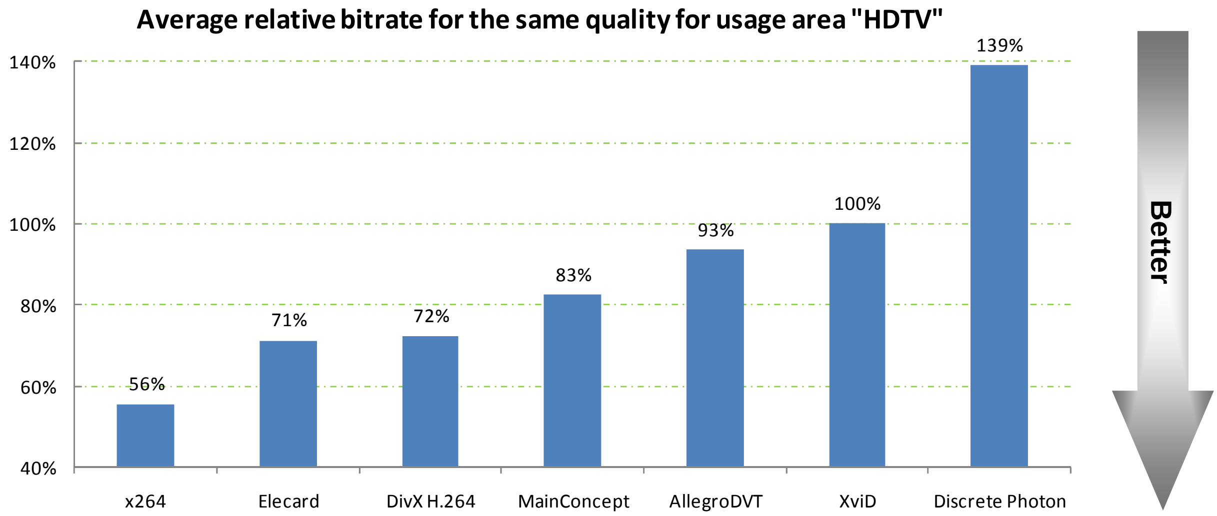


Figure 152. Average bitrate ratio for a fixed quality—usage area "HDTV," all presets, Y-SSIM.

4.4.4 Overall Conclusions

Overall, the leader in this comparison is x264, followed by DivX H.264, Elecard and MainConcept. The DIscretePhoton encoder demonstrates the worst results among all codecs tested.

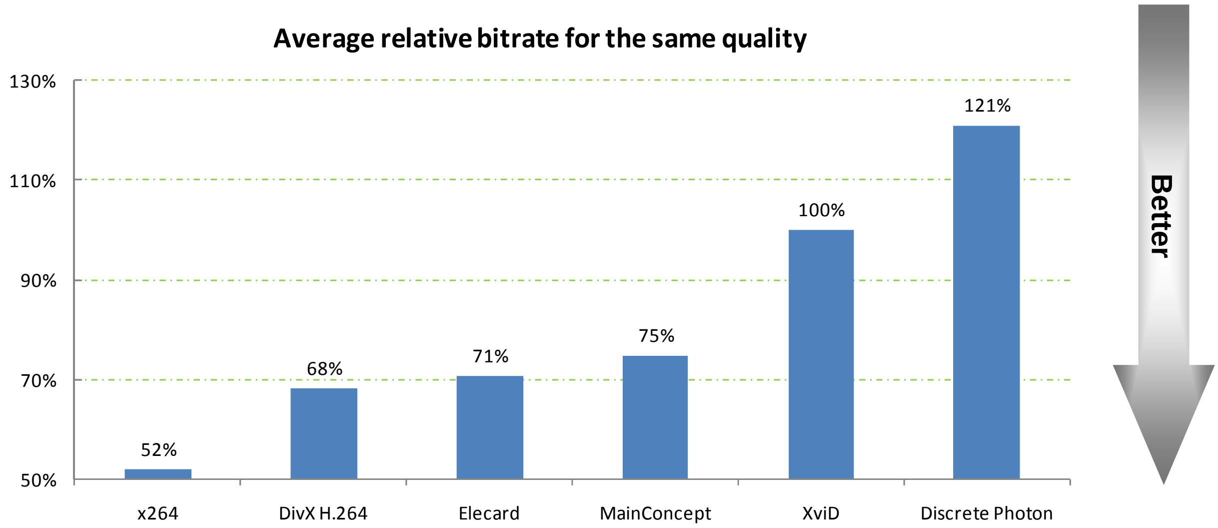


Figure 153. Average bitrate ratio for a fixed quality for all categories and all presets (Y-SSIM).

The overall ranking of the codecs tested in this comparison is as follows:

1. x264
2. DivX H.264
3. Elecard
4. MainConcept
5. XviD
6. DiscretePhoton

- MSE encoder
- WebM encoder

WebM and Microsoft Expression encoders could not be placed in this list because of their longer encoding time compared with other encoders.

The leader in this comparison is x264—its quality difference (according to the SSIM metric) could be explained by the special encoding option ("tune-SSIM").

The difference between the Elecard and DivX H.264 encoders is almost nothing, and between these encoders and MainConcept is not overly significant, so these encoders tied for second and third in this comparison.

This rank is based only on the encoders' quality results (see Figure 153). Encoding speed is not considered here.

4.4.5 Codec Conclusions

- **Discrete Photon—**one of the fastest encoder for this comparison, but because of its speed the encoding quality was not very good
- **DivX H.264—**one of comparison leaders, quite balanced encoder with not very big number of parameters, this fact could be comfortable for users. This encoder is designed as a free sample application for DivX Plus HD compliant video encoding, and is a feature-constrained, for-purpose application.

- **Elecard—**one of comparison leaders, codec with good encoding quality and very flexible settings. Many adjustable encoding settings are provided.
- **Microsoft Expression Encoder—**encoder with good encoding quality but due to the fact of long initial loading time, the encoding time for Microsoft Expression Encoder is significantly higher than for other encoders**.**
- **MainConcept—**good balanced encoder; has many encoding settings that can be adjusted**.** The results for Movie use-case was second, so this codec has a good potential to be one of comparison leaders
- **x264—**one of the best codecs by encoding quality; has very user-friendly predefined presets, as well as many adjustable encoding settings.
- **XviD—**an MPEG-4 ASP codec; its quality could be very close to or even higher than that of some commercial H.264 standard implementations, especially for encoding “Movie” sequences, but not for “HDTV” sequences.
- **WebM—**good new non H.264 encoder, it shows good quality but due to the low encoding speed it is not presented in encoders list by quality

4.4.6 Comments from Developers

4.4.6.1 DiscretePhoton

DiscretePhoton encoder is basically targeted to real-time / low-latency encoding scenarios.

It's quite constrained 1-pass CBR rate-control and baseline profile features might have hurt quality measurements.

Anyway, MSU's test is a very precious experience for us. Thank you.

4.4.6.2 Microsoft Expression Encoder

Microsoft Expression Encoder 4, unlike its previous versions, no longer uses a Microsoft-developed H.264 codec. It uses the MainConcept H.264/AVC Encoder SDK for both its H.264 and AAC encoding. The latest version of Expression Encoder 4 (SP1) also supports CUDA-accelerated H.264 encoding, also through the MainConcept CUDA H.264/AVC Encoder SDK. Consequently, any differences in results presented in this paper between the MainConcept H.264 Encoder and Microsoft Expression Encoder are actually due to the differences in encoding presets (as Microsoft and MainConcept were each asked to supply their own presets), rather than in the underlying compression engine. When configured the same, both encoders should produce nearly identical results.

The long startup times noted in this paper are due to the fact that Microsoft Expression Encoder, both its frontend application and its SDK, are built on the .NET framework. The advantage of a managed code (e.g. C#) SDK, as opposed to a native code (e.g. C++) SDK, is that it allows easy development and quick deployment of applications. For example, Expression Encoder 4 comes with a number of sample C# and VB.NET applications which demonstrate how to automate various encoding scenarios - and most of them can be modified, compiled and deployed in a matter of minutes. A known *downside of managed code applications, however, is their longer startup time. The command-line Expression Encoder app provided to MSU for testing took an average of 5-6 seconds to initialize and start encoding. Due to the short length of most MSU test sources (many were only 10 seconds long), the long startup times added a significant overhead to these short encoding times. It should be noted that in a typical encoding scenario where the source content is several minutes or hours long, a startup time of 5-6 seconds would actually represent a negligible overhead to the total encoding time.*

4.4.6.3 WebM vp8 Codec

In a typical videoconferencing setup, video is captured from a device (typically a camera) with video frames being captured in fixed time intervals. For example, capturing a video at 10 frame/second means that video frames are coming from capturing device at time sequences of 0ms, 100ms, 200ms, 300ms, and so on.

To maintain audio video sync (AV-sync) in video conference, latency has to be kept very low at all time in the encoding process. Encoding speed is important in keeping the latency low, but high encoding speed only is not enough. For example:

1. *No frame reordering in encoder is allowed: An H.264 encoder using 3 B-frames will require encoder/decoder reorder input frames and therefore cause at least a 4 frame delay no matter how fast the encoding speed is.*
2. *No forced delay in encoder is allowed: While frame-level parallel processing is helpful for offline transcoding, it is useless for video conferencing due to the low latency requirement. For example, even if an encoder can speed up encoding by 8X on an 8 core machine, video conferencing applications can not wait for 8 input frames of latency to make use of the parallel processing.*

Most of the H.264 encoder settings used in the videoconferencing test use B frames, and some even include frame-level parallel encoding. Those settings are invalid for a videoconferencing usage case.

In contrast, VP8 does not have B frames, therefore no frame reordering is ever required in encoding. Also, a VP8 encoder can use parallel processing at the macroblock row level, which improves encoding speed and keeps latency low at the same time.

5 Appendix 1. Hardware and Software Encoders Comparison

5.1 Brief Description

In this work, we analyzed not only software video, but also video codecs that used hardware acceleration. There are two main directions for this type of acceleration: using discrete GPU (basically with help of CUDA) and using SandyBridge Platform.

There were two different use-cases for this comparison:

- HDTV encoding at low bitrates 1-10Mbps
- HDTV encoding at high bitrates 6-18Mbps

In this part of the comparison we have analyzed next encoders:

1. Intel Sandy Bridge Transcoder (hardware)
2. MainConcept CUDA based encoder (hardware)
3. x264
4. Elecard
5. DivX H.264
6. XviD (only for low bitrates)

5.1.1 Hardware Description

During tests we have used next hardware:

- CPU: Intel Core i7-2600 @ 3.4GHz , 4 Cores 8 Threads
- GPU:
    - Geforce GTX 580 (for MainConcept CUDA based encoder)
    - Intel GT1 (for Intel Sandy Bridge Transcoder)
- Memory: 8 Gb RAM
- OS: Windows7 Ultimate 64-bit edition, Service Pack 1

5.2 Results for Low Bitrates Encoding

The following graphs show results for this comparison. RD-curves analysis is not very interesting because of different encoding speed and because of it speed/quality graphs have more information.

5.2.1 RD-curves and Speed/Quality Graphs

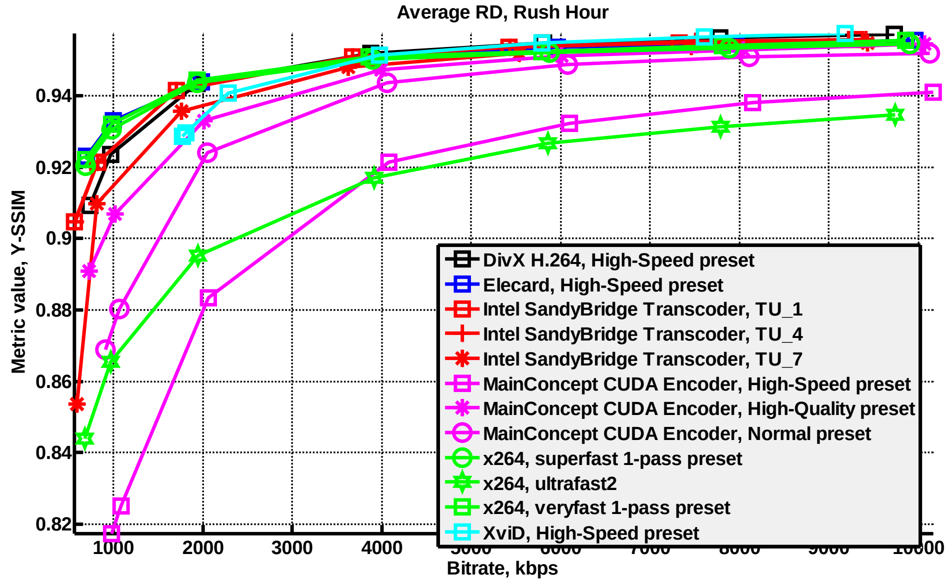


Figure 63. Bitrate/quality, “Rush Hour” sequence, Y-SSIM metric

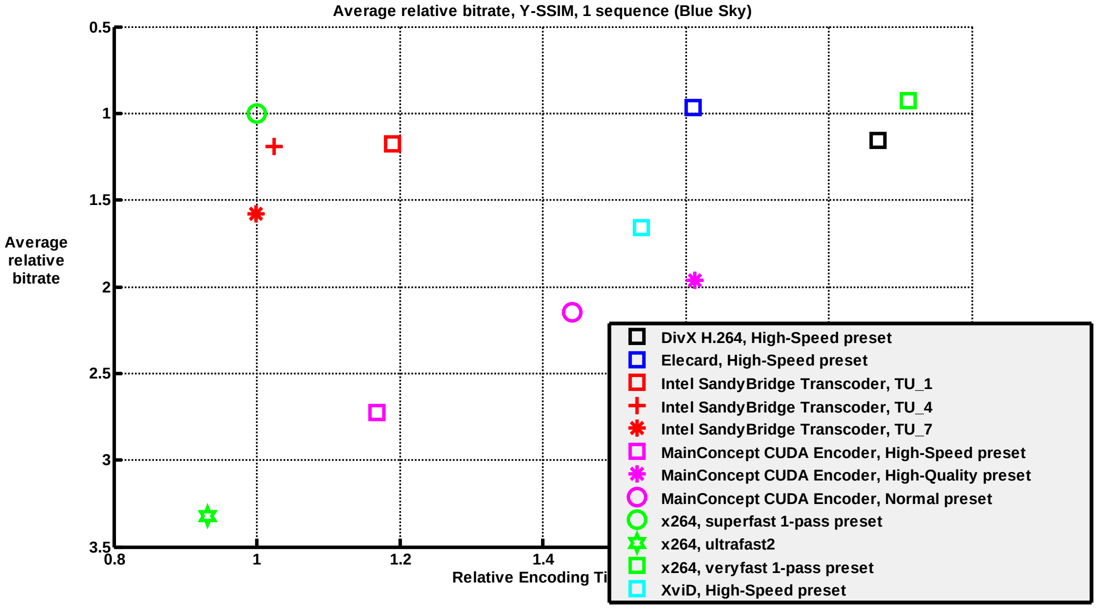


Figure 96. Speed/quality trade-off, “Blue Sky” sequence, Y-SSIM metric

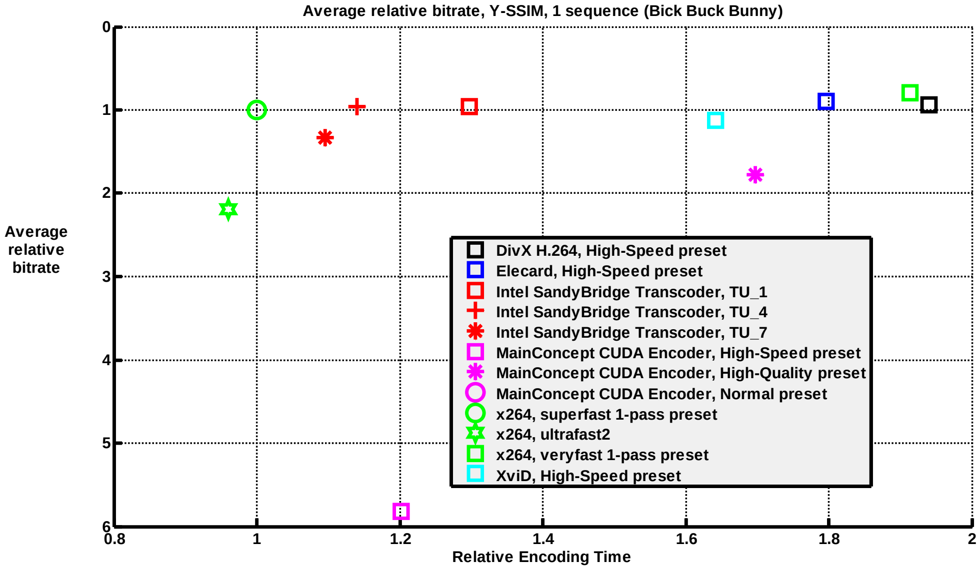


Figure 154. Speed/quality trade-off, “Big Buck Bunny” sequence, Y-SSIM metric

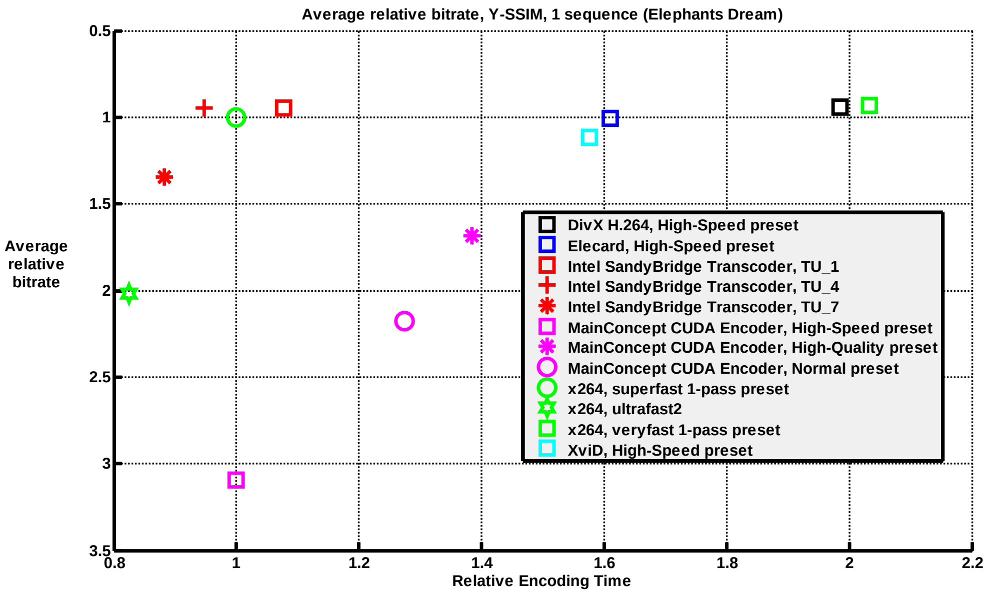


Figure 155. Speed/quality trade-off, “Elephants Dream” sequence, Y-SSIM metric

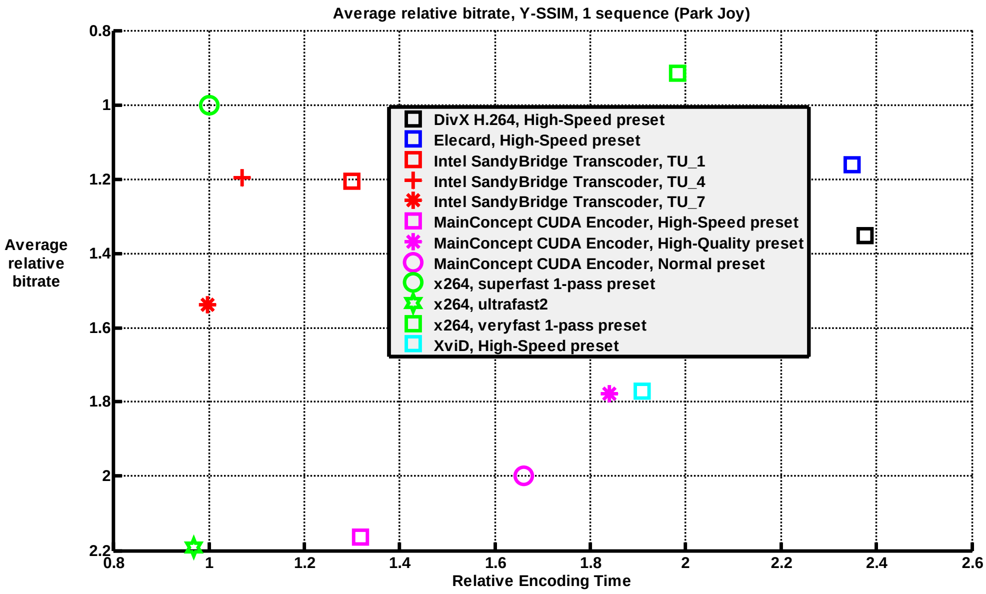


Figure 156. Speed/quality trade-off, “Park Joy” sequence, Y-SSIM metric

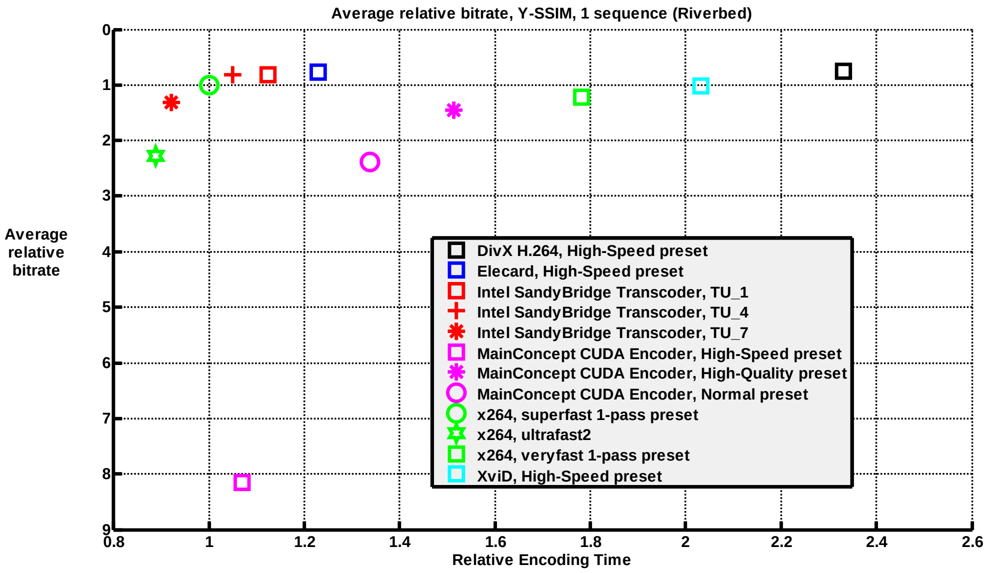


Figure 157. Speed/quality trade-off, “Riverbed” sequence, Y-SSIM metric

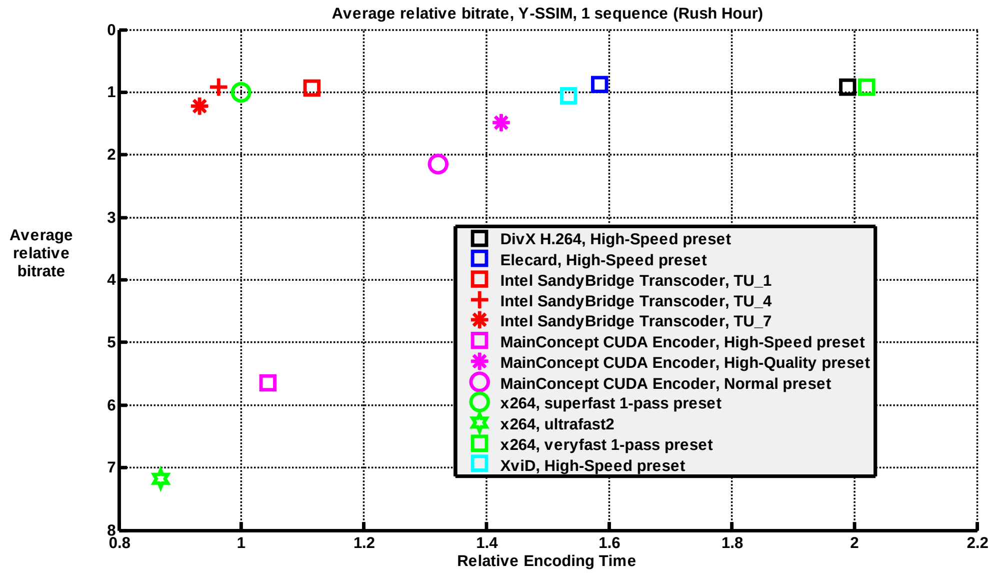


Figure 158. Speed/quality trade-off, "Rush Hour" sequence, Y-SSIM metric

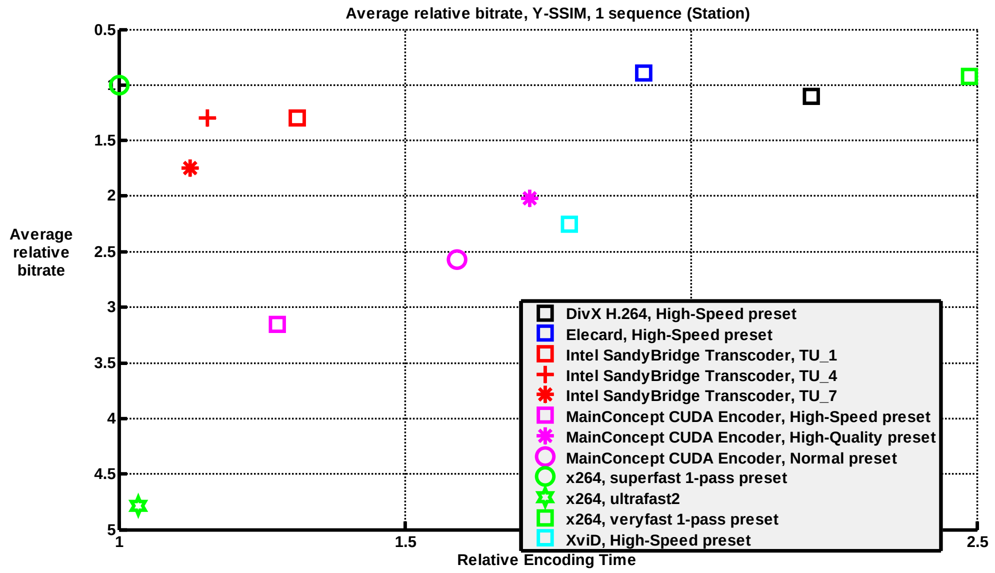


Figure 159. Speed/quality trade-off, "Station" sequence, Y-SSIM metric

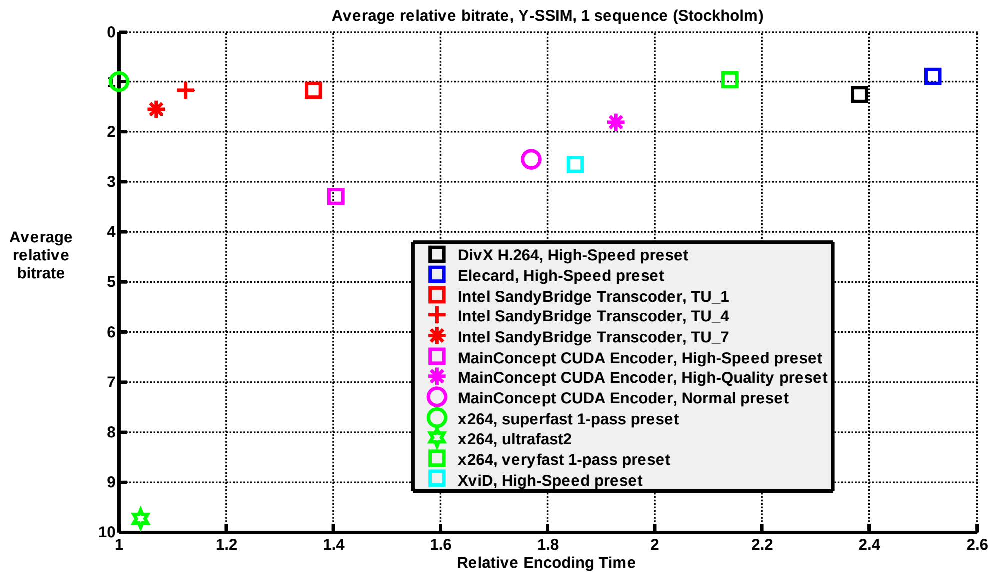


Figure 160. Speed/quality trade-off, "Stockholm" sequence, Y-SSIM metric

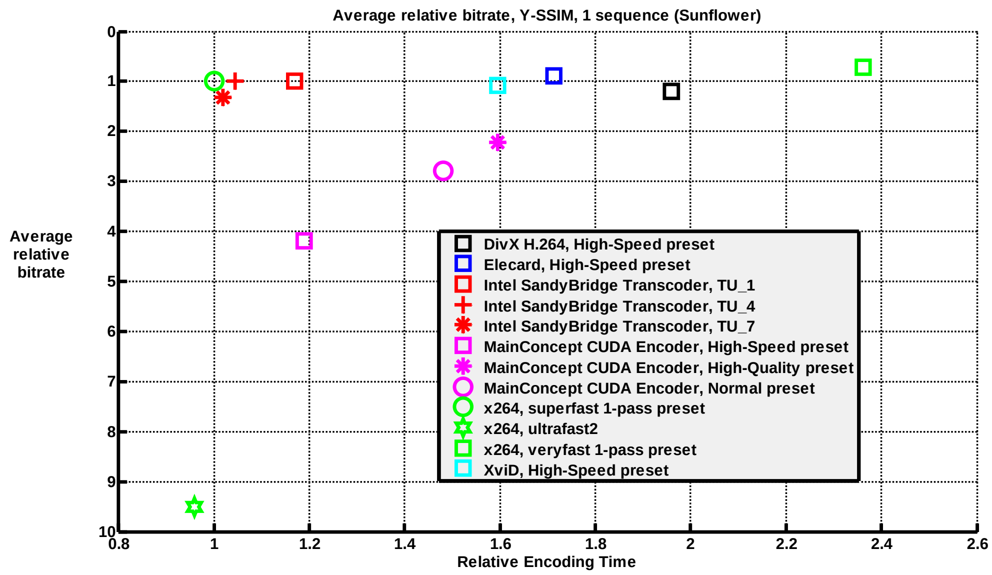


Figure 161. Speed/quality trade-off, "Sunflower" sequence, Y-SSIM metric

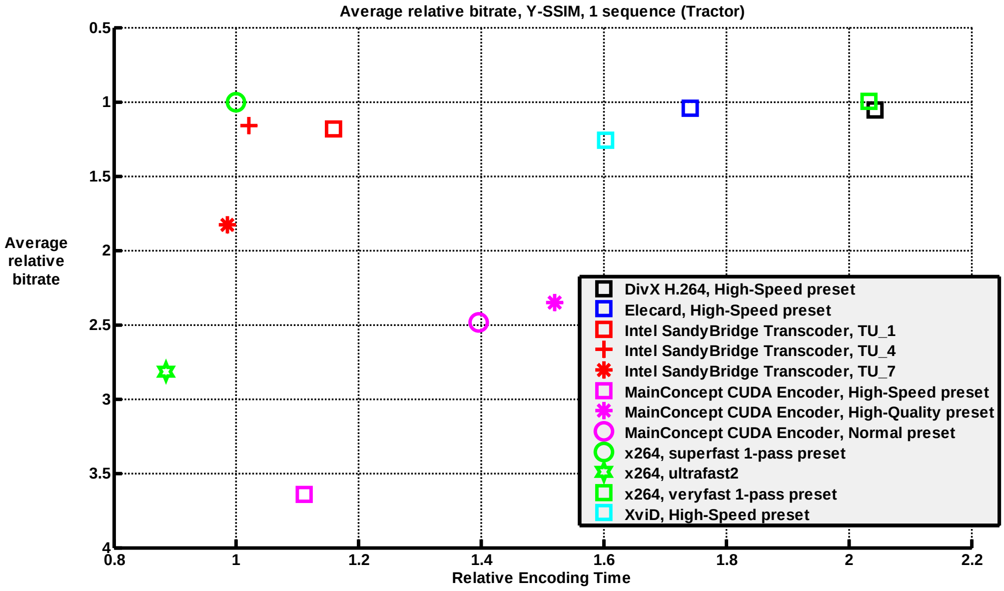


Figure 162. Speed/quality trade-off, "Tractor" sequence, Y-SSIM metric

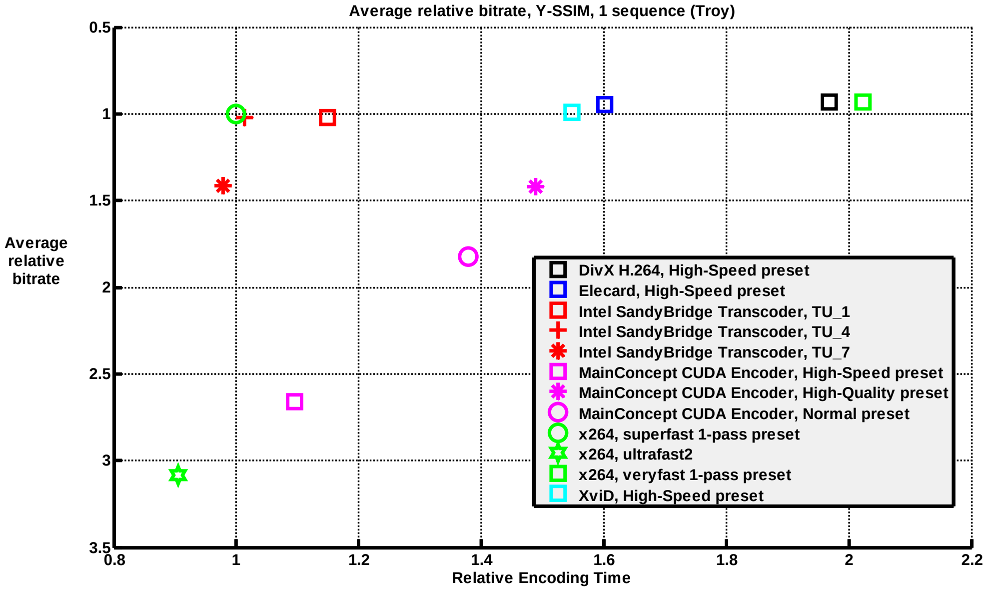


Figure 163. Speed/quality trade-off, "Troy" sequence, Y-SSIM metric

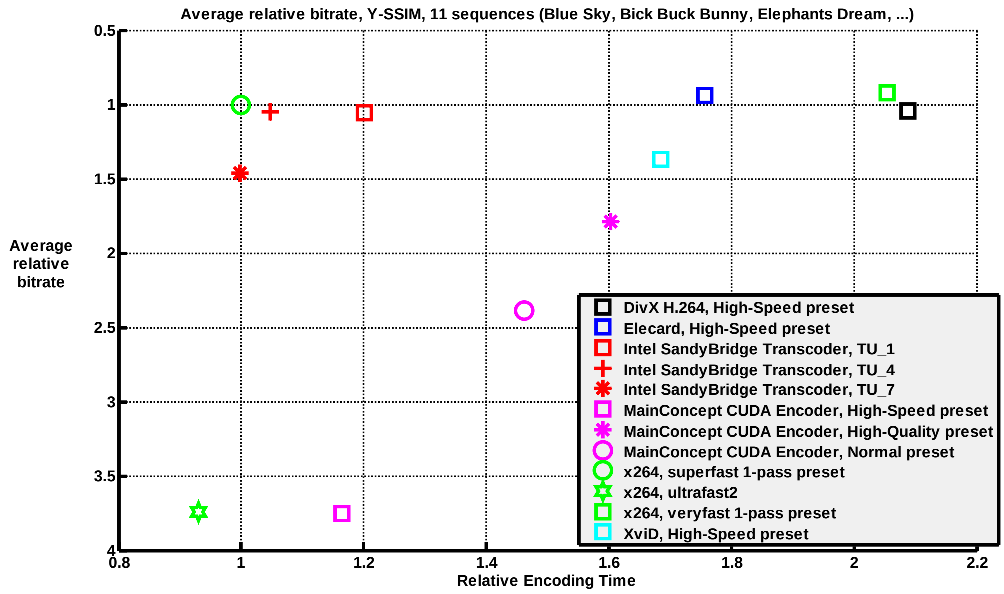


Figure 164. Speed/quality trade-off, all sequences, Y-SSIM metric

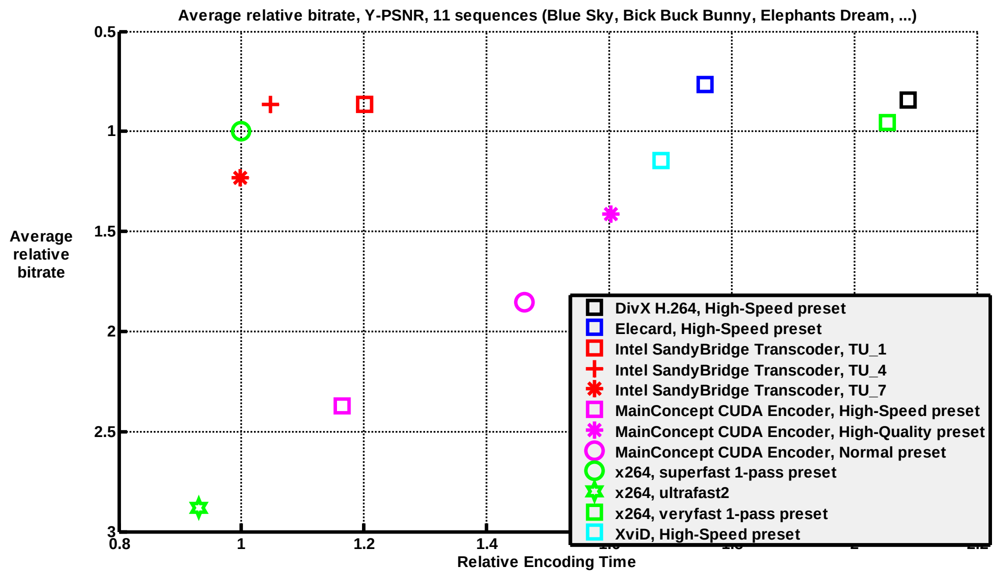


Figure 165. Speed/quality trade-off, all sequences, Y-PSNR metric

5.2.2 Encoding Speed

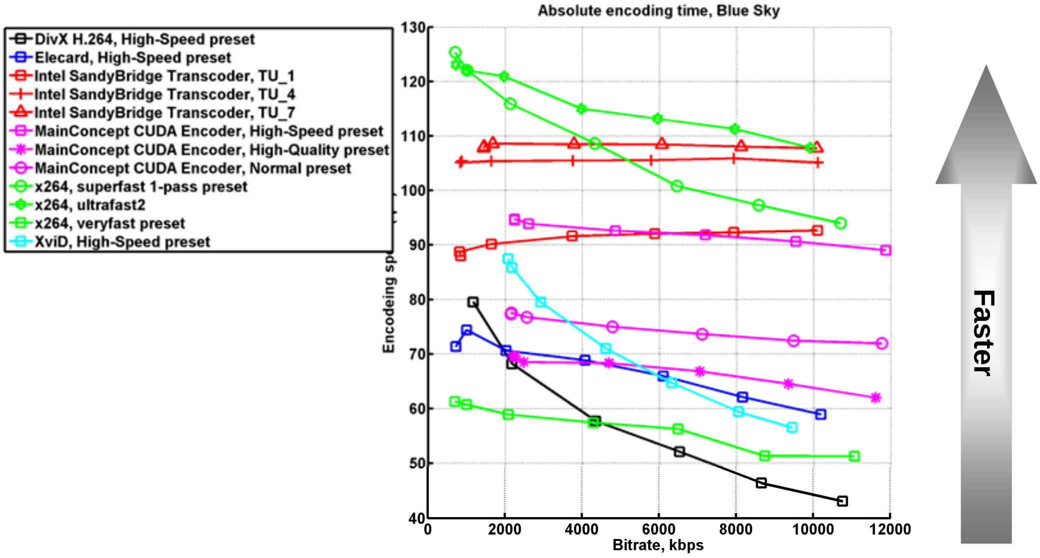


Figure 166. Encoding speed, "Blue Sky" sequence

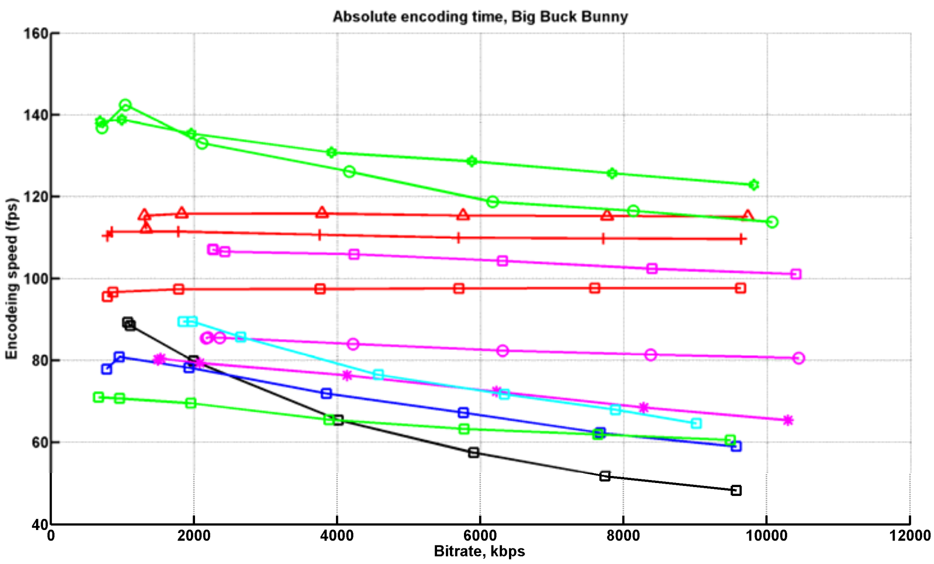


Figure 167. Encoding speed, "Big Buck Bunny" sequence

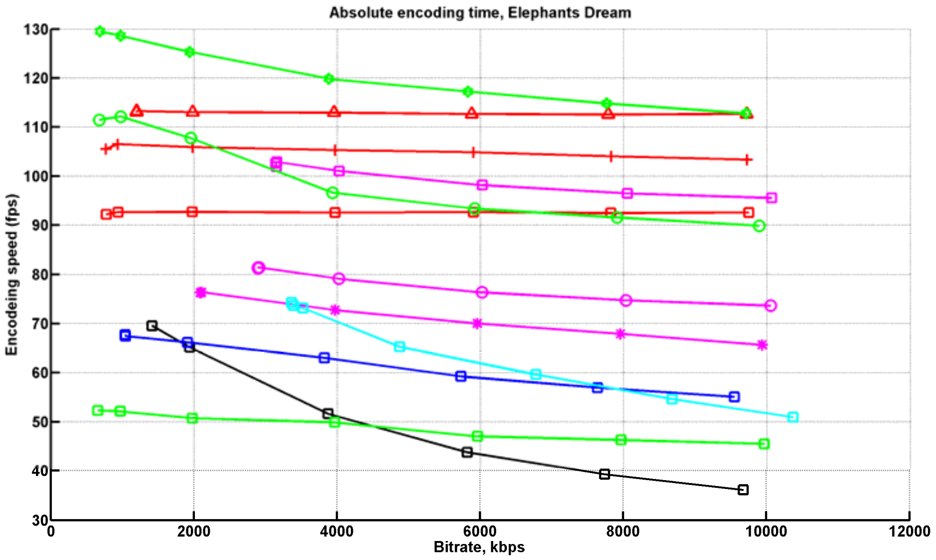


Figure 168. Encoding speed, "Elephants Dream" sequence

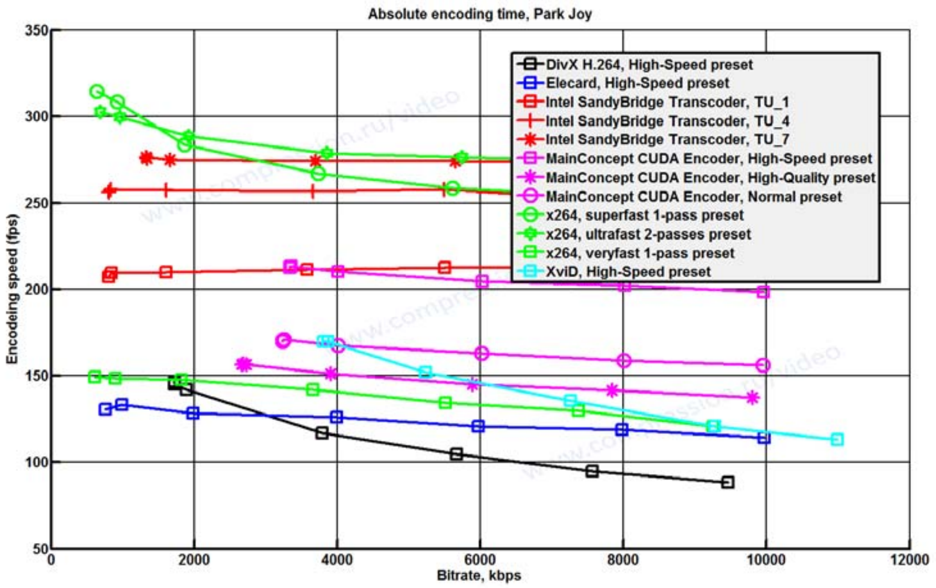


Figure 169. Encoding speed, "Park Joy" sequence

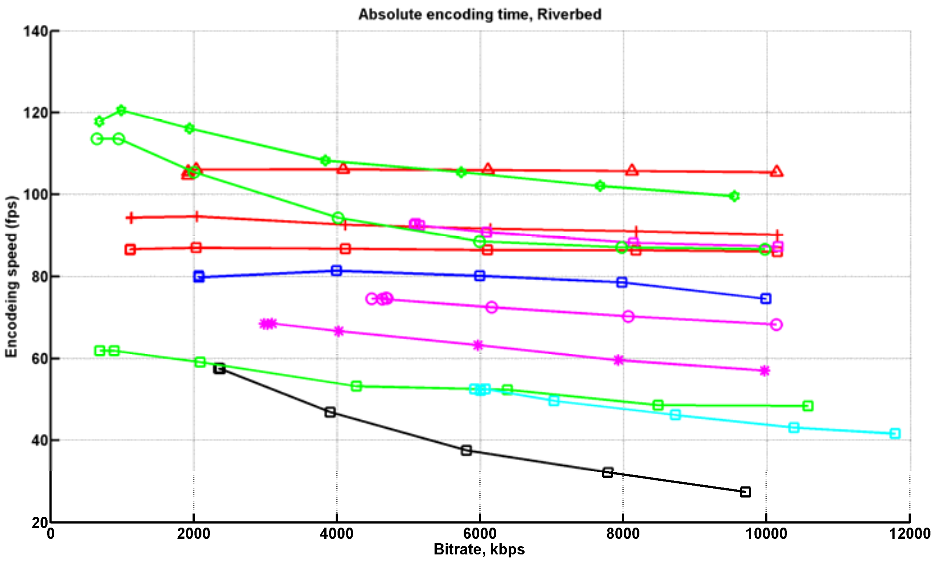


Figure 170. Encoding speed, "Riverbed" sequence

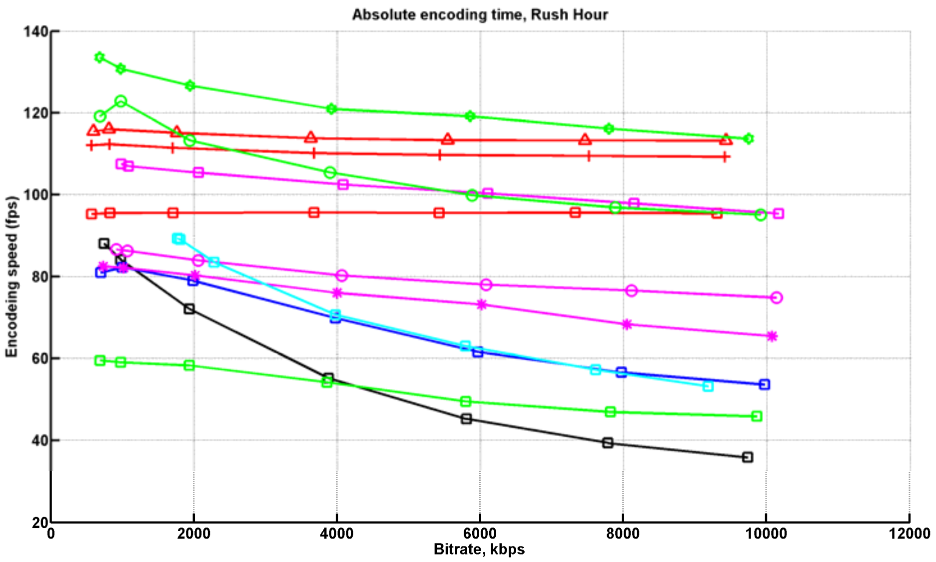


Figure 171. Encoding speed, "Rush Hour" sequence

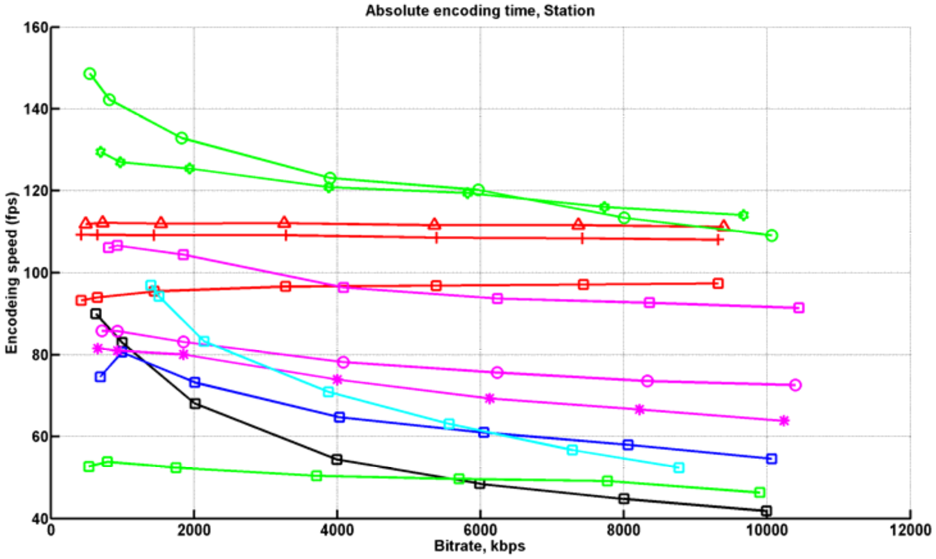


Figure 172. Encoding speed, "Station" sequence

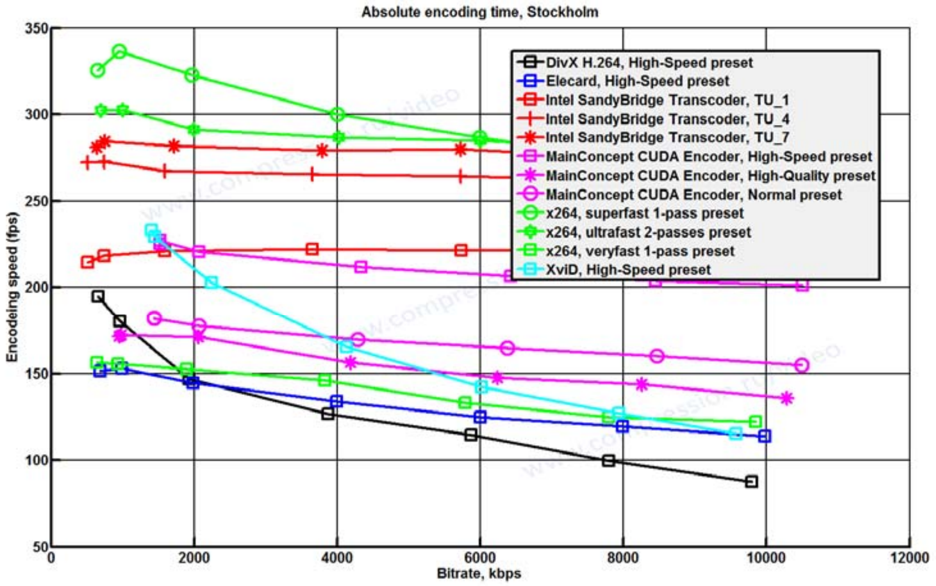


Figure 173. Encoding speed, "Stockholm" sequence

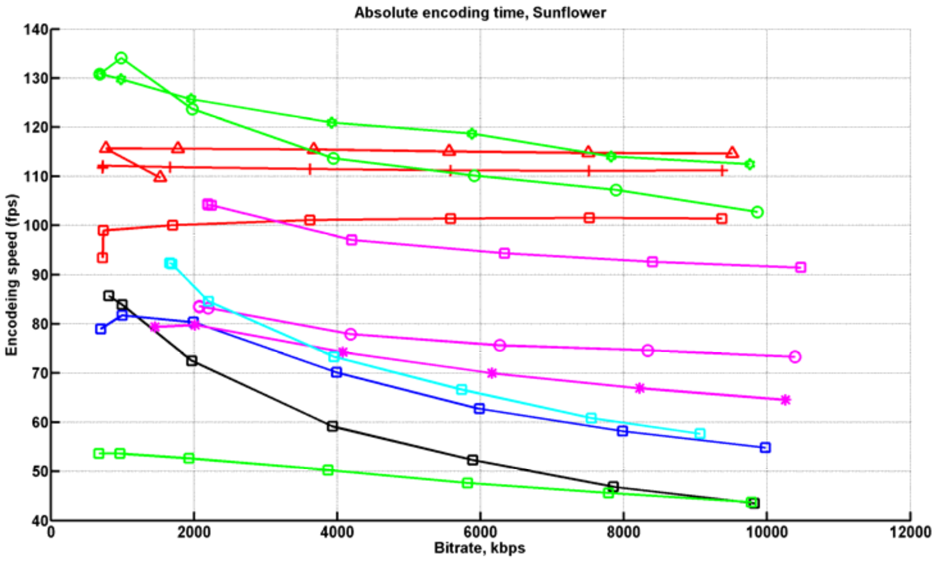


Figure 174. Encoding speed, “Sunflower” sequence

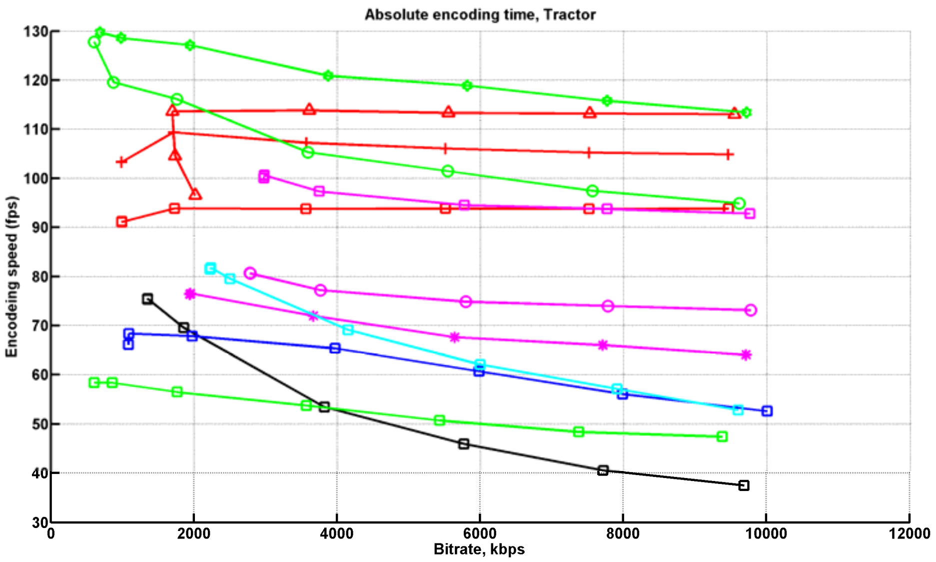


Figure 175. Encoding speed, “Tractor” sequence

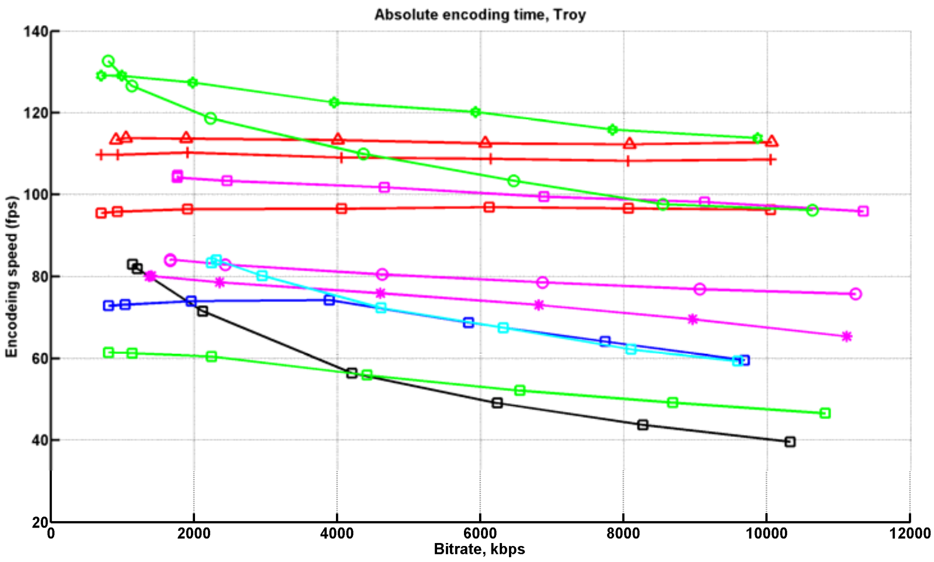


Figure 176. Encoding speed, "Troy" sequence

5.3 Results for High Bitrates Encoding

The following graphs show results for this comparison. Some presets are missing on some speed/quality graphs due to its low encoding quality and low RD-curve.

5.3.1 RD-curves and Speed/Quality Graphs

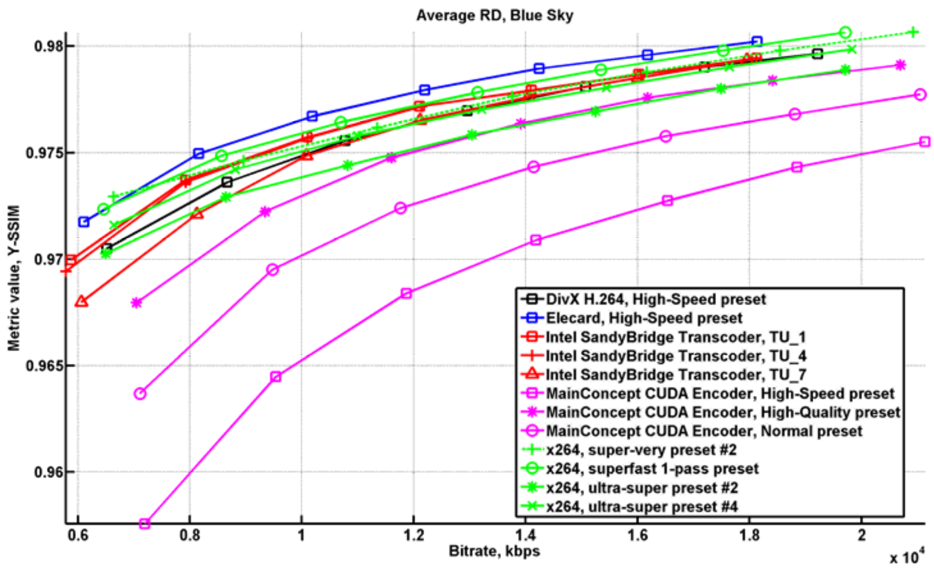


Figure 177. Bitrate/quality, "Blue Sky" sequence, Y-SSIM metric

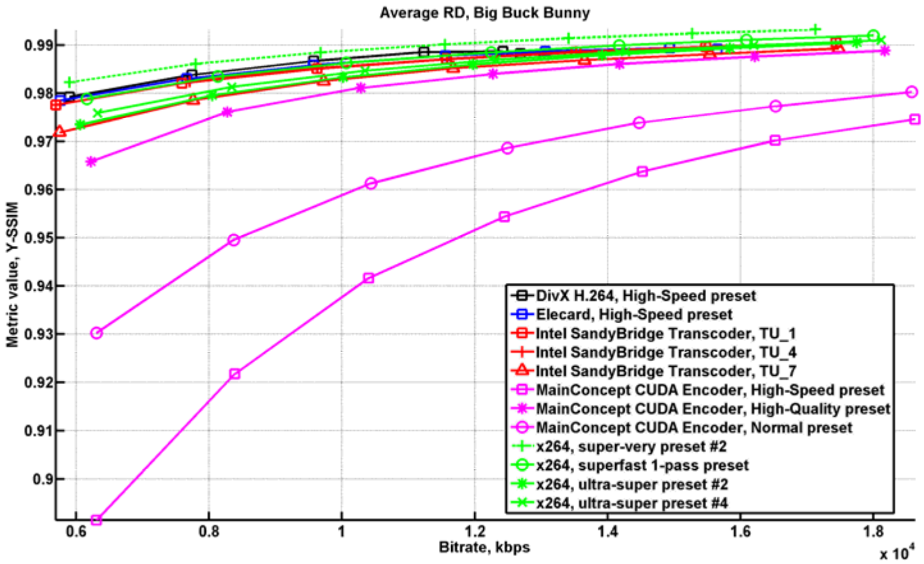


Figure 178. Bitrate/quality, "Big Buck Bunny" sequence, Y-SSIM metric

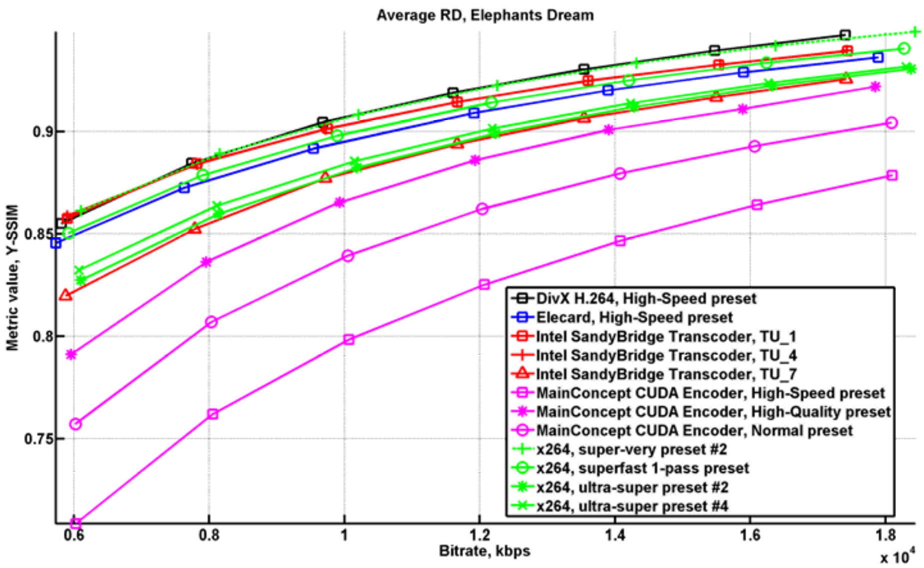


Figure 179. Bitrate/quality, "Elephants Dream" sequence, Y-SSIM metric

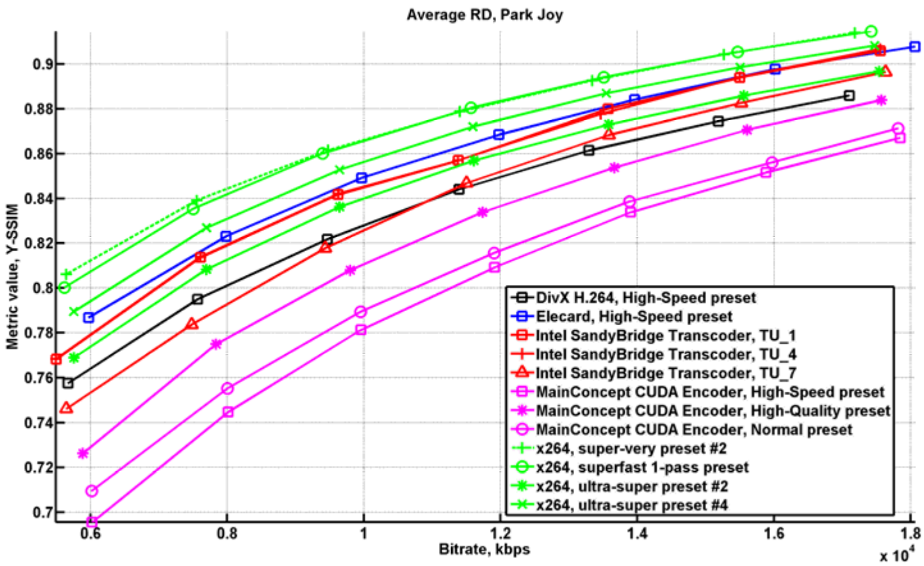


Figure 180. Bitrate/quality, “Park Joy” sequence, Y-SSIM metric

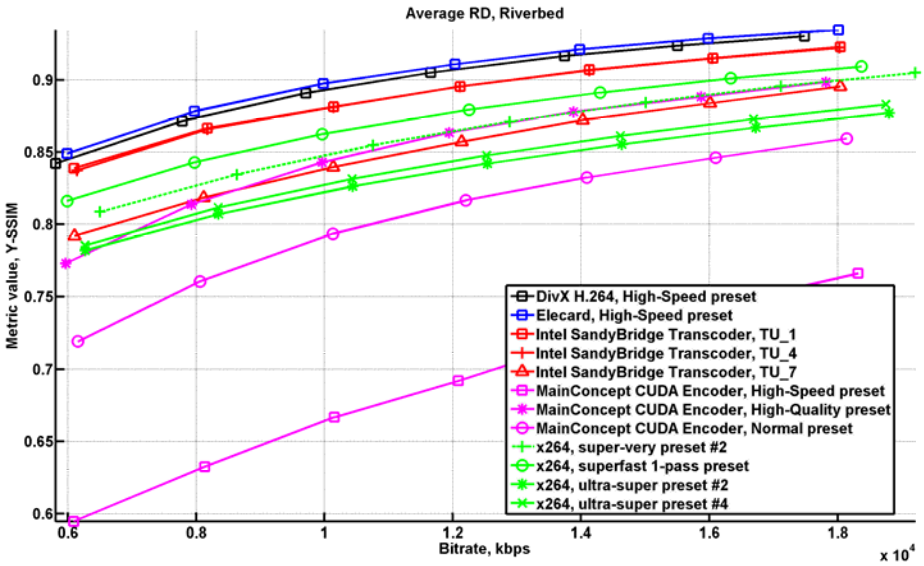


Figure 181. Bitrate/quality, “Riverbed” sequence, Y-SSIM metric

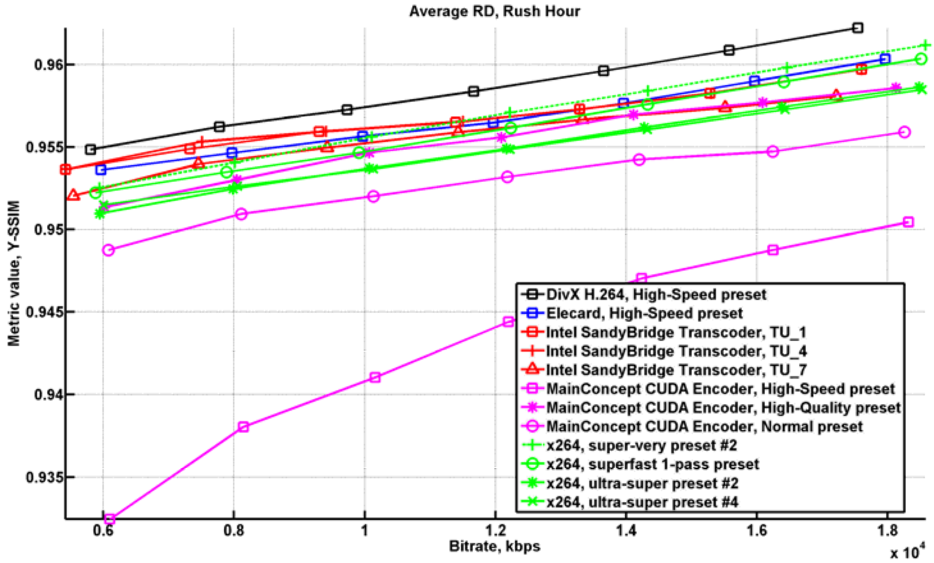


Figure 182. Bitrate/quality, "Rush Hour" sequence, Y-SSIM metric

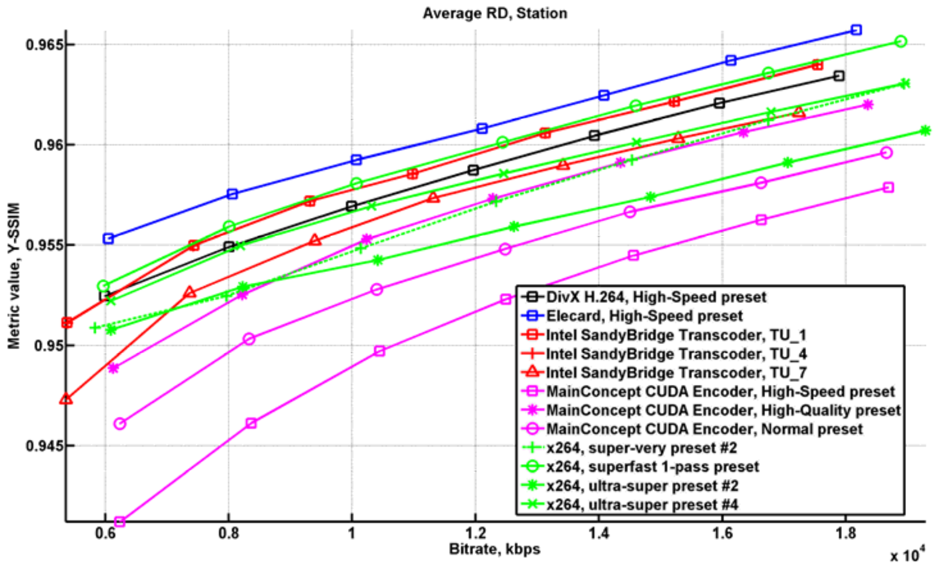


Figure 183. Bitrate/quality, "Stattion" sequence, Y-SSIM metric

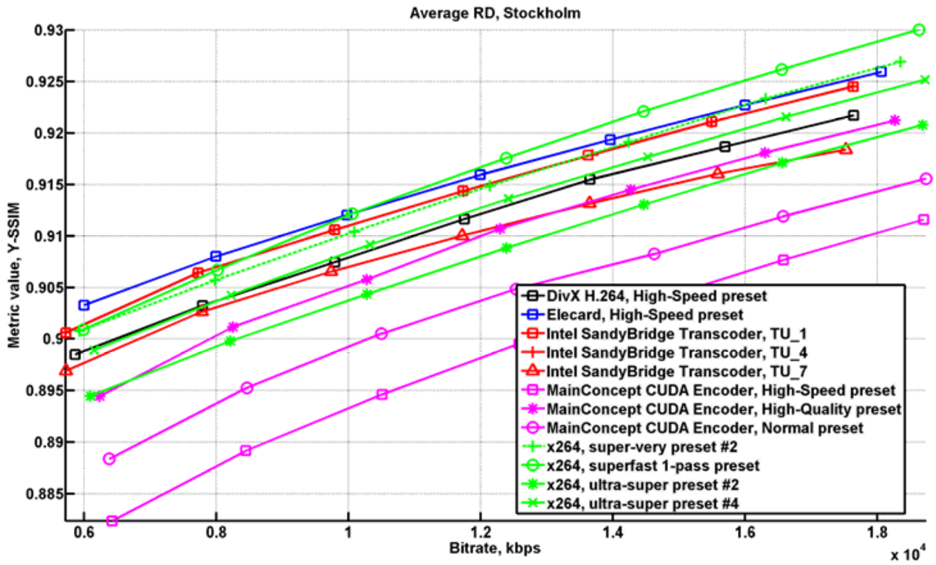


Figure 184. Bitrate/quality, "Stockholm" sequence, Y-SSIM metric

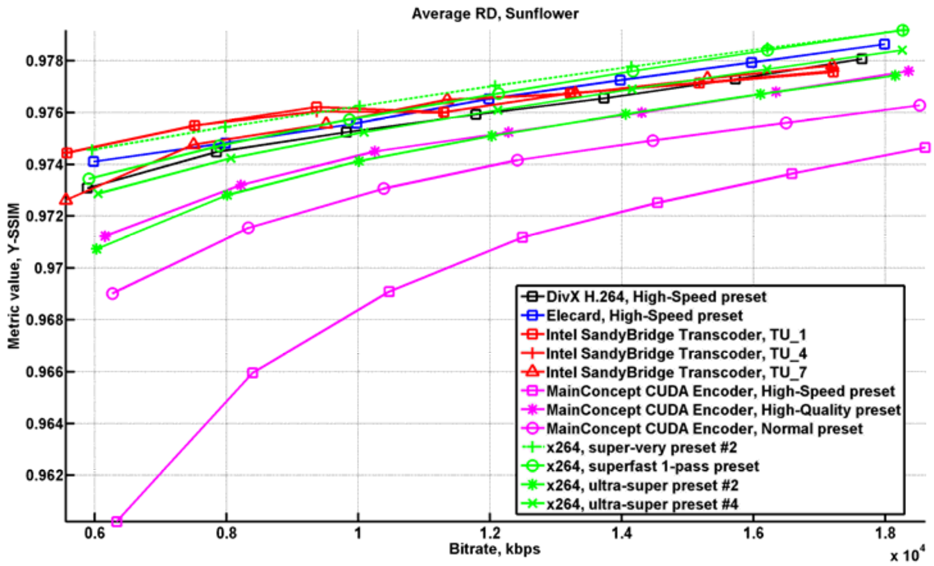


Figure 185. Bitrate/quality, "Sunflower" sequence, Y-SSIM metric

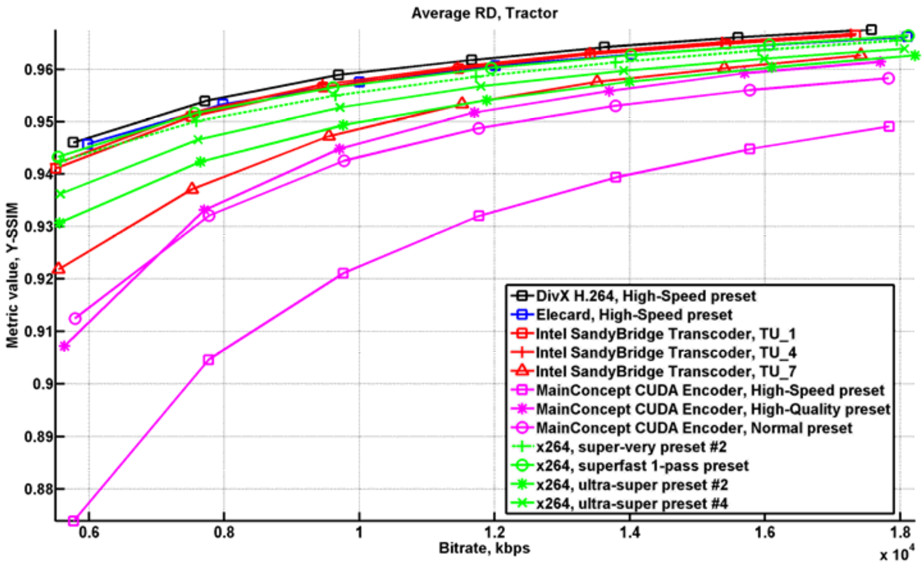


Figure 186. Bitrate/quality, “Tractor” sequence, Y-SSIM metric

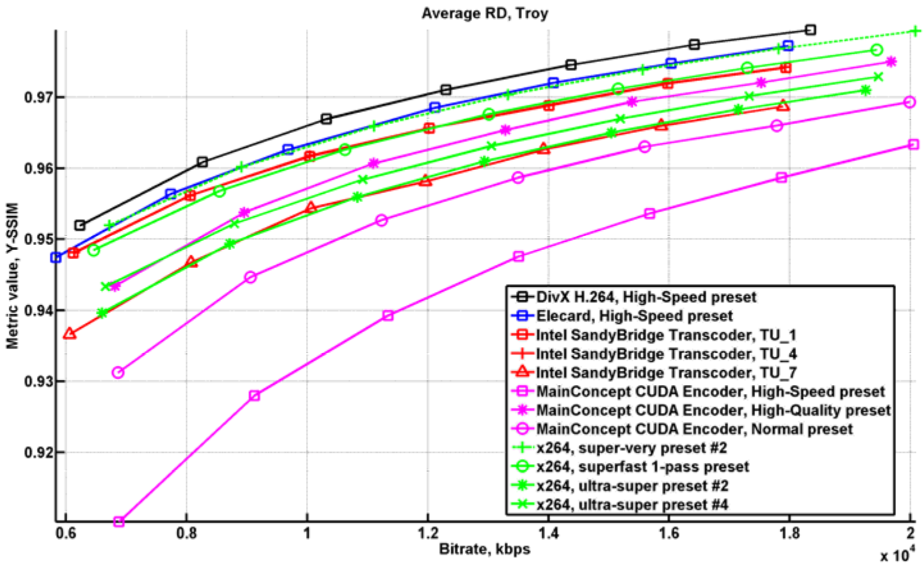


Figure 187. Bitrate/quality, “Troy” sequence, Y-SSIM metric

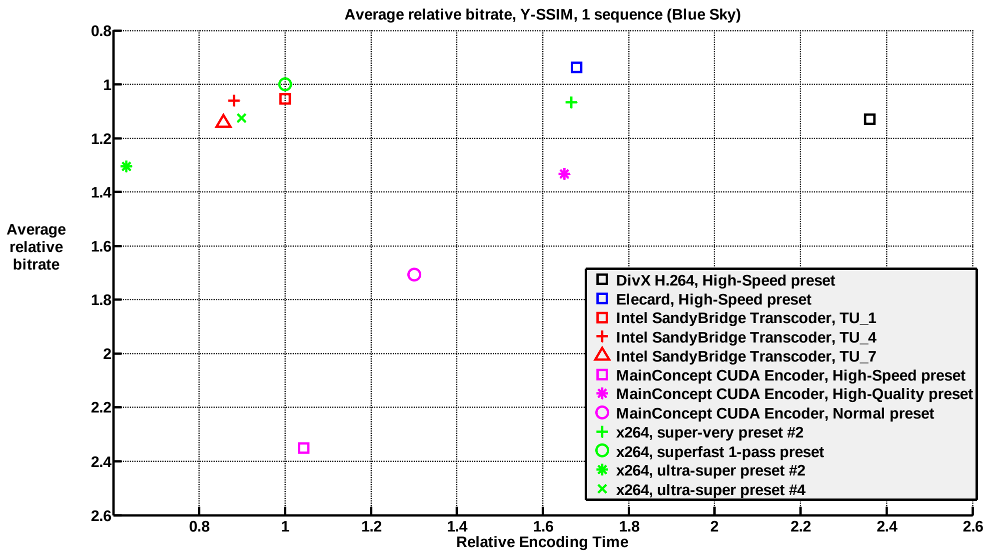


Figure 96. Speed/quality trade-off, "Blue Sky" sequence, Y-SSIM metric

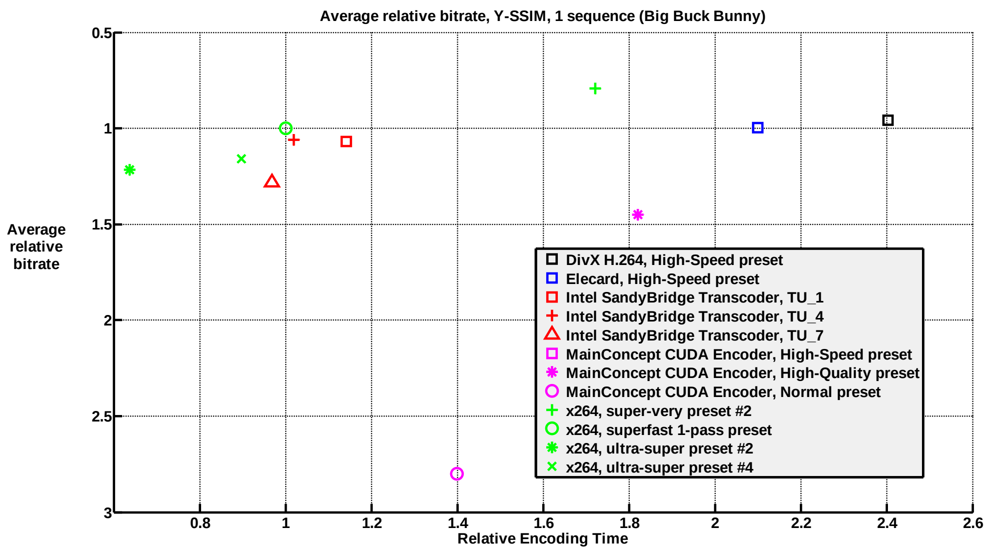


Figure 188. Speed/quality trade-off, "Big Buck Bunny" sequence, Y-SSIM metric

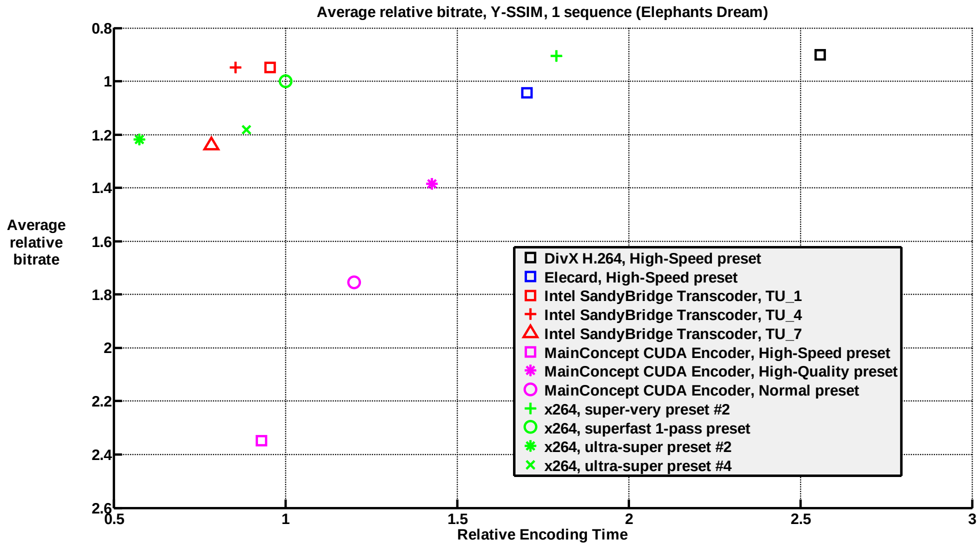


Figure 189. Speed/quality trade-off, "Elephants Dream" sequence, Y-SSIM metric

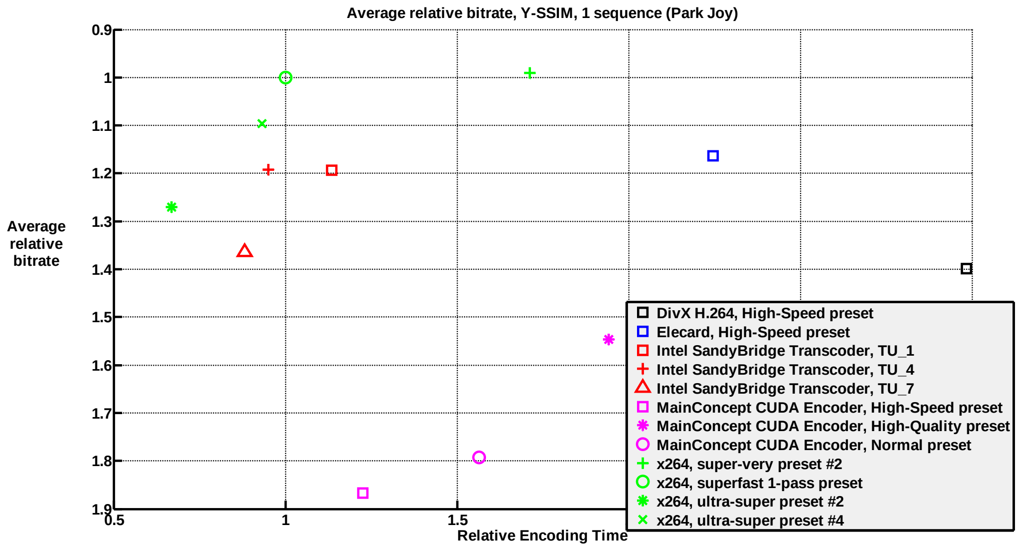


Figure 190. Speed/quality trade-off, "Park Joy" sequence, Y-SSIM metric

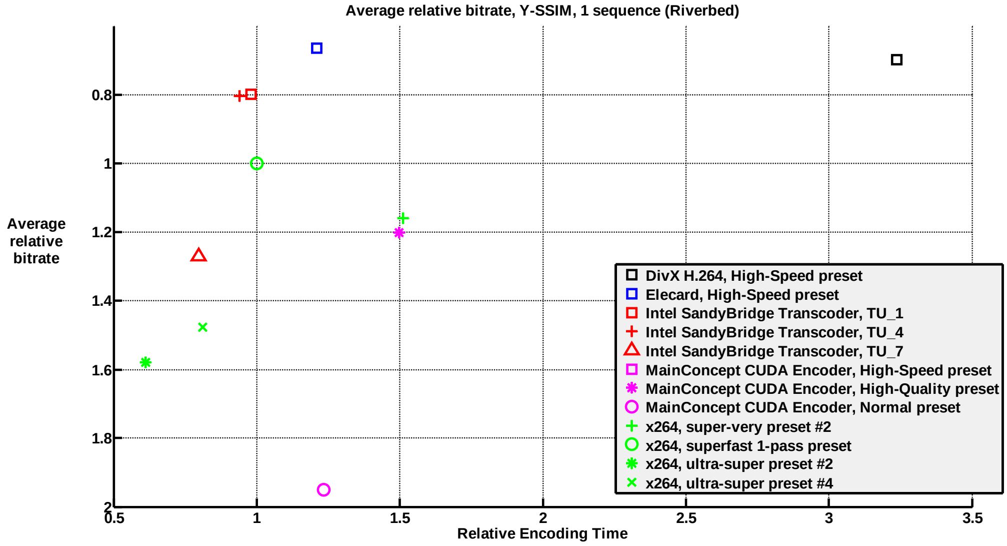


Figure 191. Speed/quality trade-off, “Riverbed” sequence, Y-SSIM metric

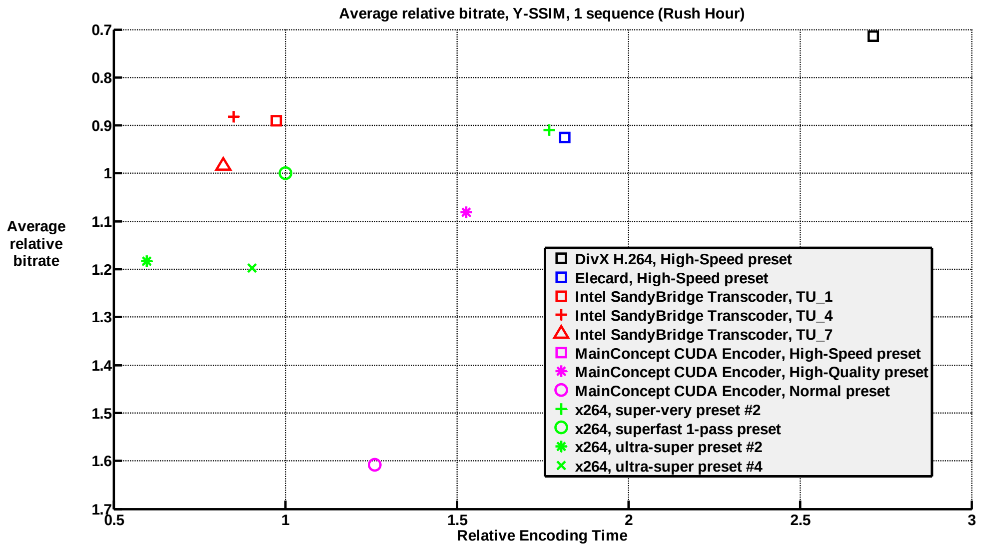


Figure 192. Speed/quality trade-off, “Rush Hour” sequence, Y-SSIM metric

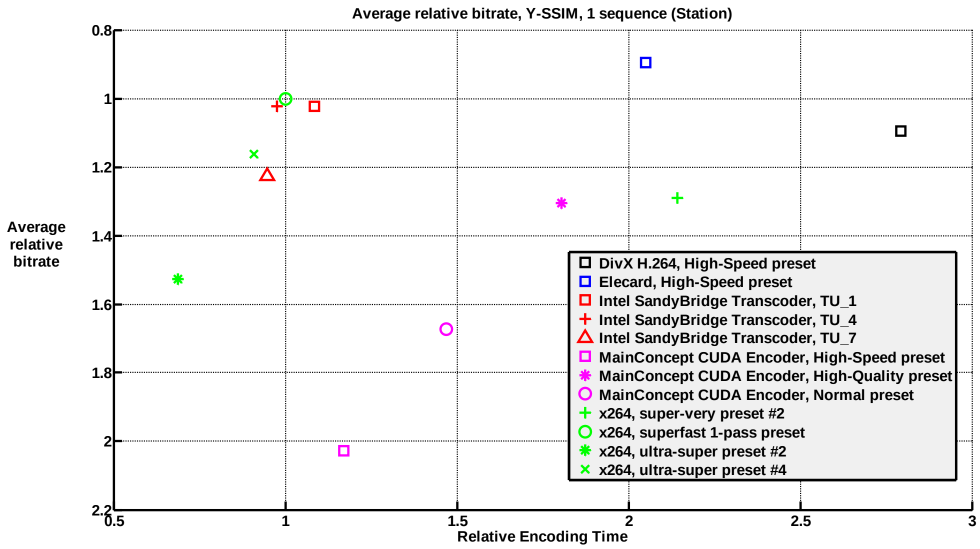


Figure 193. Speed/quality trade-off, “Station” sequence, Y-SSIM metric

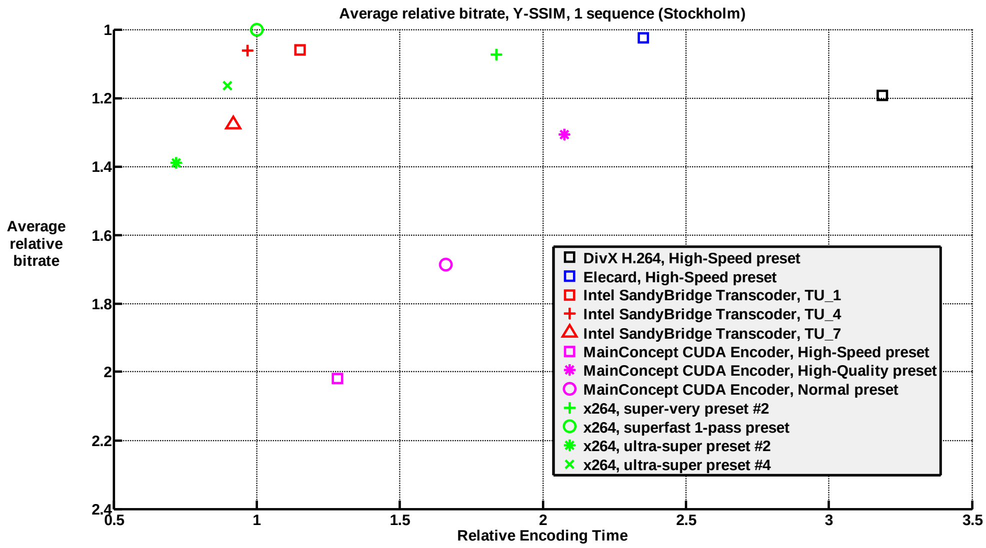


Figure 194. Speed/quality trade-off, “Stockholm” sequence, Y-SSIM metric

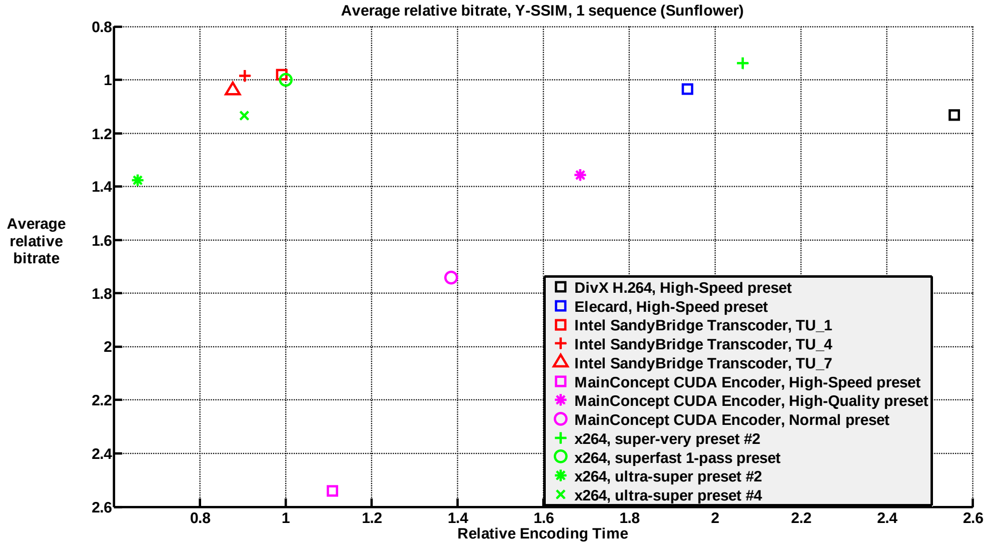


Figure 195. Speed/quality trade-off, “Sunflower” sequence, Y-SSIM metric

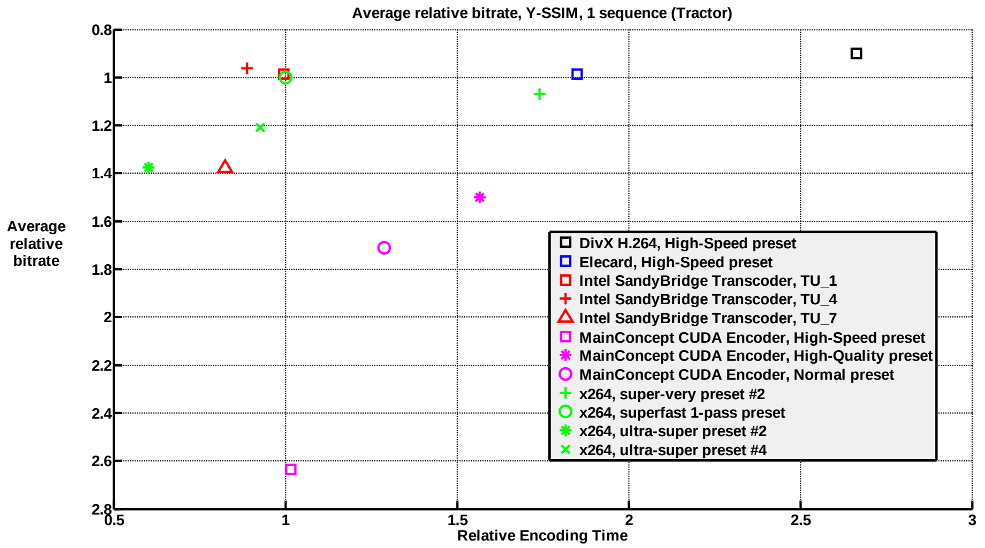


Figure 196. Speed/quality trade-off, “Tractor” sequence, Y-SSIM metric

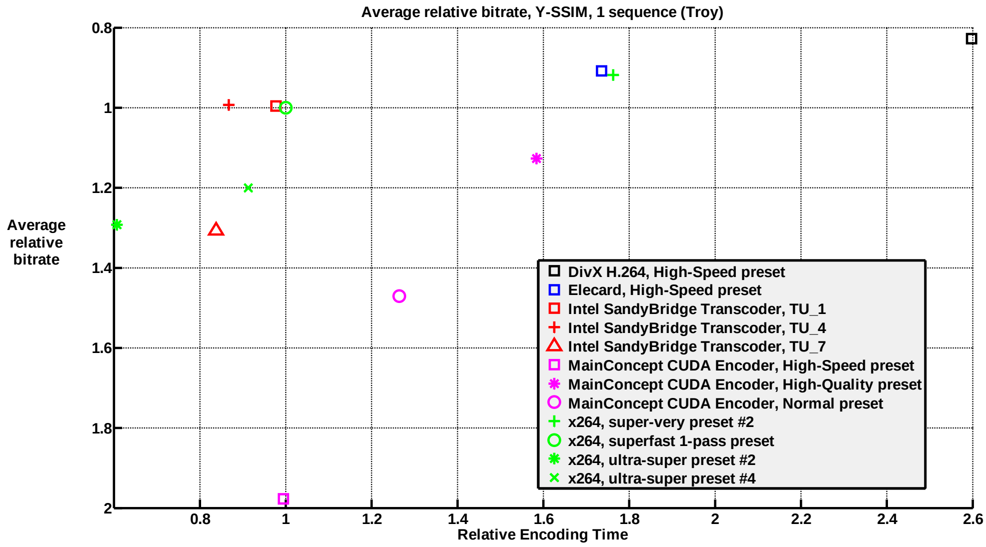


Figure 197. Speed/quality trade-off, "Troy" sequence, Y-SSIM metric

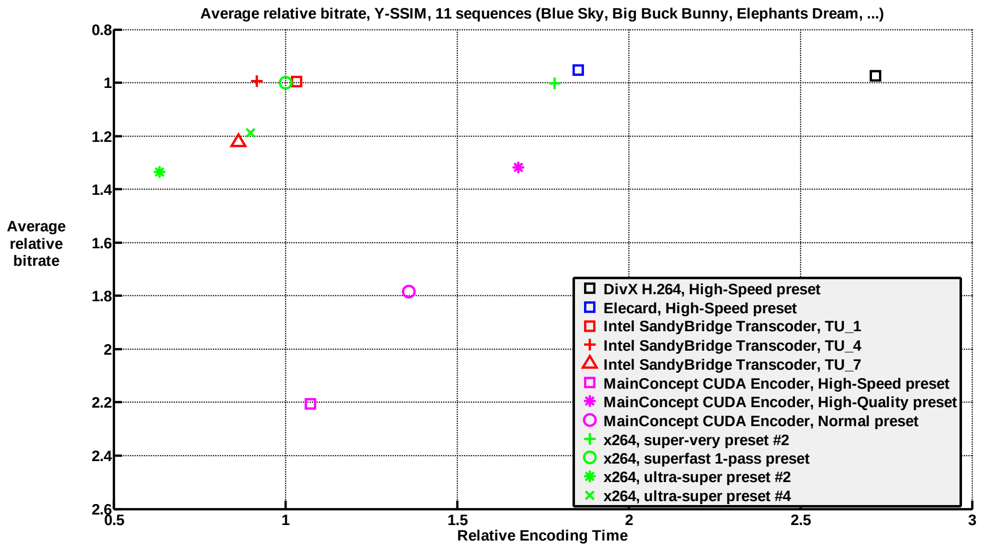


Figure 198. Speed/quality trade-off, all sequences, Y-SSIM metric

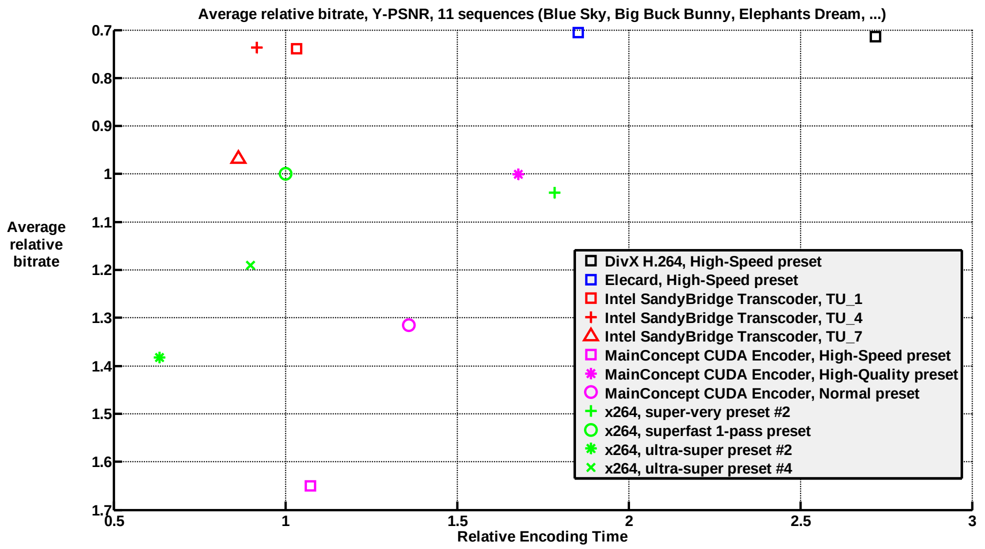


Figure 199. Speed/quality trade-off, all sequences, Y-PSNR metric

5.3.2 Encoding Speed

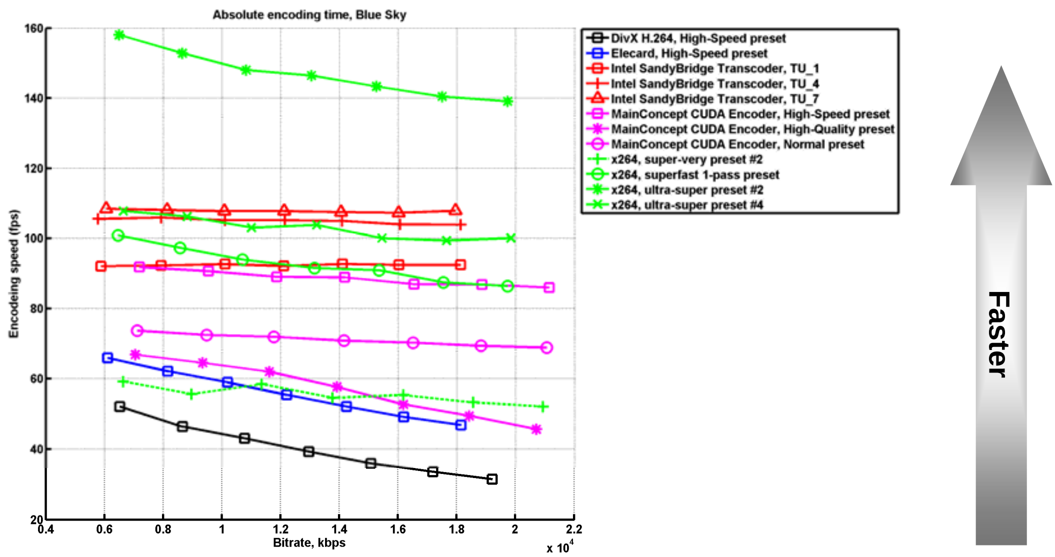


Figure 200. Encoding speed, “Blue Sky” sequence

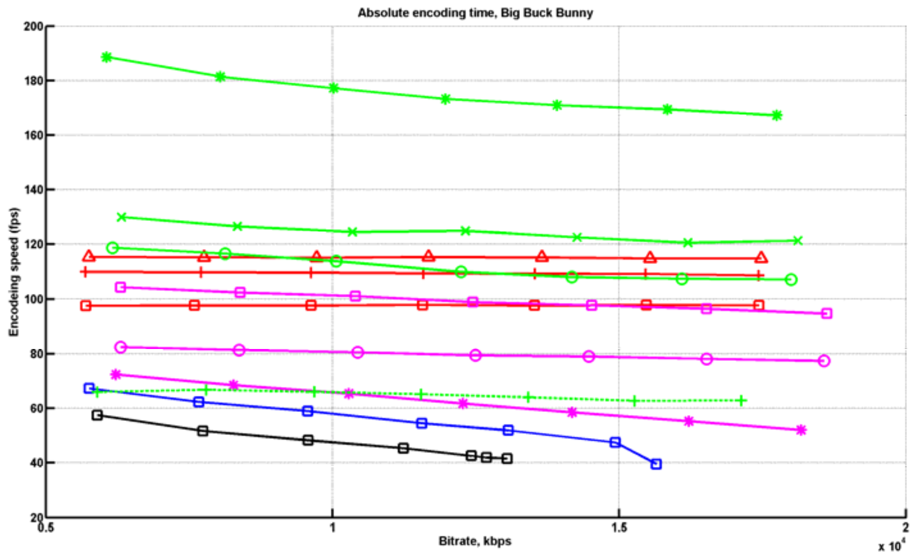


Figure 201. Encoding speed, “Big Buck Bunny” sequence

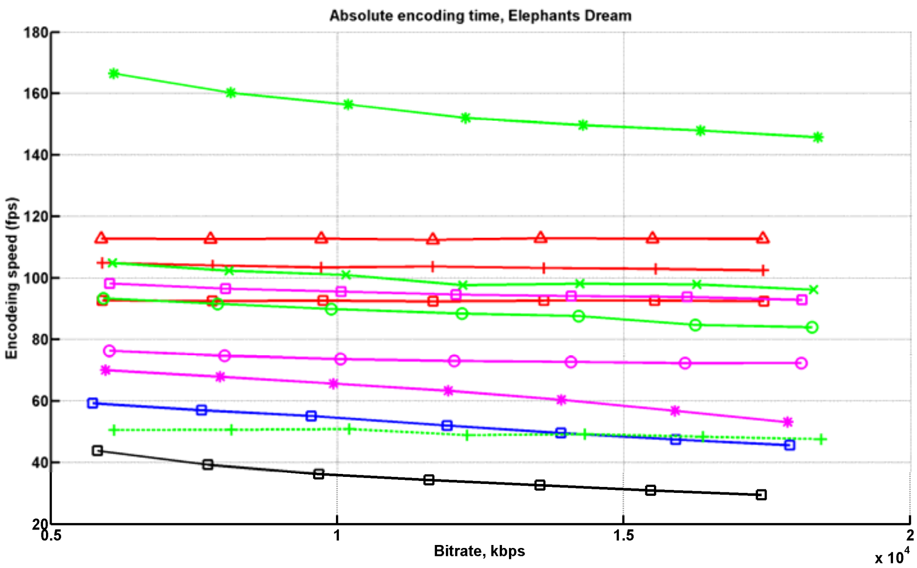


Figure 202. Encoding speed, “Elephants Dream” sequence

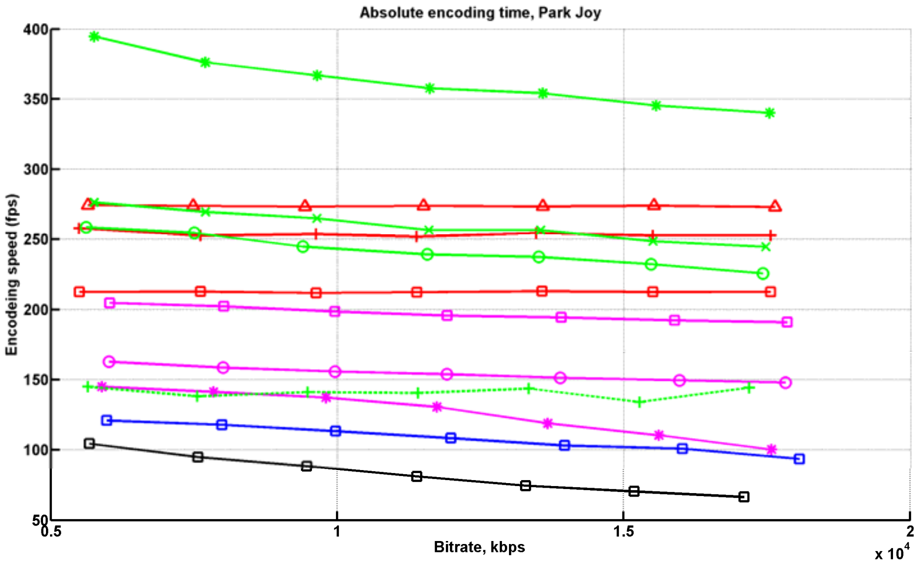


Figure 203. Encoding speed, "Park Joy" sequence

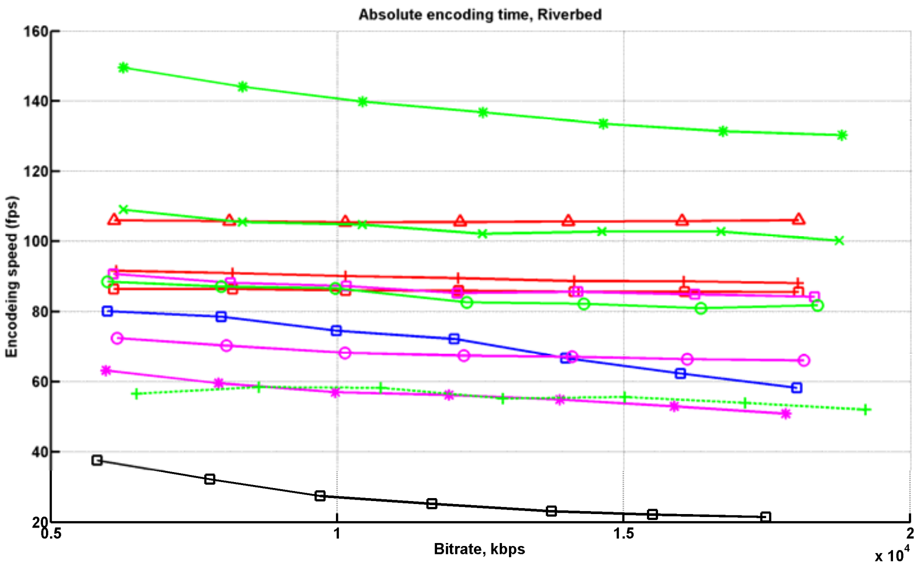


Figure 204. Encoding speed, "Riverbed" sequence

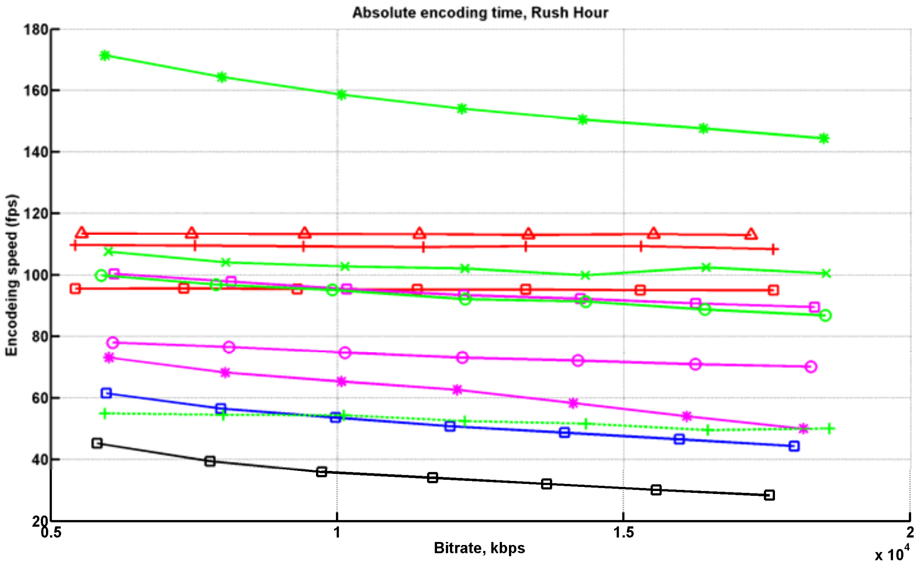


Figure 205. Encoding speed, “Rush Hour” sequence

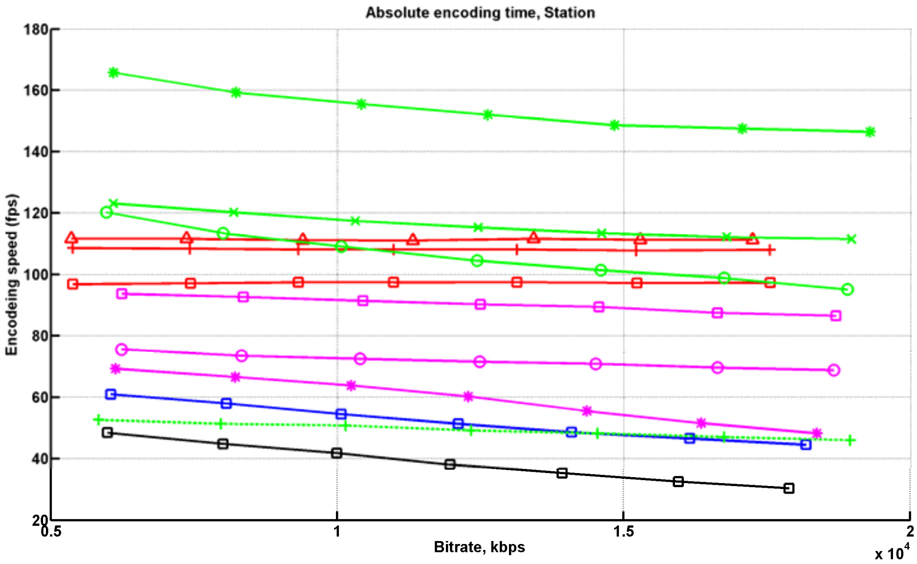


Figure 206. Encoding speed, “Station” sequence

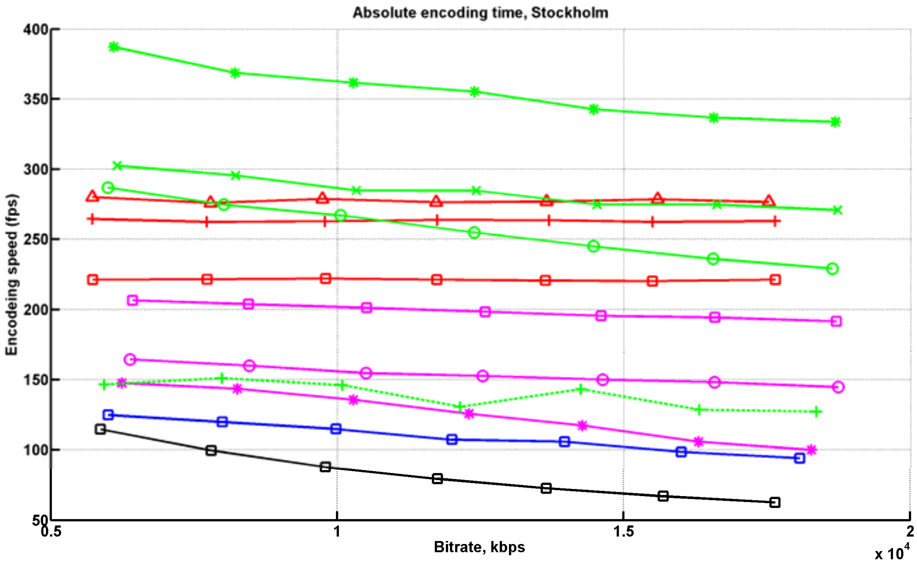


Figure 207. Encoding speed, “Stockholm” sequence

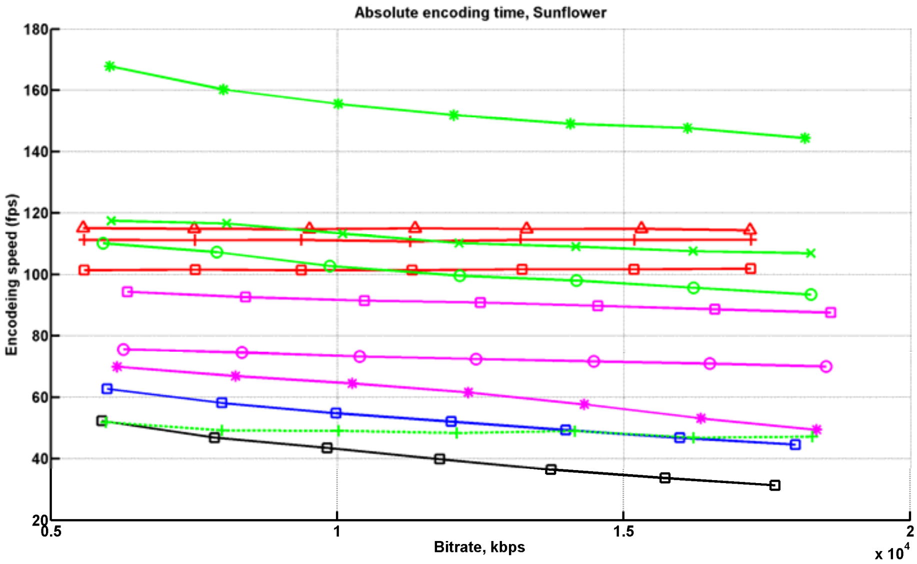


Figure 208. Encoding speed, “Sunflower” sequence

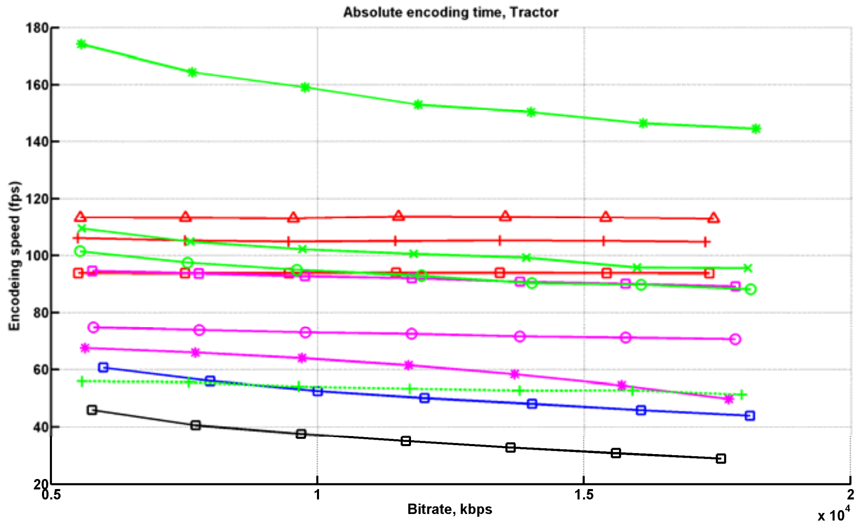


Figure 209. Encoding speed, “Tractor” sequence

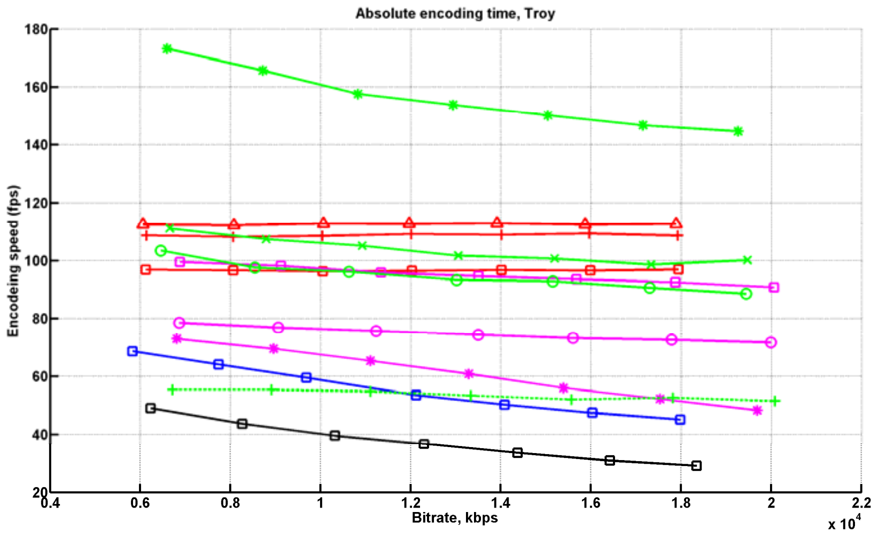


Figure 210. Encoding speed, “Troy” sequence

5.4 Conclusion

Analyzing the graphs for High bitrate and Low bitrate HDTV encoding we could make a conclusion:

1. Software x264 and Intel SandyBridge transcoder are very close to each other
    a. For High Bitrates: software x264 and Intel SandyBridge transcoder are very close to each other, but Intel shows slightly better results than x264 in speed/quality trade-off.
    b. For low Bitrates: at present time software x264 could be best encoder even comparing to hardware-based encoder solution
2. Among hardware solutions Intel SandyBridge transcoder is best.
3. Most software encoders are slower and produce lower quality than Intel SandyBridge transcoder and x264 at very high encoding speed.

5.5 Presets

Codec	Preset Name	Preset
x264	Ultra-super #2	--tune ssim --keyint 500 --preset superfast --weightp 0 --partitions none --no-cabac --no-8x8dct --bframes 0
	Ultra-super #4	--tune ssim --keyint 500 --preset superfast --weightp 0 --partitions none --no-cabac
	Superfast	--tune ssim --keyint 500 --preset superfast
	Super-very #2	--tune ssim --keyint 500 --preset veryfast --partitions i8x8,i4x4
	veryfast	--tune ssim --keyint 500 --preset veryfast
Intel Sandy Bridge Transcoder	Use-case 1	-h264 -hw -d3d -async 10 -s 0 –l 1-u 1 –i:yv12
	Use-case 2	-h264 -hw -d3d -async 10 -s 0 –l 1-u 4 –i:yv12
	Use-case 3	-h264 -hw -d3d -async 10 -s 0 –l 1-u 7 –i:yv12

6 Appendix 2. x264 and WebM Comparison Over Time

6.1 x264 Comparison Over Time

The quality of an H.264 codec, over several years, can be compared for a given video sequence. The x264 encoder was chosen for this task because it is present in almost every MSU VIDEO MPEG-4 AVC/H.264 codec comparison, and it produces good results compared with other encoders. For all years except 2005, x264 shows the best results. For years 2006–2011, we have shown results using Y-SSIM as the quality metric; for 2005, we did not use this as the main metric. In light of these results, x264 could be a good reference encoder for analyzing the overall progress of H.264 encoders over time.

Figure 211 shows the RD curve for the “Battle” sequence using x264 encoders from different years. The best encoder is this year’s x264; the worst is the 2005 version. Using SSIM, the codecs can be ranked as follows:

1. x264 (2011)
2. x264 (2010)
3. x264 (2009)
4. x264 (2007)
5. x264 (2006)
6. x264 (2005)

These results are shown in Figure 212. This figure indicates that the overall progress is very good, and that the x264 encoder has increased in speed and quality over recent years. But the old x264 does not use multithreading, so encoding speed varies considerably.

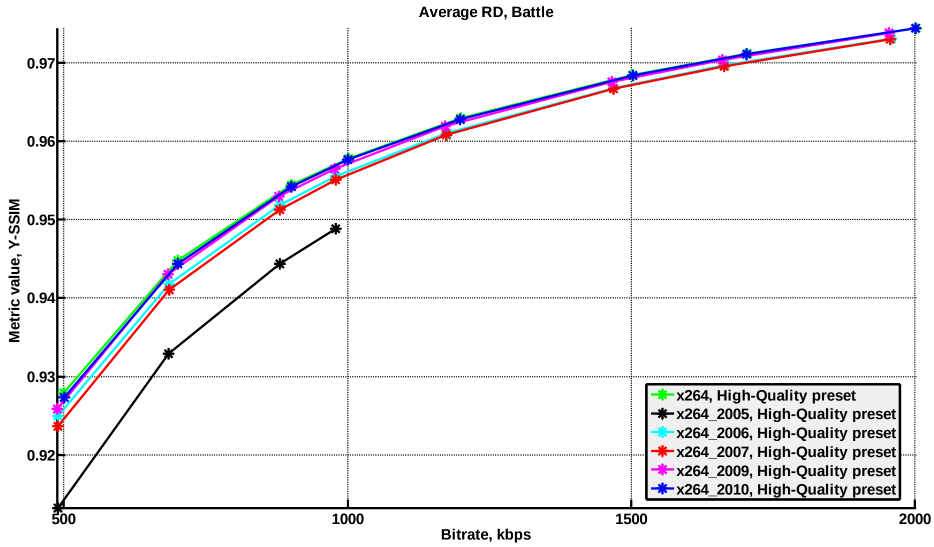


Figure 211. Bitrate/quality for different x264 encoder versions—usage area "Movies," "Battle" sequence, High Quality preset, Y-SSIM metric

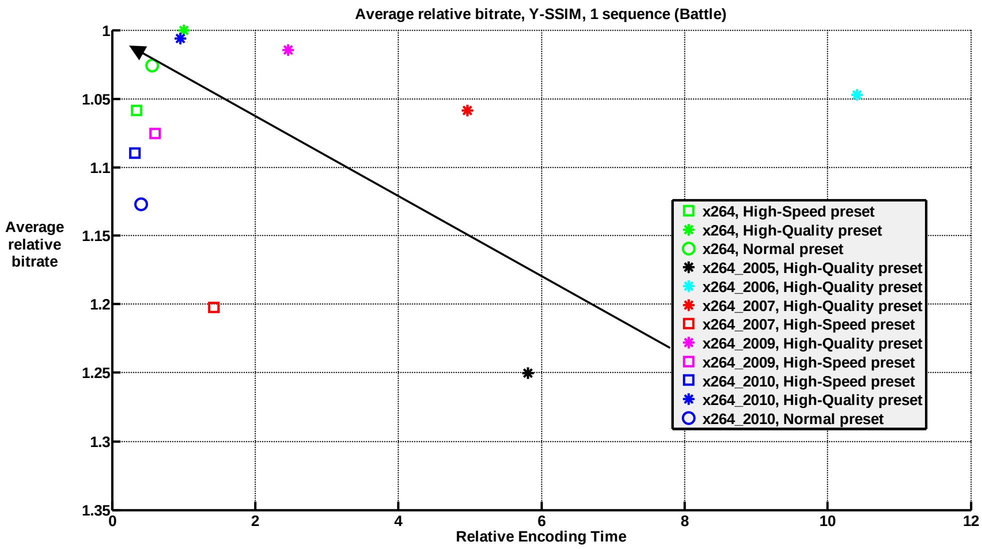


Figure 212. Progress of the x264 encoder over several years—Y-SSIM metric

6.2 WebM Comparison Over Time

WebM encoder participates comparison only for two years so next graphs show progress for WebM encoder for 2010 and 2011 years.

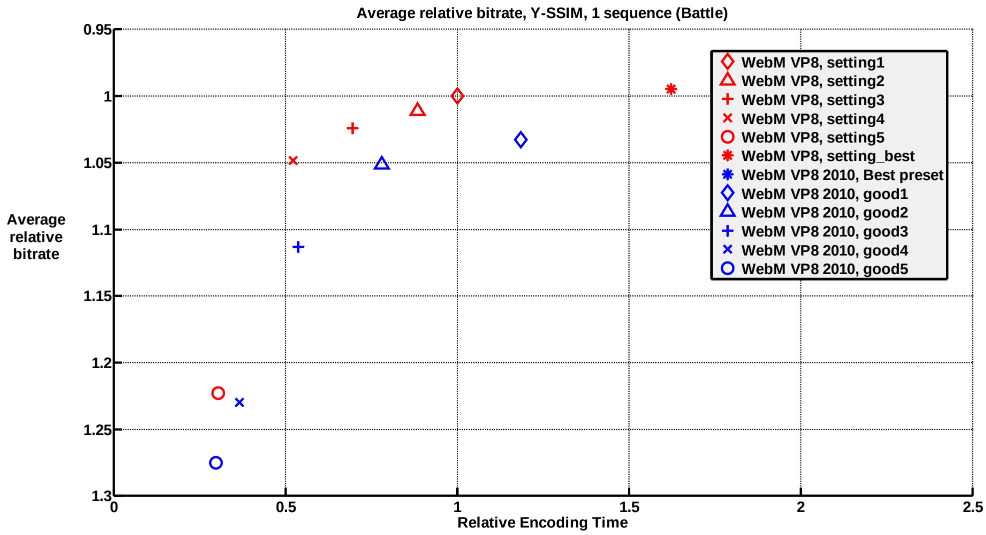


Figure 213. Speed/quality trade-off, “Battle” sequence, Y-SSIM metric

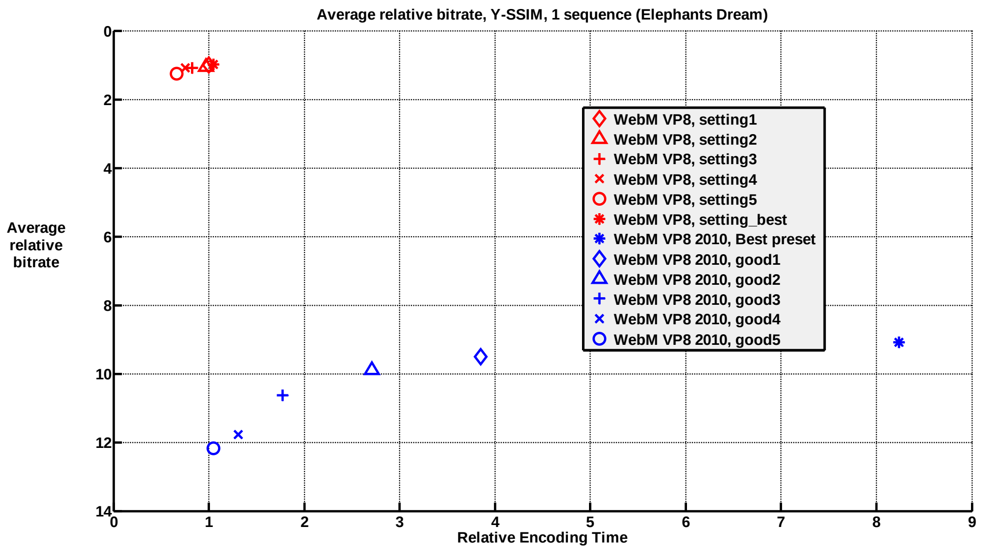


Figure 214. Speed/quality trade-off, “Elephant’s Dream” sequence, Y-SSIM metric

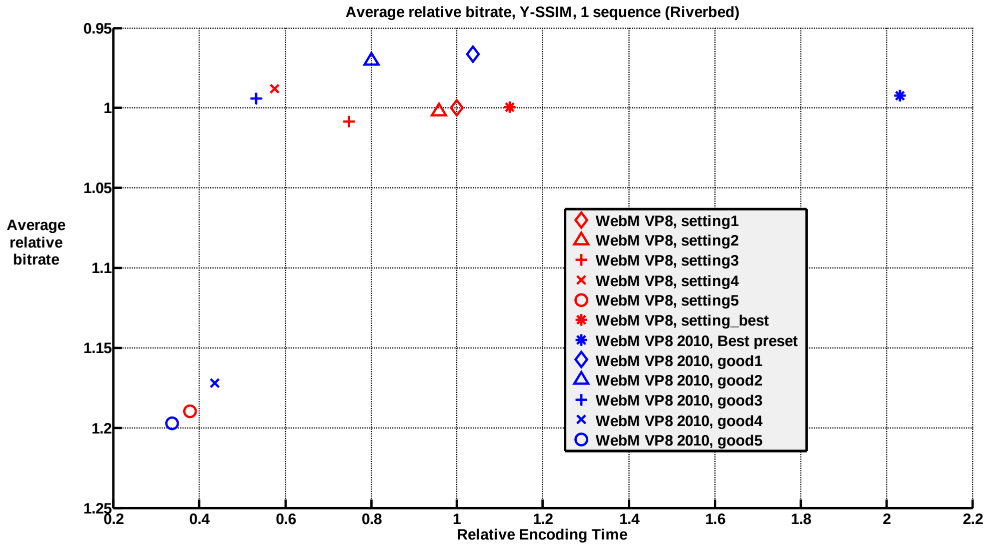


Figure 215. Speed/quality trade-off, "Riverbed" sequence, Y-SSIM metric

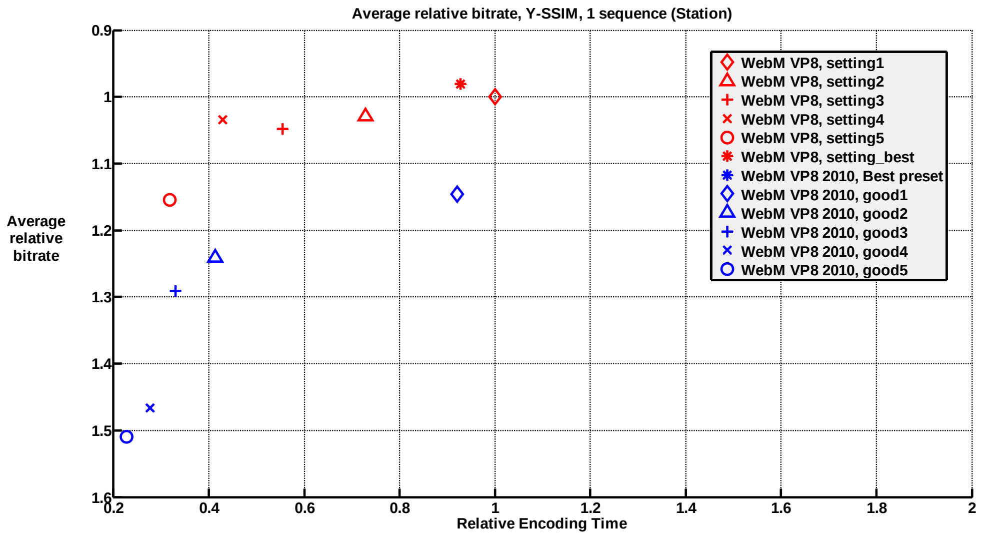


Figure 216. Speed/quality trade-off, "Station" sequence, Y-SSIM metric

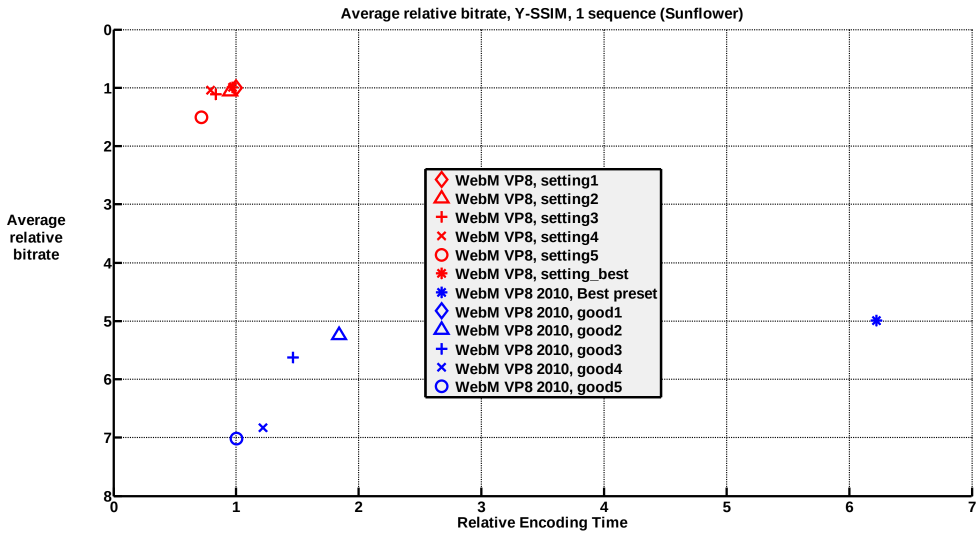


Figure 217. Speed/quality trade-off, "Sunflower" sequence, Y-SSIM metric

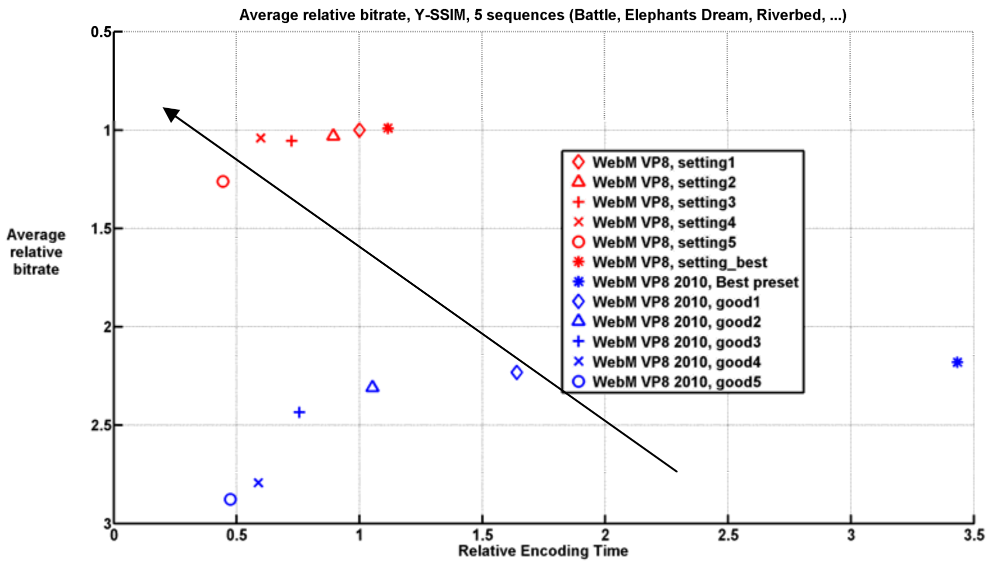


Figure 218. Speed/quality trade-off, 5 sequecnes, Y-SSIM metric

7 Appendix 3. Another Quality Metrics

To analyze quality results Y-SSIM metrics was used as main metric, but this appendix contains results for other quality metrics (MS-SSIM and 3-SSIM).

MS-SSIM and 3-SSIM descriptions could be found here:

http://compression.ru/video/quality_measure/info_en.html#msssim

http://compression.ru/video/quality_measure/info_en.html#3ssim

7.1 Movies

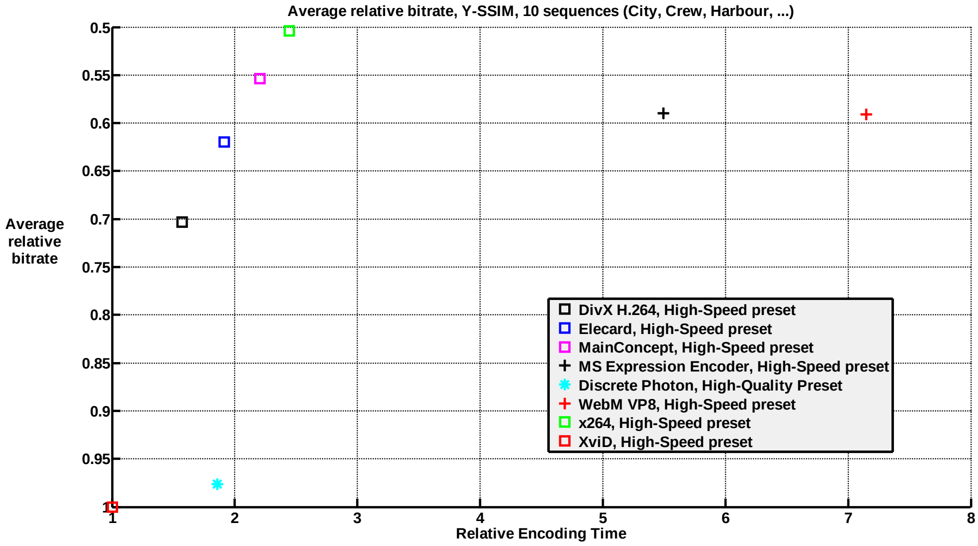


Figure 219. Speed/quality trade-off, Movies, High-Speed preset, all sequences, Y-SSIM metric

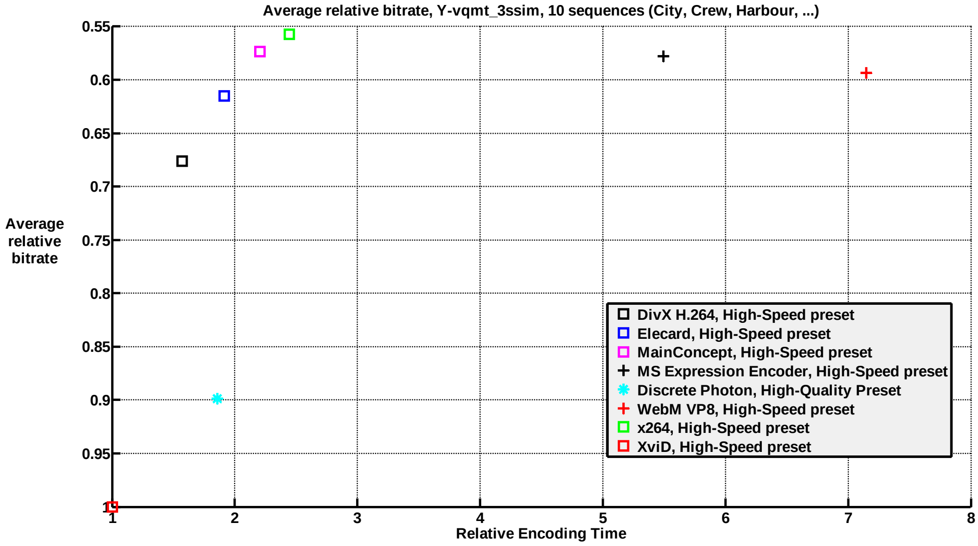


Figure 220. Speed/quality trade-off, Movies, High-Speed preset, all sequences, 3SSIM metric

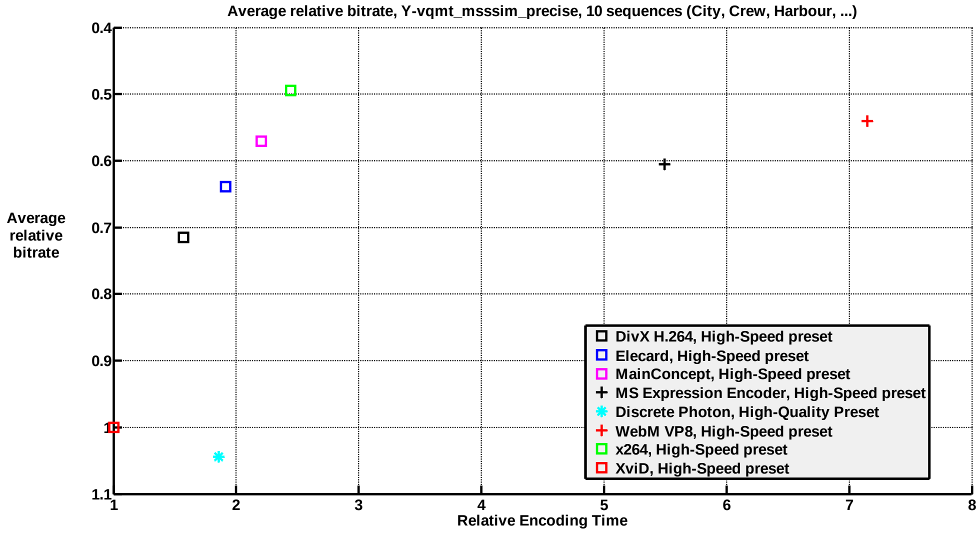


Figure 221. Speed/quality trade-off, Movies, High-Speed preset, all sequences, MS-SSIM metric

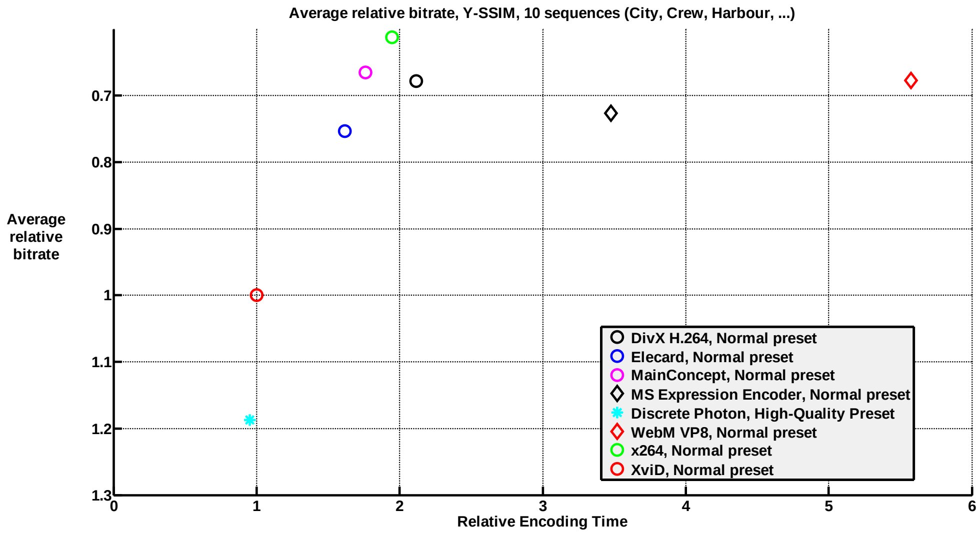


Figure 222. Speed/quality trade-off, Movies, Normal preset, all sequences, Y-SSIM metric

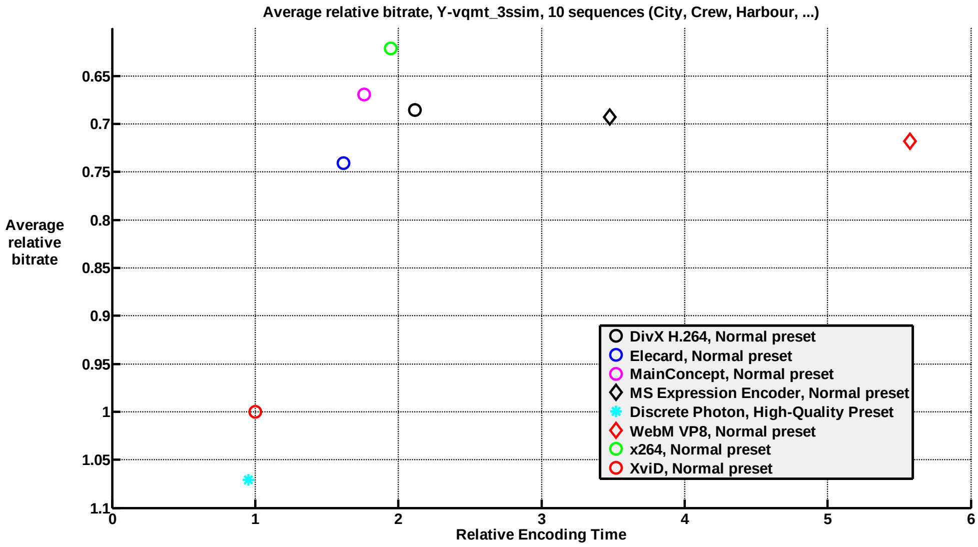


Figure 223. Speed/quality trade-off, Movies, Normal preset, all sequences, 3SSIM metric

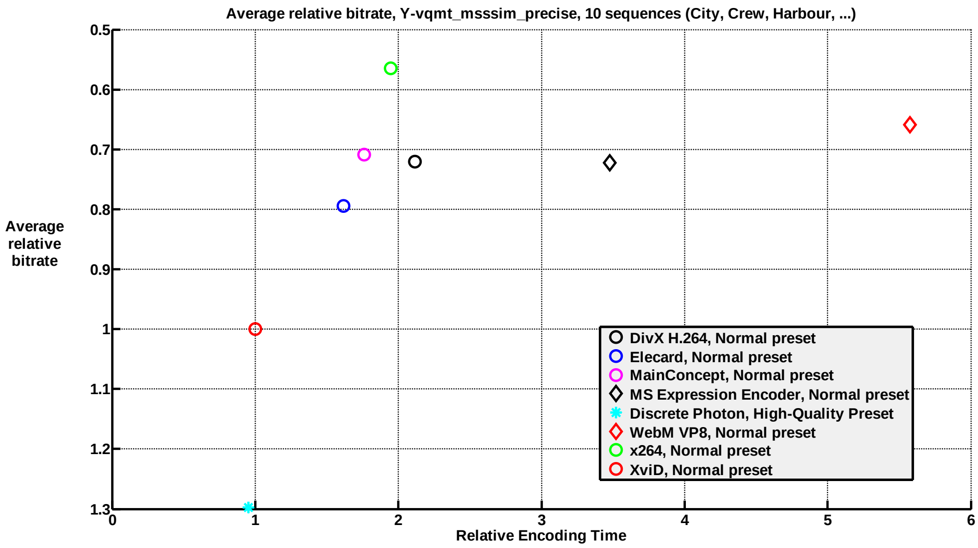


Figure 224. Speed/quality trade-off, Movies, Normal preset, all sequences, MS-SSIM metric

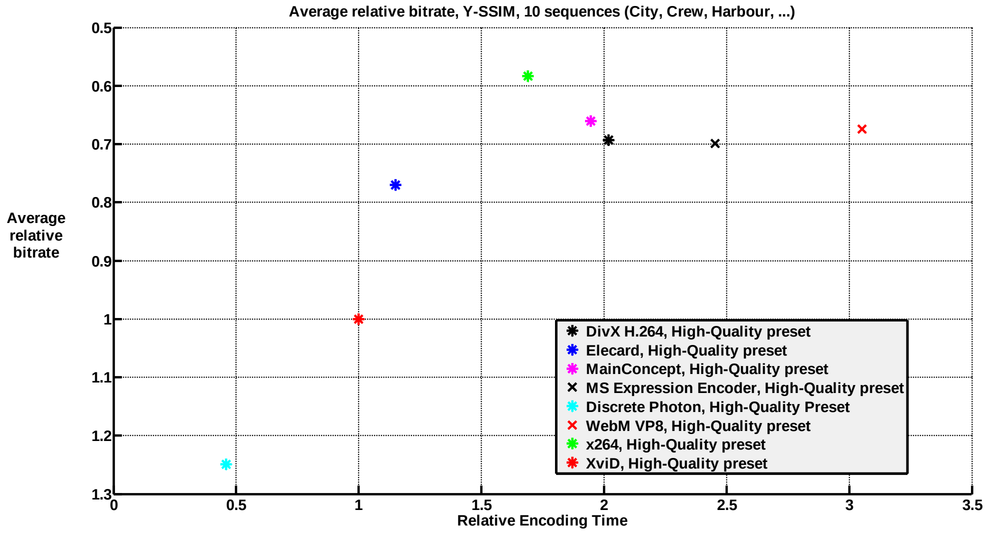


Figure 225. Speed/quality trade-off, Movies, High Quality preset, all sequences, Y-SSIM metric

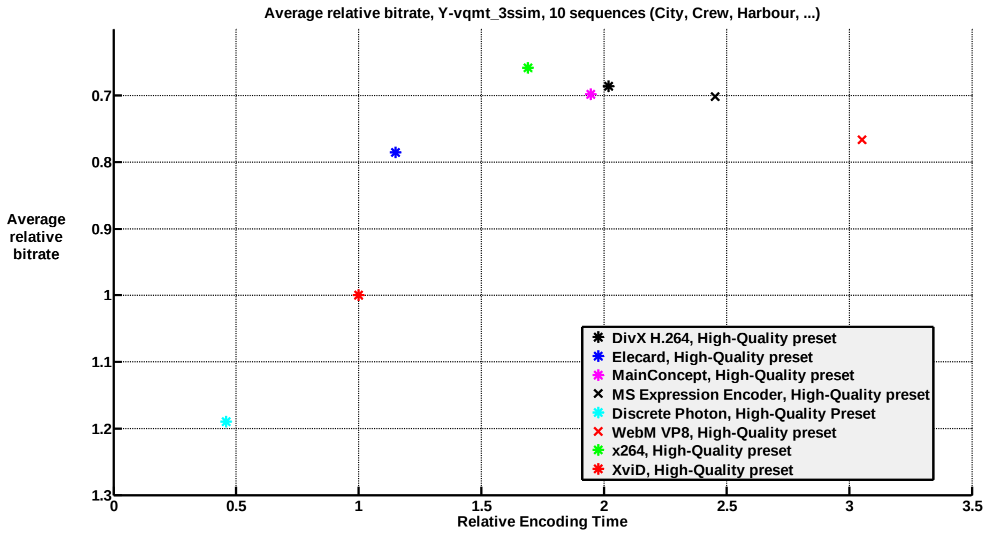


Figure 226. Speed/quality trade-off, Movies, High Quality preset, all sequences, 3SSIM metric

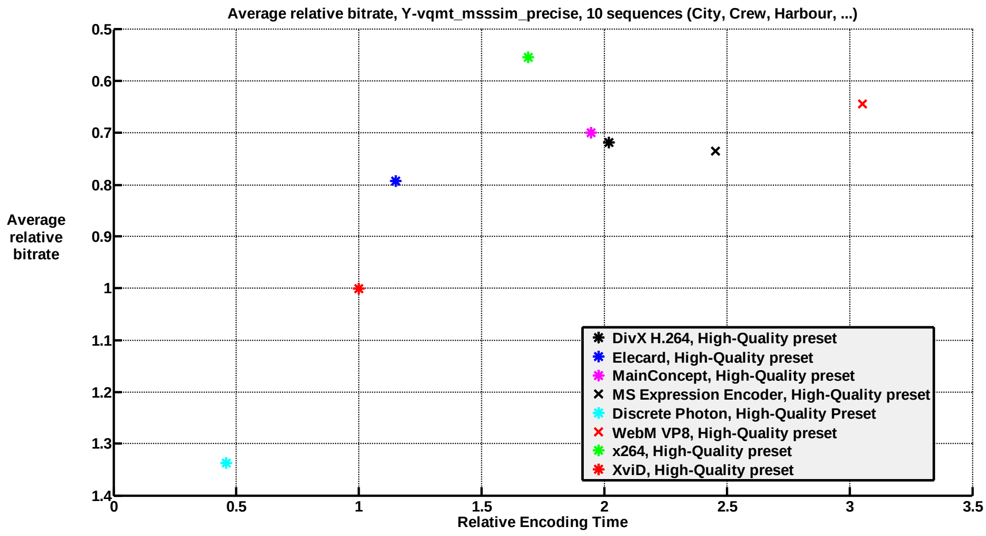


Figure 227. Speed/quality trade-off, Movies, High Quality preset, all sequences, MS-SSIM metric

7.2 HDTV

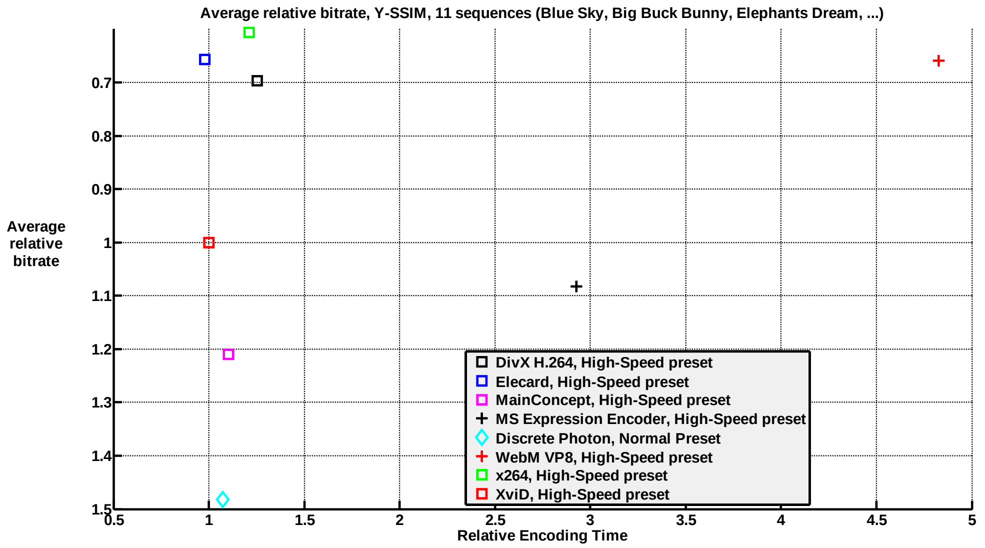


Figure 228. Speed/quality trade-off, HDTV, High-Speed preset, all sequences, Y-SSIM metric

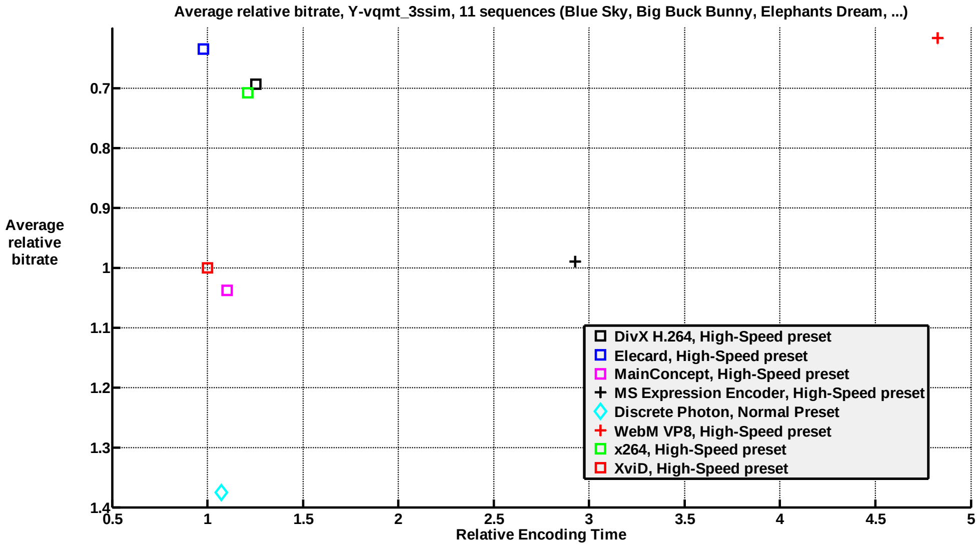


Figure 229. Speed/quality trade-off, HDTV, High-Speed preset, all sequences, 3SSIM metric

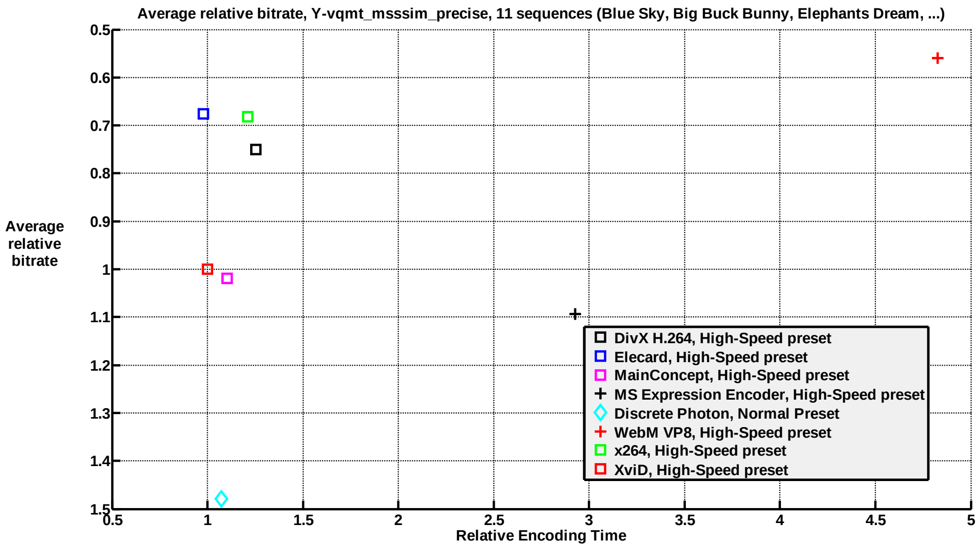


Figure 230. Speed/quality trade-off, HDTV, High-Speed preset, all sequences, MS-SSIM metric

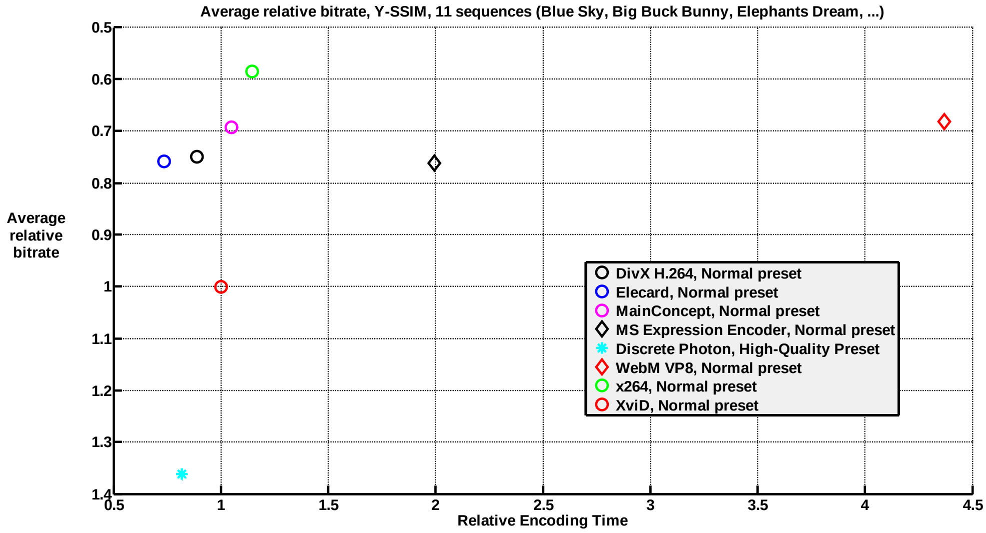


Figure 231. Speed/quality trade-off, HDTV, Normal preset, all sequences, Y-SSIM metric

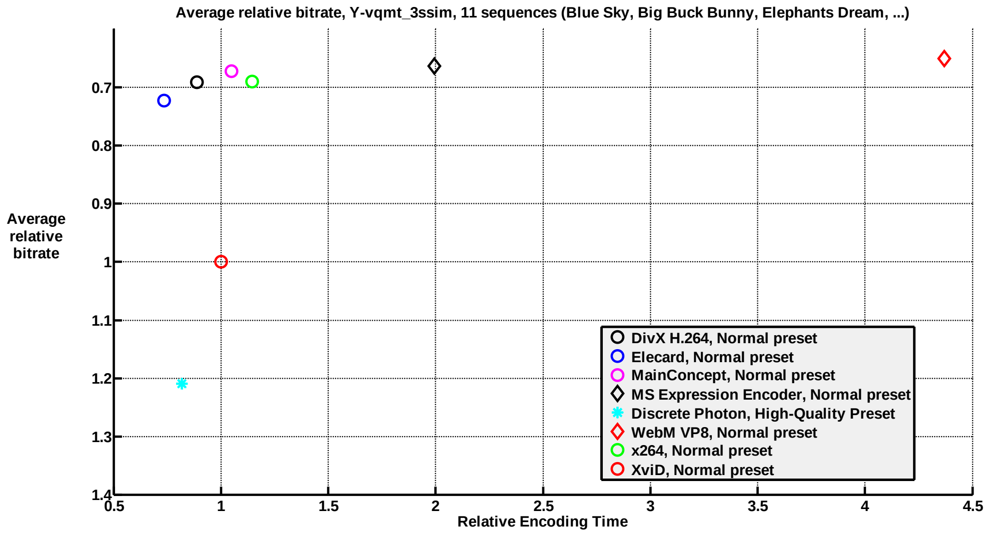


Figure 232. Speed/quality trade-off, HDTV, Normal preset, all sequences, 3SSIM metric

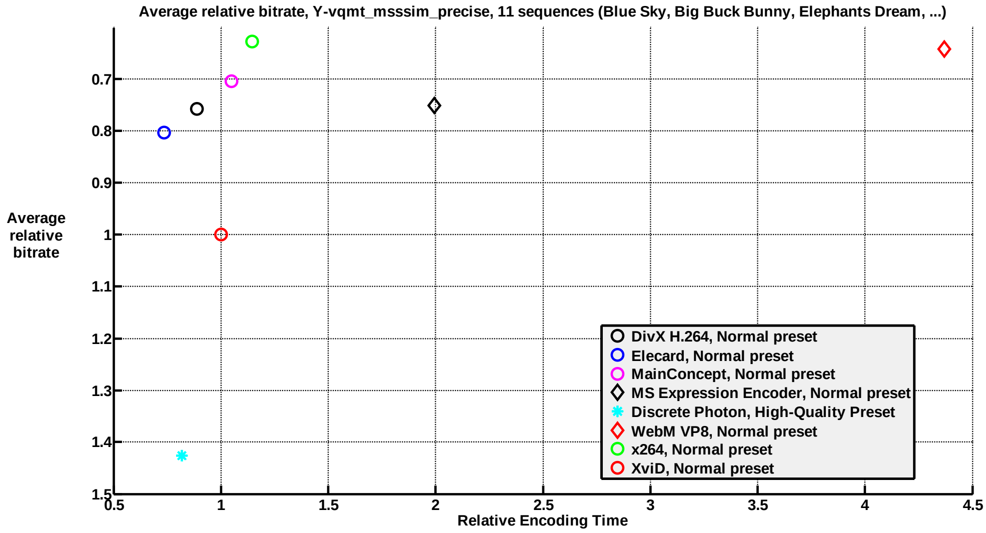


Figure 233. Speed/quality trade-off, HDTV, Normal preset, all sequences, MS-SSIM metric

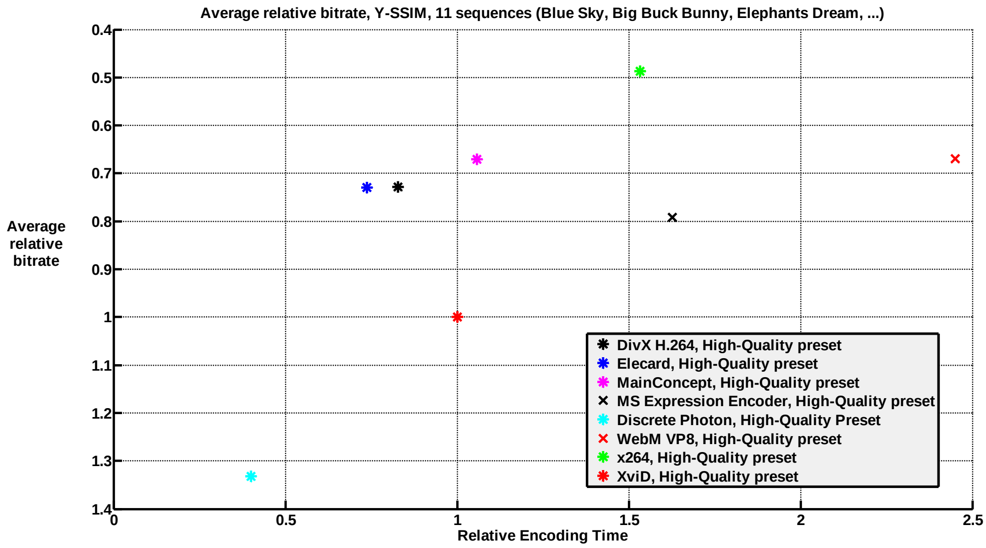


Figure 234. Speed/quality trade-off, HDTV, High Quality preset, all sequences, Y-SSIM metric

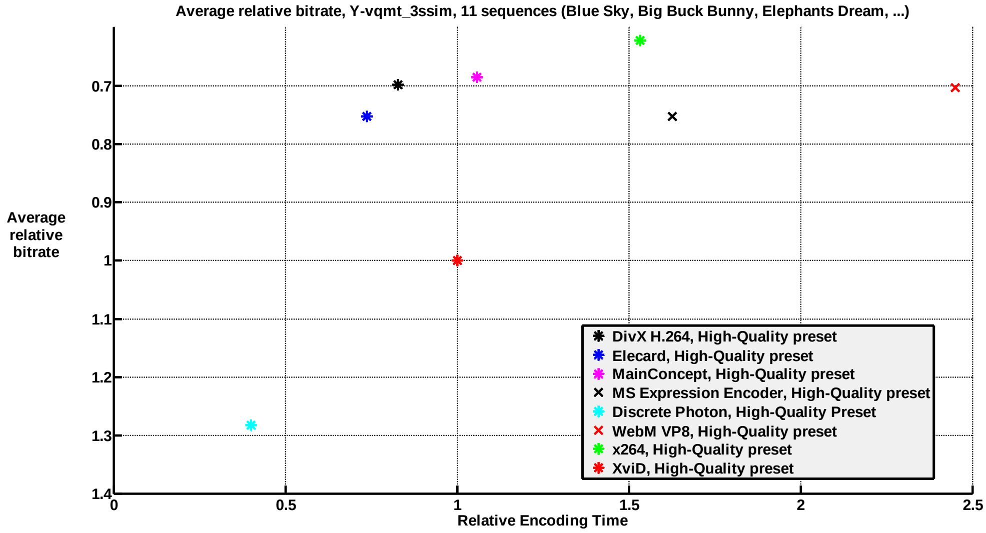


Figure 235. Speed/quality trade-off, HDTV, High Quality preset, all sequences, 3SSIM metric

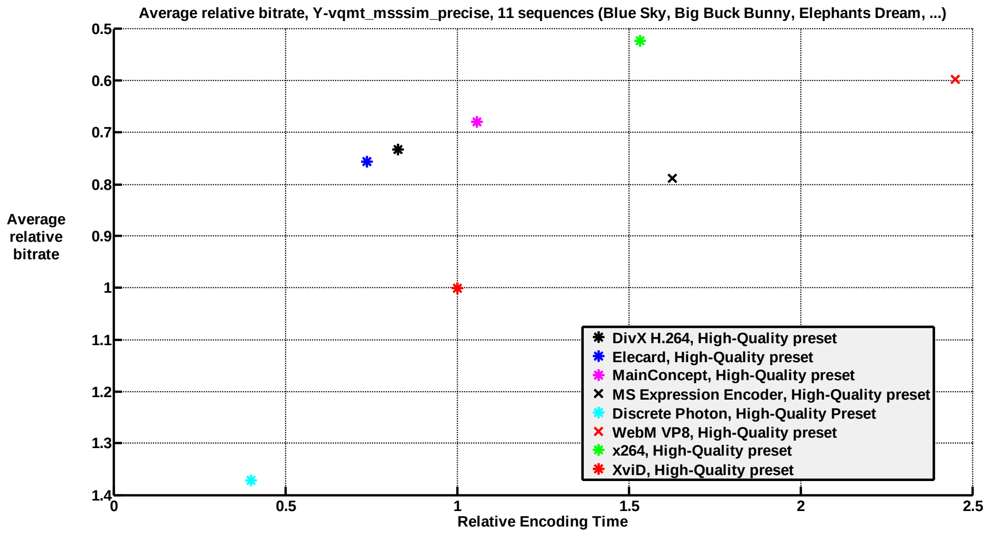


Figure 236. Speed/quality trade-off, HDTV, High Quality preset, all sequences, MS-SSIM metric

7.3 Conclusion on Different Metric Usage

Analyzing graphs with SSIM, 3SSIM and MS-SSIM one could make a conclusion – the chosen metric has no strong influence on results analysis, it changes positions of encoders slightly sometimes only.

8 Appendix 5. Test Set of Video Sequences

8.1 Videoconference Sequences

8.1.1 "CIF" ("Deadline")

Sequence title	Conference CIF
Resolution	352x288
Number of frames	1374
Color space	YV12
Frames per second	30
Source	Uncompressed, progressive

Figure 237. Deadline sequence, frame 1

Figure 238. Deadline sequence, frame 190

This is standard sequence. This sequence includes static background and foreground with very low motion – only announcer's face with not very rich mimic, except when he takes off his glasses. As a result, this sequence can e used to test the behavior of the codec for typical conference.

8.1.2 "4CIF"

Sequence title	Conference CIF
Resolution	640x480
Number of frames	3600
Color space	YV12
Frames per second	30
Source	HuffYUV, 57Mbps, progressive

Figure 239. 4CIF sequence, frame 743

Video with some movement and facial expressions in foreground and some very bright movement at background (man in red shirt at background).

8.1.3 “720p”

Sequence title	Conference 720p
Resolution	1280x720
Number of frames	1500
Color space	YV12
Frames per second	30
Source	HuffYUV, 160Mbps, progressive

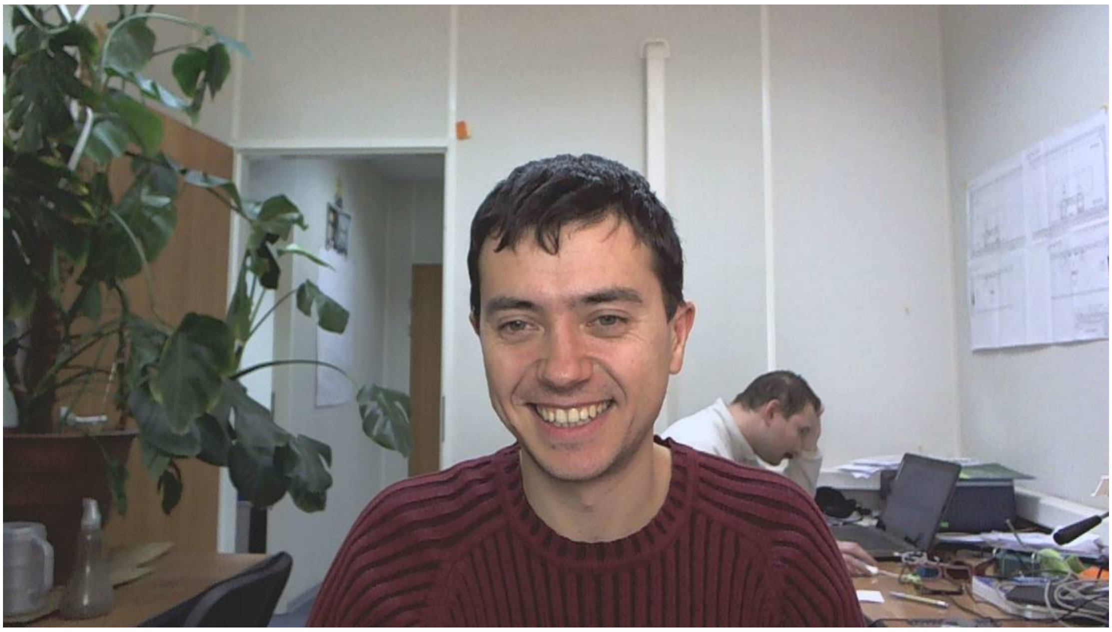


Figure 240. 720p sequence, frame 750

Same as in 4CIF version, typical videoconference sequence with talking head.

8.2 Movie Sequences

8.2.1 "City"

Sequence title	City
Resolution	704x576
Number of frames	600
Color space	YV12
Frames per second	60
Source	Uncompressed, progressive

Figure 241. City sequence, frame 400

This sequence is a panorama of New York city. A lot of small details such as building windows. Pretty similar colors all over the frames of the sequence. Camera shakes a little through the sequence.

8.2.2 "Indiana Jones"

Sequence title	Indiana Jones
Resolution	704x288
Number of frames	5000
Color space	YV12
Frames per second	30
Source	MPEG-2 (DVD), FlaskMPEG deinterlace

Figure 242. Indiana Jones sequence, frame 1

This sequence is a fragment from the *Indiana Jones* movie. Compression of this sequence is difficult for two main reasons: the presence of low-contrast scenes and the high level of motion in different scenes. Also, several scenes have very different types of motion, ranging from almost static scenes with talking people to scenes with strong motion (for example, the scene where stones fall).

8.2.3 "State Enemy"

Sequence title	State Enemy
Resolution	720x304
Number of frames	6500
Color space	YV12
Frames per second	24
Source	MPEG-2 (DVD), FlaskMPEG deinterlace

Figure 243. State Enemy sequence, frame 1115

This sequence is a fragment from the *Enemy of the State* movie. This sequence includes outdoor scenes with strong motion at the beginning when the bicyclist runs, as well as scenes with low motion and indoor scenes with normal motion. This sequence has scenes with different lighting conditions.

8.2.4 "Crew"

Sequence title	Crew
Resolution	704x576
Number of frames	600
Color space	YV12
Frames per second	60
Source	Uncompressed, progressive

Figure 244. Crew sequence, frame 301

This is a standard sequence of NASA crew. A lot of movement on the frames. Crew wears very bright suits and they are very distinct from grey background. A lot of camera flashes.

8.2.5 "Harbour"

Sequence title	Harbour
Resolution	704x576
Number of frames	600
Color space	YV12
Frames per second	60
Source	Uncompressed, progressive

Figure 245. Harbour sequence, frame 150

Standard sequence with harbor scene. A lot of vertical lines (boats masts) and other small details. Boats move a little, so there is pretty much movement of vertical lines. Also some water waving and sparkling included.

8.2.6 "Ice Skating"

Sequence title	Ice Skating
Resolution	704x576
Number of frames	600
Color space	YV12
Frames per second	60
Source	Uncompressed, progressive

Figure 246. Ice Skating sequence, frame 425

Standard sequence with public ice skating. People moves around on a pretty monotonous background. In the second half of the scene camera zooms out.

8.2.7 “Soccer”

Sequence title	Soccer
Resolution	704x576
Number of frames	600
Color space	YV12
Frames per second	60
Source	Uncompressed, progressive

Figure 247. Soccer sequence, frame 550

This sequence is a fragment from soccer team training in a sunny day. A lot of fast moving figures. Camera zooms out at the end of the sequence.

8.2.8 "Race Horses"

Sequence title	Race Horses
Resolution	832x480
Number of frames	300
Color space	YV12
Frames per second	30
Source	Uncompressed, progressive

Figure 248. Race Horses sequence, frame 196

This sequence contains horses walk in different directions. Camera moves around a bit too. Also sequence includes camera focus\defocus of the horses\grass. Some small details such as focused grass, horse hairs. A lot of horses overlapping.

8.2.9 “Party Scene”

Sequence title	Party Scene
Resolution	832x480
Number of frames	500
Color space	YV12
Frames per second	30
Source	Uncompressed, progressive

Figure 249. Party Scene sequence, frame 193

This sequence contains a party scene with camera zooming in. There are some transparent bubbles moving around through the sequence. There are some background movement such as kids on the left and dancing chicken. Some small details and contrast colors.

8.2.10 “Ice Age”

Sequence title	Ice Age
Resolution	720x480
Number of frames	2014
Color space	YV12
Frames per second	24
Source	MPEG-2 (DVD9), 5.7Mbps

Figure 250. Ice Age sequence, frame 500

This sequence is a fragment from the *Ice Age 3* animated movie. This movie has low-contrast portions and high-contrast portions, and it has many types of motion: camera panning, slow motion and very fast motion. Also, it has a scene with colors that differ completely from those of other scenes. Small black letterboxes appear at the top and bottom of the video.

8.3 HDTV Sequences

8.3.1 "Park Joy"

Sequence title	Park Joy
Resolution	1280x720
Number of frames	500
Color space	YV12
Frames per second	50
Source	Uncompressed, progressive

Figure 251. Park Joy sequence, frame 210

This standard sequence with strictly horizontal camera movement contains small figures of running people. Sometimes a large objects (trees) near the camera moves to the left, overlapping all the scene. At the end of the sequence camera slows the motion. Very bright colors on the top and some dark tones on the bottom.

8.3.2 “Riverbed”

Sequence title	Riverbed
Resolution	1920x1080
Number of frames	250
Color space	YV12
Frames per second	25
Source	Uncompressed, progressive

Figure 252. Riverbed sequence, frame 125

Riverbed seen through the water. Very hard to code. Static camera, no global moving, but there is no static parts in this sequence.

8.3.3 “Troy”

Sequence title	Troy
Resolution	1920x1072
Number of frames	300
Color space	YV12
Frames per second	24
Source	MPEG-2

Figure 253. Troy sequence, frame 1

This sequence is a fragment of the “Troy” movie and contains three parts with sharp scene changes. The video includes medium scene motion and slow camera motion. In terms of compression, this sequence is difficult to compress because of the many small details.

8.3.4 "Stockholm"

Sequence title	Stockholm
Resolution	1280x720
Number of frames	604
Color space	YV12
Frames per second	50
Source	Uncompressed, progressive

Figure 254. Stockholm sequence, frame 574

Panning view over the Old Town of Stockholm. Detailed houses, water and moving cars. Panning view over the Old Town of Stockholm. Detailed houses, water and moving cars. This sequence is interesting for compression because of high level of noise and sharp details in the scenes and moving camera and objects such as cars a and water

8.3.5 "Rush Hour"

Sequence title	Rush Hour
Resolution	1920x1080
Number of frames	250
Color space	YV12
Frames per second	25
Source	Uncompressed, progressive

Figure 255. Rush Hour sequence, frame 250

Rush-hour in Munich city. Many cars moving slowly, high depth of focus. Fixed camera.

8.3.6 "Blue Sky"

Sequence title	Blue Sky
Resolution	1920x1080
Number of frames	217
Color space	YV12
Frames per second	25
Source	Uncompressed, progressive

Figure 256. Blue Sky sequence, frame 100

Top of two trees against blue sky. High contrast, small color differences in the sky, many details. Camera rotation.

8.3.7 "Station"

Sequence title	Station
Resolution	1920x1080
Number of frames	313
Color space	YV12
Frames per second	25
Source	Uncompressed, progressive

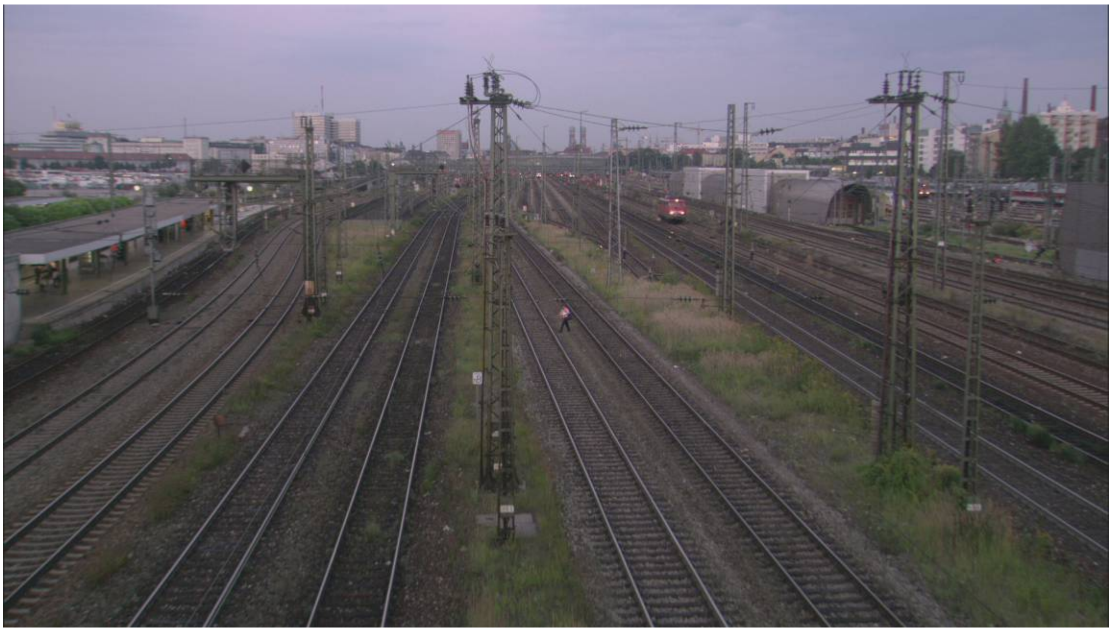


Figure 257. Station sequence, frame 155

View from a bridge to Munich station. Evening shot. Long zoom out. Many details, regular structures (tracks)

8.3.8 “Sunflower”

Sequence title	Sunflower
Resolution	1920x1080
Number of frames	500
Color space	YV12
Frames per second	25
Source	Uncompressed, progressive

Figure 258. Sunflower sequence, frame 370

Sunflower, very detailed shot. One bee at the sunflower, small color differences and very bright yellow. Fixed camera, small global motion.

8.3.9 “Tractor”

Sequence title	Tractor
Resolution	1920x1080
Number of frames	690
Color space	YV12
Frames per second	25
Source	Uncompressed, progressive

Figure 259. Tracktor sequence, frame 470

A tractor in a field. Whole sequence contains parts that are very zoomed in and a total view. Camera is following the tractor, chaotic object movement, structure of a harvested field.

8.3.10 “Big Buck Bunny”

Sequence title	Big Buck Bunny
Resolution	1920x1080
Number of frames	600
Color space	YV12
Frames per second	24
Source	Uncompressed, progressive

Figure 260. Big Buck Bunny sequence, frame 110

Scene from a cartoon movie Big Buck Bunny. Contains a lot of movement, very bright colors, different type of motion. The web-site for this movie is http://www.bigbuckbunny.org/

8.3.11 “Elephants Dream”

Sequence title	Elephants Dream
Resolution	1920x1080
Number of frames	600
Color space	YV12
Frames per second	24
Source	Uncompressed, progressive

Figure 261. Elephants Dream sequence, frame 460

Part of a cartoon movie Elephants Dream. Contains a lot of contrast thin lines and motion all over the scenes. Combination of dark colors with very bright small details makes this sequence pretty hard for encoding. The web-site for this movie is http://www.elephantsdream.org/

9 Appendix 6. Tested Codecs and Presets

9.1 Codecs

9.1.1 DivX AVC/H.264 Video Encoder

- Console encoding program version 1.1.1.9
- Presets were chosen by ourselves to meet the comparison requirements

***Remarks**: Owing to our choice of presets, the results for the DivX H.264 encoder could be slightly diminished compared with the case where the developers provide the presets.*

```
DivX AVC/H.264 Video Encoder (version 1.1.1.9) Copyright (c) 2010 DivX, Inc.
Usage: [options] -i <input file> -o <output file>

Available options are:

General:

                    -help   This help information
                       -h
              -noprogress   Do not display progress information
                 -version   Display version information
                 -v <0|1>   Verbose level

Input/Output:

          -i <input file>   AVI file or AVISynth script (avi|avs)
                              Pixel format in one of these formats:
                              YV12 IYUV YUY2 YVYU UYVY BGR24 BGR32
                            or raw yuv (use - for stdin) (requires -y)
         -o <output file>   Raw AVC bit stream
                            (Annex B raw byte stream format, type II)
      -y <width>x<height>   Input resolution (e.g. 1920x1080)
       -fps <int>[/<num>]   Override input frame rate (e.g. 30 or 30/1.001)
                     -tff   Interlaced input, top field first
                     -bff   Interlaced input, bottom field first
    -sar <width>:<height>   Sample Aspect Ratio [1:1]
              -start <int>  First frame to encode
             -frames <int>  Maximum number of frames to encode

Rate control:

                -br <int>   Target bitrate in kbps
                -qf <int>   Target Quality Factor
                              Lower number results in higher quality [0..51]

  Multipass:

             -npass <1|2>   Specify multipass mode
          -sf <stat file>   Specify multipass statistics file name [divx264_stat
.dat]

Encoder:

             -aqo <0|1|2>   Algorithm quality optimized for:
                             0 = Fast encoding
                             1 = Balanced performance/quality (default)
                             2 = Highest quality
                -I <1..4>   Gop length (seconds) [4]
             -fmode <1|2>   Interlace coding mode
                             1 = MBAFF
                             2 = Field
              -ref <1..4>   Maximum number of reference frames [4]
                 -pyramid   Enables pyramid encoding (implies -bref)
                    -bref   Enables B as reference
               -bf <0..3>   Maximum consecutive B-frames [2]
            -threads <int>  Maximum number of threads [auto]


The following frame rates for DivX Plus are permitted:

  60 Hz
  60000/1001 Hz
  50 Hz
  30 Hz
  30000/1001 Hz (i)
  25 Hz (i)
  24 Hz
  24000/1001 Hz

This pre-release version will expire on Wed Aug 29 01:00:00 2012
Please check http://labs.divx.com for new versions.
```

Figure 262. DivX AVC/H.264 video encoder

9.1.2 Elecard AVC Video Encoder 8-bit edition,

- Console encoding program version 2.1.026895.110204
- Codec and presets were provided by Elecard Ltd Company specifically for this test

```
Elecard AVC Video Encoder 8-bit edition, ver. 2.1.026895.110204
usage: avcenc.exe config.cfg [parameters list]
```

Figure 263. Elecard AVC Video Encoder 8-bit edition

9.1.3 MainConcept AVC/H.264 Video Encoder Console Application

- Console encoding version 1.5.0
- Codec and presets were provided by MainConcept AG Company specifically for this test

```
*****************************************************************
* A.G.E.N.T v1.5.0                                              *
*                                                               *
* Advanced General Encoding Tool:                               *
* Console application for testing any available codecs.       *
*****************************************************************
Usage: agent <config> [company]

         - <config>    the configuration file
         - [company]   the company prefix to search for (default: 'mc')
         - ? or help   display detailed help texts
                       There are different sections of interest:
                       'agent ?'
                       'agent ? input'
                       'agent ? fourcc'
                       'agent ? codec'
                       'agent ? profile'
                       'agent ? special'
```

Figure 264. MainConcept H.264/AVC encoder

9.1.4 Microsoft Expression Encoder 4

- CLI version of Microsoft Expression Encoder 4 was provided by Microsoft specifically for this test
- Codec and presets were provided by Microsoft specifically for this test

> ***Remarks:*** Owing to a long initial loading time, the encoding time for Microsoft Expression Encoder is significantly higher than for other encoders.

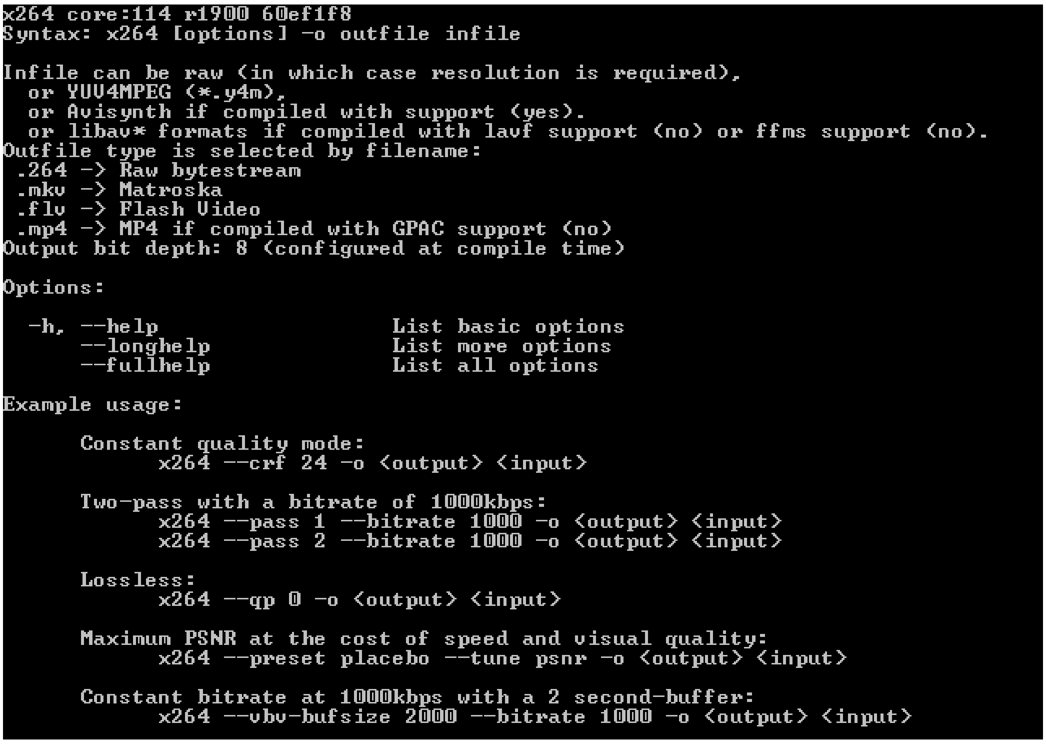
```
Microsoft Expression Encoder Command-Line Utility v1.0.1.0
Copyright (c) 2011 Microsoft Corporation
Encoder license installed: Pro

EECmd.exe -input <MediaFile> -preset [PresetXML] -output [MediaFile] -cuda [on|O
FF]
```

Figure 265. Microsoft Expression encoder

9.1.5 x264

- Console encoding program version core:114 r1900 60ef1f8 was provided by developers specifically for this test
- Codec and presets were provided by developers specifically for this test

Remarks: The presets provided by the developers for this comparison were specifically chosen for the SSIM metric.

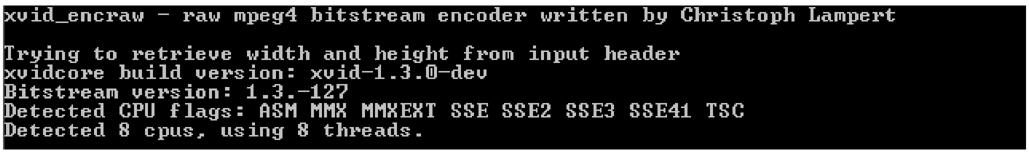


```
x264 core:114 r1900 60ef1f8
Syntax: x264 [options] -o outfile infile

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 (no) or ffms support (no).
Outfile type is selected by filename:
 .264 -> Raw bytestream
 .mkv -> Matroska
 .flv -> Flash Video
 .mp4 -> MP4 if compiled with GPAC support (no)
Output bit depth: 8 (configured at compile time)

Options:

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

Example usage:

      Constant quality mode:
            x264 --crf 24 -o <output> <input>

      Two-pass with a bitrate of 1000kbps:
            x264 --pass 1 --bitrate 1000 -o <output> <input>
            x264 --pass 2 --bitrate 1000 -o <output> <input>

      Lossless:
            x264 --qp 0 -o <output> <input>

      Maximum PSNR at the cost of speed and visual quality:
            x264 --preset placebo --tune psnr -o <output> <input>

      Constant bitrate at 1000kbps with a 2 second-buffer:
            x264 --vbv-bufsize 2000 --bitrate 1000 -o <output> <input>
```

Figure 266. x264 encoder

9.1.6 XviD raw mpeg4 bitstream encoder

- Console encoding program
- Codec and presets used was taken from previous comparison



```
xvid_encraw - raw mpeg4 bitstream encoder written by Christoph Lampert

Trying to retrieve width and height from input header
xvidcore build version: xvid-1.3.0-dev
Bitstream version: 1.3.-127
Detected CPU flags: ASM MMX MMXEXT SSE SSE2 SSE3 SSE41 TSC
Detected 8 cpus, using 8 threads.
```

Figure 267. XviD encoder

9.1.7 Discrete Photon

- Console encoding program version core:114 r1900 60ef1f8 was provided by developers specifically for this test
- Codec and presets were provided by developers specifically for this test

```
Usage: dpcl <input-file(.yuv or .yv12)> <output-file(.264)> <options>

Options:
-size:<width>x<height>          frame size
-fps:<frame-rate>               Input frame-rate (in frames per second)
-bps:<bit-rate>          Output bit-rate (in bits per second)
-speed:<0,1,2>           Encoding speed (0:slow(default), 1:fast, 2:faster)
-dbv:<0,1,2>             Decoder buffer verifier (0:hard, 1:soft(default), 2:none
)
-filler:<0,1>            Insert filler data to prevent decoder buffer overflow (0
:no(default), 1:yes)
```

Figure 268. Discrete Photon encoder

9.1.8 WebM vp8 Codec

- Encoder, decoder and presets was provided by WebM specifically for this test
- Encoder version: v0.9.2-522-gddd260e

> ***Remarks:*** The presets provided by the developers for this comparison were specifically chosen for the SSIM metric.

```
Usage: vpxdec.exe <options> filename

Options:
            --codec=<arg>                Codec to use
            --yv12                       Output raw YV12 frames
            --i420                       Output raw I420 frames
            --flipuv                     Flip the chroma planes in the output
            --noblit                     Don't process the decoded frames
            --progress                   Show progress after each frame decodes
            --limit=<arg>                Stop decoding after n frames
            --postproc                   Postprocess decoded frames
            --summary                    Show timing summary
  -o <arg>, --output=<arg>               Output file name pattern (see below)
  -t <arg>, --threads=<arg>              Max threads to use
  -v,       --verbose                    Show version string
            --md5                        Compute the MD5 sum of the decoded frame

VP8 Postprocessing Options:
            --noise-level=<arg>          Enable VP8 postproc add noise
            --deblock                    Enable VP8 deblocking
            --demacroblock-level=<arg>   Enable VP8 demacroblocking, w/ level
            --pp-debug-info=<arg>        Enable VP8 visible debug info
            --pp-dbg-ref-frame=<arg>     Display only selected reference frame pe
r macro block
            --pp-dbg-mb-modes=<arg>      Display only selected macro block modes
            --pp-dbg-b-modes=<arg>       Display only selected block modes
            --pp-dbg-mvs=<arg>           Draw only selected motion vectors

Output File Patterns:

  The -o argument specifies the name of the file(s) to write to. If the
  argument does not include any escape characters, the output will be
  written to a single file. Otherwise, the filename will be calculated by
  expanding the following escape characters:

        %w   - Frame width
        %h   - Frame height
        %<n> - Frame number, zero padded to <n> places (1..9)

  Pattern arguments are only supported in conjunction with the --yv12 and
  --i420 options. If the -o option is not specified, the output will be
  directed to stdout.

Included decoders:

    vp8    - WebM Project VP8 Decoder v0.9.2-522-gddd260e
```

Figure 269. WebM vp8 encoder

9.2 Presets

The table below lists the settings used in this comparison for all of the codecs.

<table>
<tr><th>Codec</th><th>Preset Name</th><th>Preset</th></tr>
<tr><td rowspan="6">DivX H.264</td><td>Movie
“High Speed”</td><td>-aqo 0 -ref 1 -bf 0</td></tr>
<tr><td>Movie “Normal”</td><td>Default presets</td></tr>
<tr><td>Movie
“High Quality”</td><td>1-st pass: -npass 1
2-nd pass: -npass 2</td></tr>
<tr><td>HDTV
“High Speed”</td><td>-aqo 0 -ref 1 -bf 0</td></tr>
<tr><td>HDTV “Normal”</td><td>-aqo 0</td></tr>
<tr><td>HDTV
“High Quality”</td><td>-bf 3 -pyramid -bref</td></tr>
<tr><td rowspan="3">Elecard</td><td>Movie
“Normal”</td><td>
<table>
<tr><th>Parameter name</th><th>Value</th><th>Comment</th></tr>
<tr><td>BMax</td><td>3</td><td>max number of b-frames</td></tr>
<tr><td>BMode</td><td>2</td><td>2 - hierarchical structure</td></tr>
<tr><td>ModeDecision</td><td>1</td><td>1 - SATD</td></tr>
<tr><td>WPredMode</td><td>1</td><td>1 - explicit mode (for both P- and B-frames)</td></tr>
<tr><td>NumRefFrames</td><td>3</td><td>actual size of DPB</td></tr>
<tr><td>AQMode</td><td>0</td><td>0 - do not use</td></tr>
<tr><td>Lookahead</td><td>1</td><td>lookahead length in seconds</td></tr>
<tr><td>OffsetCb</td><td>1</td><td>[-10,+10] i prefer 0 or -1</td></tr>
<tr><td>OffsetCr</td><td>1</td><td>[-10,+10] i prefer 0 or -1.</td></tr>
<tr><td>AQMode</td><td>0</td><td>0 - do not use</td></tr>
<tr><td>DeblockAlpha</td><td>-1</td><td>[-6,+6] really depends on source</td></tr>
<tr><td>DeblockBeta</td><td>-1</td><td>[-6,+6] really depends on source.</td></tr>
<tr><td>DeblockMode</td><td>0</td><td>0 - filter whole picture</td></tr>
<tr><td>IntraNewInI
Intra8x8InI
Intra4x4InI
IntraNewInP
Intra8x8InP
Intra4x4InP
IntraNewInB
Intra8x8InB
Intra4x4InB</td><td>1
1
1
1
1
1
1
1
1</td><td>enables MB intra in I-slices

same for P-slices

same for B-slices</td></tr>
<tr><td>BlockMode</td><td>1</td><td>1 - MC down to 8x8</td></tr>
<tr><td>MaxVectorLen</td><td>511</td><td>for both horz and vert components</td></tr>
</table>
</td></tr>
<tr><td>Movie
“High Speed”</td><td>
<table>
<tr><th>Parameter name</th><th>Value</th><th>Comment</th></tr>
<tr><td>BMax</td><td>1</td><td>max number of b-frames</td></tr>
<tr><td>BMode</td><td>0</td><td>0 - plain vanilla</td></tr>
<tr><td>ModeDecision</td><td>0</td><td>0 - SAD</td></tr>
<tr><td>WPredMode</td><td>0</td><td>0 - not used</td></tr>
<tr><td>NumRefFrames</td><td>2</td><td>actual size of DPB</td></tr>
</table>
</td></tr>
<tr><td>Movie “High Quality”</td><td>1-st pass:
<table>
<tr><th>Parameter name</th><th>Value</th><th>Comment</th></tr>
<tr><td>BMax</td><td>3</td><td>max number of b-frames</td></tr>
<tr><td>BMode</td><td>2</td><td>2 - hierarchical structure</td></tr>
<tr><td>ModeDecision</td><td>1</td><td>1 - SATD</td></tr>
<tr><td>WPredMode</td><td>1</td><td>1 - explicit mode (for both P- and B-frames)</td></tr>
<tr><td>NumRefFrames</td><td>3</td><td>actual size of DPB</td></tr>
</table>
2-nd pass:
<table>
<tr><th>Parameter name</th><th>Value</th><th>Comment</th></tr>
<tr><td>NumRefFrames</td><td>4</td><td>actual size of DPB</td></tr>
</table>
</td></tr>
</table>

HDTV “Normal”

Parameter name	Value	Comment
Intra8x8InP Intra4x4InP IntraNewInB Intra8x8InB Intra4x4InB	0 0 0 0 0	disables MB intra in corresponding slices

HDTV “High Speed”

Parameter name	Value	Comment
BMax	1	max number of b-frames
BMode	0	0 - plain vanilla
Intra4x4InI	0	disables 4x4 in I-slices
ModeDecision	0	0 - SAD
WPredMode	0	0 - not used
NumRefFrames	1	actual size of DPB
MaxVectorLen	255	for both horz and vert components

HDTV “High Quality”

1-st pass:

Parameter name	Value	Comment
Intra8x8InP Intra4x4InP IntraNewInB Intra8x8InB Intra4x4InB	1 1 1 1 1	enables MB intra in corresponding slices

2-nd pass:

Parameter name	Value	Comment
Intra8x8InP Intra4x4InP IntraNewInB Intra8x8InB Intra4x4InB	1 1 1 1 1	enables MB intra in corresponding slices
NumRefFrames	4	actual size of DPB

VideoConference

Same as “Movie Normal”

Microsoft Expression Encoder 4

Movie “Normal”

Parameter name	Value
BFrameCount	3
EntropyMode	“Cabac”
RDOptimizationMode	“Speed”
HadamardTransform	False
SubBlockMotionSearchMode	“Balanced”
MultiReferenceMotionSearchMode	“Speed”
ReferenceBFrames	True
AdaptiveBFrames	True
SceneChangeDetector	True
FastIntraDecisions	False
FastInterDecisions	False
SubPixelMode	“Quarter”
SliceCount	0
KeyFrameDistance	“00:00:10”
InLoopFilter	True
MEPartitionLevel	“EightByEight”
ReferenceFrames	5
SearchRange	64
AutoFit	False
Force16Pixels	False
SmoothStreaming	False

Movie “High Speed”

Parameter name	Value
MultiReferenceMotionSearchMode	“Balanced”
FastIntraDecisions	True
FastInterDecisions	True

Movie “High Quality”

Parameter name	Value
RDOptimizationMode	“Quality”
HadamardTransform	True
SubBlockMotionSearchMode	“Quality”

HDTV “Normal”

Parameter name	Value
SearchRange	32
FastIntraDecisions	True
FastInterDecisions	True

Codec	Preset	Parameter name	Value
	HDTV “High Speed”	EntropyMode	“Cavlc”
		MultiReferenceMotionSearchMode	Balanced
		SearchRange	16
	HDTV “High Quality”	RDOptimizationMode	“Quality”
		HadamardTransform	True
		SubBlockMotionSearchMode	“Quality”
		FastIntraDecisions	False
		FastInterDecisions	False
	VideoConference	SearchRange	32
		FastIntraDecisions	True
		FastInterDecisions	True
		MultiReferenceMotionSearchMode	Balanced
MainConcept	Movie “Normal”	num_reference_frames	4
		search_range	192
		rd_optimization	1
		bit_rate_mode	2
		video_type	32
		inter_search_shape	1
		entropy_coding_mode	1
		use_hadamard_transform	1
		hrd_maintain	1
		use_deblocking_filter	1
		me_subpel_mode	1
		me_weighted_p_mode	1
		me_weighted_b_mode	0
		enable_fast_intra_decisions	1
		enable_fast_inter_decisions	1
		cpu_opt	0
		fast_multi_ref_me	0
		fast_sub_block_me	0
		constrained_ref_list	1
		fast_rd_optimization	1
		enable_intra_big	1
		enable_intra_8x8	1
		enable_intra_4x4	1
		enable_intra_pcm	0
		enable_inter_big	1
		enable_inter_8x8	1
		enable_inter_4x4	1
		enable_inter_pcm	0
	Movie “High Speed”	search_range	144
		use_hadamard_transform	0
		fast_multi_ref_me	1
		fast_sub_block_me	1
		constrained_ref_list	0
	Movie, HDTV “High Quality”	search_range	208
		enable_fast_intra_decisions	0
		enable_fast_inter_decisions	0
		fast_rd_optimization	0
		constrained_ref_list	0
	HDTV “Normal”	num_reference_frames	3
		search_range	320
		use_hadamard_transform	0
		hrd_maintain	0
		video_type	11
		fast_sub_block_me	1
		constrained_ref_list	0
		fast_rd_optimization	0
	HDTV “High Speed”	num_reference_frames	3
		search_range	192
		inter_search_shape	0
		entropy_coding_mode	0

<table>
<tr><td></td><td></td><td><table>
<tr><td>use_deblocking_filter</td><td>0</td></tr>
<tr><td>me_subpel_mode</td><td>0</td></tr>
<tr><td>me_weighted_p_mode</td><td>0</td></tr>
<tr><td>fast_multi_ref_me</td><td>1</td></tr>
<tr><td>enable_intra_4x4</td><td>0</td></tr>
<tr><td>enable_inter_4x4</td><td>0</td></tr>
<tr><td>fast_rd_optimization</td><td>1</td></tr>
</table></td></tr>
<tr><td></td><td>HDTV
“High Quality”</td><td><table>
<tr><th>Parameter name</th><th>Value</th></tr>
<tr><td>num_reference_frames</td><td>4</td></tr>
<tr><td>search_range</td><td>416</td></tr>
<tr><td>enable_fast_inter_decisions</td><td>0</td></tr>
<tr><td>fast_sub_block_me</td><td>0</td></tr>
</table></td></tr>
<tr><td></td><td>VideoConference</td><td><table>
<tr><th>Parameter name</th><th>Value</th></tr>
<tr><td>bit_rate_mode</td><td>0</td></tr>
<tr><td>drop_frame_timecode</td><td>1</td></tr>
<tr><td>fast_rd_optimization</td><td>0</td></tr>
<tr><td>adaptive_quant_strength</td><td>{0,0,50,0,0,0,0,0}</td></tr>
</table></td></tr>
<tr><td>x264</td><td>Movie
“Normal”</td><td>1-st pass:
--tune ssim --pass 1 --keyint 500 --preset medium
2-nd pass:
--tune ssim --pass 2 --keyint 500 --preset medium</td></tr>
<tr><td></td><td>Movie
“High Speed”</td><td>--tune ssim --keyint 500 --preset fast</td></tr>
<tr><td></td><td>Movie
“High Quality”</td><td>1-st pass:
--tune ssim --pass 1 --keyint 500 --preset slow
2-nd pass:
--tune ssim --pass 2 --keyint 500 --preset slow</td></tr>
<tr><td></td><td>HDTV
“Normal”</td><td>1-st pass:
--tune ssim --pass 1 --keyint 500 --preset faster
2-nd pass:
--tune ssim --pass 2 --keyint 500 --preset faster</td></tr>
<tr><td></td><td>HDTV
“High Speed”</td><td>--tune ssim --keyint 500 --preset veryfast</td></tr>
<tr><td></td><td>HDTV
“High Quality”</td><td>1-st pass:
--tune ssim --pass 1 --keyint 500 --preset slow

2-nd pass:
--tune ssim --pass 2 --keyint 500 --preset slow</td></tr>
<tr><td></td><td>VideoConference</td><td>--tune ssim --keyint 500 --preset slower</td></tr>
<tr><td>XviD</td><td>Movie
“High Speed”</td><td>-type 0 -quality 5 -vhqmode 1 -max_bframes 0 -reaction 8 -averaging 50 -smoother 50</td></tr>
<tr><td></td><td>Movie
“Normal”</td><td>1-st pass:
-type 0 -pass1 -quality 6 -vhqmode 1 -ostrength 20 -oimprove 10 -odegrade 10

2-nd pass:
-type 0 -pass2 -quality 6 -vhqmode 1 -ostrength 20 -oimprove 10 -odegrade 10</td></tr>
<tr><td></td><td>Movie
“High Quality”</td><td>1-st pass:
-type 0 -pass1 -quality 6 -vhqmode 4 -bvhq -qpel -ostrength 20 -oimprove 10 -odegrade</td></tr>
</table>

		10 2-nd pass: -type 0 -pass2 -quality 6 -vhqmode 4 -bvhq -qpel -ostrength 20 -oimprove 10 -odegrade 10
	HDTV “High Speed”	-type 0 -quality 5 -vhqmode 1 -max_bframes 0 -reaction 8 -averaging 50 -smoother 50
	HDTV “Normal”	1-st pass: -type 0 -pass1 -quality 6 -vhqmode 1 -ostrength 20 -oimprove 10 -odegrade 10 2-nd pass: -type 0 -pass2 -quality 6 -vhqmode 1 -ostrength 20 -oimprove 10 -odegrade 10
	HDTV “High Quality”	1-st pass: -type 0 -pass1 -quality 6 -vhqmode 4 -bvhq -qpel -ostrength 20 -oimprove 10 -odegrade 10 2-nd pass: -type 0 -pass2 -quality 6 -vhqmode 4 -bvhq -qpel -ostrength 20 -oimprove 10 -odegrade 10
	VideoConference	-type 0 -max_bframes 2 -quality 6 -vhqmode 4 -bvhq -qpel -gmc
Discrete Photon	Normal preset	-dbv:0 -speed:1
	Quality preset	-dbv:0 -speed:0
WebM vp8	Movie “High Speed”	--tune=ssim --good --cpu-used=4 --lag-in-frames=25 --min-q=0 --max-q=63 --end-usage=0 --codec=vp8 --auto-alt-ref=1 -p 1 --kf-max-dist=999999 --kf-min-dist=0 --drop-frame=0 --static-thresh=0 --i420 --bias-pct=50 --minsection-pct=0 --maxsection-pct=1000 --arnr-maxframes=7 --arnr-strength=6 --arnr-type=3
	Movie “Normal”	--tune=ssim --good --cpu-used=2 --lag-in-frames=25 --min-q=0 --max-q=63 --end-usage=0 --codec=vp8 --auto-alt-ref=1 -p 2 --kf-max-dist=999999 --kf-min-dist=0 --drop-frame=0 --static-thresh=1 --i420 --bias-pct=50 --minsection-pct=0 --maxsection-pct=1000 --arnr-maxframes=7 --arnr-strength=6 --arnr-type=3 --sharpness=0
	Movie “High Quality”	--tune=ssim --good --cpu-used=0 --lag-in-frames=25 --min-q=0 --max-q=63 --end-usage=0 --codec=vp8 --auto-alt-ref=1 -p 2 --kf-max-dist=999999 --kf-min-dist=0 --drop-frame=0 --static-thresh=0 --i420 --bias-pct=50 --minsection-pct=0 --maxsection-pct=1000 --

		arnr-maxframes=7 --arnr-strength=6 --arnr-type=3 --sharpness=0
	HDTV “High Speed”	--tune=ssim --good --cpu-used=4 --lag-in-frames=25 --min-q=0 --max-q=63 --end-usage=0 --codec=vp8 --auto-alt-ref=1 -p 1 --kf-max-dist=999999 --kf-min-dist=0 --drop-frame=0 --static-thresh=0 --i420 --bias-pct=50 --minsection-pct=0 --maxsection-pct=1000 --arnr-maxframes=7 --arnr-strength=6 --arnr-type=3 --sharpness=0
	HDTV “Normal”	--tune=ssim --good --cpu-used=2 --lag-in-frames=25 --min-q=0 --max-q=63 --end-usage=0 --codec=vp8 --auto-alt-ref=1 -p 2 --kf-max-dist=999999 --kf-min-dist=0 --drop-frame=0 --static-thresh=1 --i420 --bias-pct=50 --minsection-pct=0 --maxsection-pct=1000 --arnr-maxframes=7 --arnr-strength=6 --arnr-type=3 --sharpness=0
	HDTV “High Quality”	--tune=ssim --good --cpu-used=0 --lag-in-frames=25 --min-q=0 --max-q=63 --end-usage=0 --codec=vp8 --auto-alt-ref=1 -p 2 --kf-max-dist=999999 --kf-min-dist=0 --drop-frame=0 --static-thresh=0 --i420 --bias-pct=50 --minsection-pct=0 --maxsection-pct=1000 --arnr-maxframes=7 --arnr-strength=6 --arnr-type=3 --sharpness=0
	VideoConference	--tune=ssim --i420 -p 1 --pass=1 -t 8 --good --cpu-used=3 --kf-min-dist=0 --kf-max-dist=99999 --static-thresh=0 --drop-frame=0 --min-q=0 --max-q=63

10 Appendix 7. 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.

10.1.1.1 RD curves

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

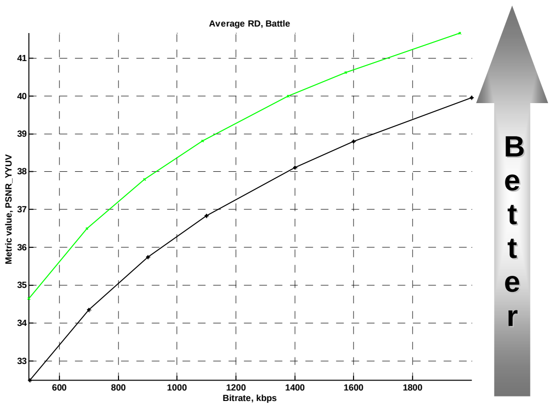
10.1.1.2 Relative Bitrate/Relative Time Charts

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.

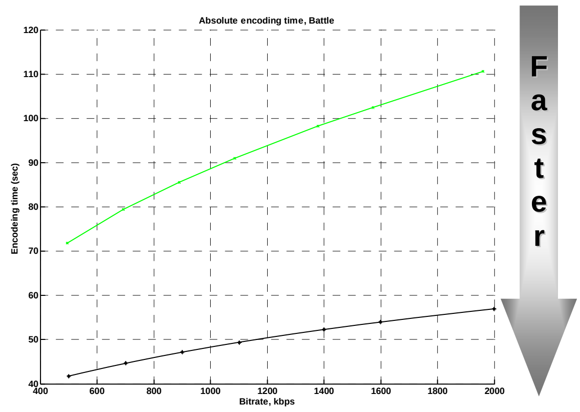
The X-axis shows the relative encoding time for the codec under test. Larger values indicate a slower codec. For example, a value of 2.5 means that the codec under test works 2.5 times slower, on average, than the reference codec.

10.1.1.3 Graph Example

Figure 270 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.



RD curve. "Green" codec is better!



Encoding time (seconds). "Green codec is slower!

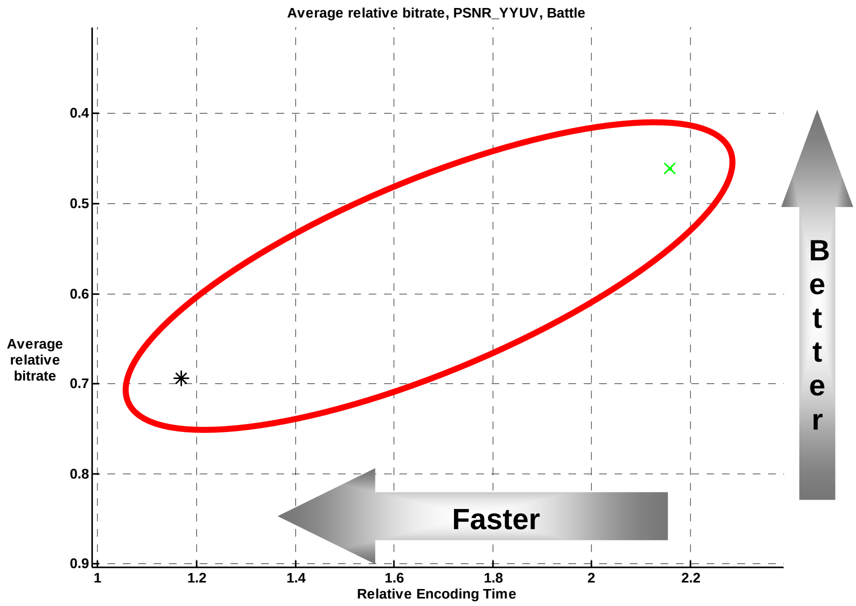


Figure 270. Integral situation with codecs. This plot shows the situation more clearly.

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.

10.2 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 272). 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 273). This result is an average bitrate ratio for a fixed quality for the two codecs. If 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.

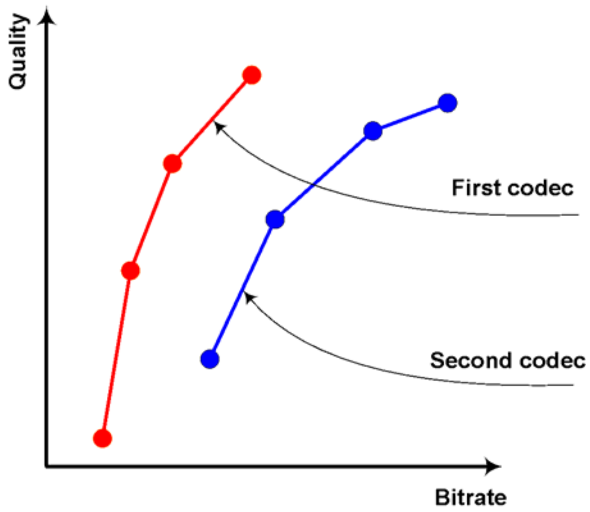


Figure 271. Source Data

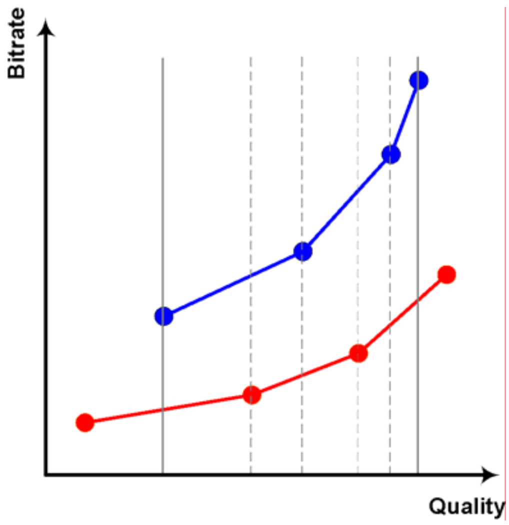


Figure 272. Axes' Inversion and Averaging Interval Choosing

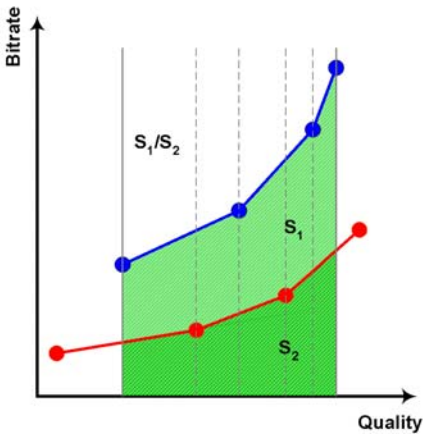


Figure 273. Areas' under Curves Ratio

11 Appendix 7. Objective Quality Metrics Description

11.1 SSIM (Structural SIMilarity)

11.1.1 Brief Description

The original paper on the SSIM metric was published by Wang, et al.[1] The paper can be found at the following URL: http://ieeexplore.ieee.org/iel5/83/28667/01284395.pdf

The SSIM author homepage is found at the following URL: 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:

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

where

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

$$\sigma_x = \left(\sum_{i=1}^{N} \omega_i (x_i - \mu_x)\right)^{\frac{1}{2}}$$

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

The constants C_1 and C_2 are defined according to the following expressions:

$$C_1=(K_1L)^2$$
$$C_2=(K_2L)^2$$

where L is the dynamic range of the pixel values (255 for 8-bit grayscale images), and $K_1, K_2 << 1$.

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

[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.

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.

11.1.2 Examples

The following is an 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.

Original Processed SSIM

Figure 274. SSIM example for compressed image

The following are more examples how various types of distortion influence the SSIM value.

Original image Image with added noise

Blurred image

Sharpen image

Figure 275. Original and processed images (for SSIM example)

The SSIM values for the Y-plane for these images are given below.

SSIM for image with itself, value = 1

SSIM for image with noisy image, value = 0.552119

SSIM for image with blurred image, value = 0.9225

SSIM for image with sharpen image, value = 0.958917

Figure 276. SSIM values for original and processed images

11.2 PSNR (Peak Signal-to-Noise Ratio)

11.2.1 Brief Description

This metric, which is often used in actual practice, is called the peak signal-to-noise ratio, or PSNR.

$$d(X,Y) = 10 \cdot \log_{10} \frac{255^2 \cdot m \cdot n}{\sum_{i=1,j=1}^{m,n} (x_{ij} - y_{ij})^2},$$

Where *d(X,Y)* – PSNR value between *X* and *Y* frames

x_{ij} – the pixel value for *(i,j)* position for the *X* frame

y_{ij} – the pixel value for *(i,j)* position for the *Y* frame

m,n – frame size *mxn*

Generally, this metric has the same form as the mean square error (MSE), but it is more convenient to use because of the logarithmic scale. It still has the same disadvantages as the MSE metric, however.

In MSU Video Quality Measurement Tool the PSNR can be calculated for all YUV and RGB components and for the L component of LUV color space. The PSNR value is quick and easy to calculate, but it is sometimes inappropriate as relates to human visual perception.

A maximum deviation of 255 is used for the PSNR for the RGB and YUV color components because, in YUV files, there is 1 byte for each color component. The maximum possible difference, therefore, is 255. For the LUV color space, the maximum deviation is 100.

The values of the PSNR in the LUV color space are in the range [0, 100]; the value 100 means that the frames are identical.

11.2.2 Examples

PSNR visualization uses different colors for better visual representation:

- Black – value is very small (99 – 100)
- Blue – value is small (35 – 99)
- Green – value is moderate (20 – 35)
- Yellow –value is high (17 – 20)
- Red –value is very high (0 – 17)

The following is an example of the PSNR metric:

Original Processed PSNR

Figure 277. PSNR example for two frames

The following are further examples demonstrating how various distortions can influence the PSNR value.

Original image

Image with added noise

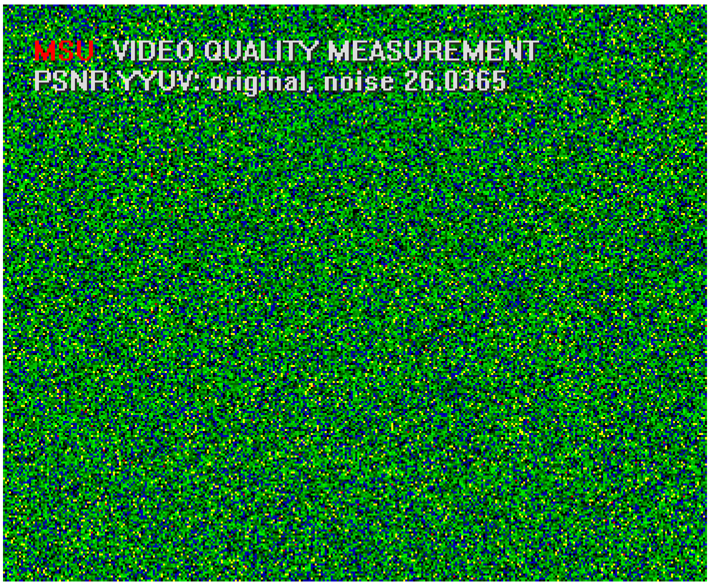
Blurred image

Sharpen image

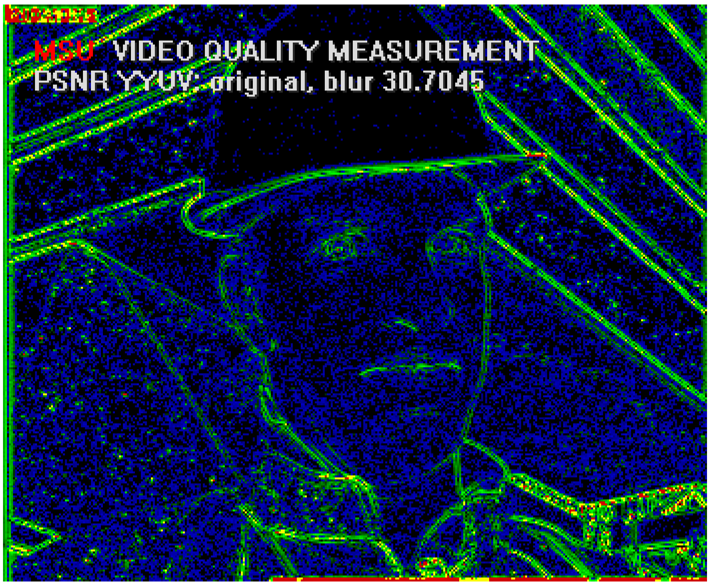
Figure 278. Original and processed images (for PSNR example)

Next are the PSNR values for the Y–plane for these images

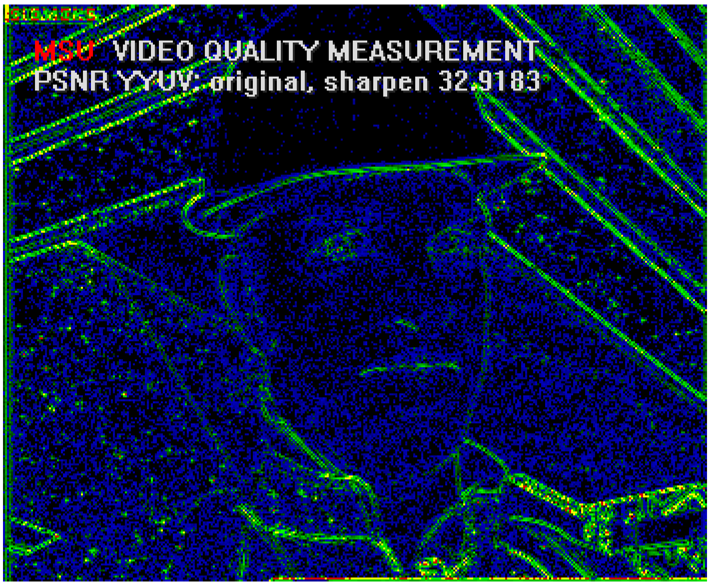
PSNR for image with itself, value = 0



PSNR for image with noisy image,
value = 26.0365



PSNR for image with blurred image,
value = 30.7045



PSNR for image with sharpen image,
value = 32.9183

Figure 279. PSNR values for original and processed images

12 List of Figures

13 About the Graphics & Media Lab Video Group

GRAPHICS & MEDIA LAB
VIDEO GROUP

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