Video Transcoding Clouds Comparison 2019

Graphics & Media Lab
Video Group

Video group head — Dr. Dmitriy Vatolin

Project head — Dr. Dmitriy Kulikov

Measurements & analysis — Egor Sklyarov, Sergey Zvezdakov, Anastasia Antsiferova

Services:

H.265

- Alibaba
- AWS Elemental MediaConvert
- Coconut
- Qencode
- Zencoder

Without H.265

- Amazon Elastic Transcoder

CS MSU Graphics & Media Lab, Video Group

November 25, 2019

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

videocodec-testing@graphics.cs.msu.ru

Contents

1. MOTIVATION

Cloud-based video-encoding services are becoming more popular owing to their simplicity and convenience, but selecting one is becoming more difficult. In our experience, the video codec may have an entirely different configuration from service to service, potentially returning output files with vastly different sizes. Our testing showed that in the case of cloud services that provide the same quality, the file-size difference can reach 100% and the price difference is 700%. This report will help you navigate the available services by highlighting the strengths and weaknesses of each one so you can decide which is best for your project.

We selected six cloud-based video-coding services with per-minute billing, opting for the most universal and affordable alternatives:

- Alibaba Cloud
- Amazon Elastic Transcoder
- AWS Elemental MediaConvert
- Coconut
- Qencode
- Zencoder

We considered other services when making our selection, but they either lacked a per-minute billing option or were difficult to access. Our next comparison will add any new services that satisfy these requirements.

The test compression employed the H.264 and HEVC standards with different settings, applying them to the same video sequences and calculating the quality losses using generally accepted metrics.

Video can be compressed independently, providing access to all video-codec settings. For more information on comparisons of popular H.264 and HEVC implementations, see our reports at http://www.compression.ru/video/codec_comparison/hevc_2018/.

2. OVERVIEW

2.1. Sequences

	Sequence	Number of frames	Frame rate	Resolution
1.	epson	599	24	1920×1080
2.	fountains	264	25	1920×1080
3.	hawk	276	30	1920×1080
4.	hockey	557	24	1920×1080

Table 1: Summary of video sequences

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

2.2. Services

For this comparison, we selected cloud-based video-coding services that are publicly available. They included the following:

Service	Developer	Test date
Alibaba	Alibaba	12.03.2019
Amazon Elastic Transcoder	Amazon	14.03.2019
AWS Elemental MediaConvert	Amazon	28.02.2019
Coconut	Coconut	26.02.2019
Qencode	Qencode	27.02.2019
Zencoder	Zencoder	12.03.2019

Table 2: Short services' descriptions

All services implement the current H.264 and HEVC compression standards (except Amazon Elastic Transcoder, which only implements H.264), so the comparison employs both standards.

Despite the extensive documentation for these services, users have few adjustable coding options. Qencode has the most options; the interface is similar to FFmpeg, albeit with some limitations.

We used AWS Elemental MediaConvert as reference service. Appendix B provides detailed descriptions of all services in our comparison.

3. MERGED RESOLUTIONS

3.1. Similar encoding options

Different services have different default options. In addition, few options are user adjustable, so knowing how good hidden options are is helpful. We selected options that nearly all services provide — level, profile and GOP — and assigned them the same average values. See Appendix B for used presets.

Even though most services lack working HEVC options, we tested them anyway. Coconut didn't describe its HEVC options, so we omitted it from this use case.

3.1.1. Cost/Quality Trade-Off

We use prices valid for October 1, 2019

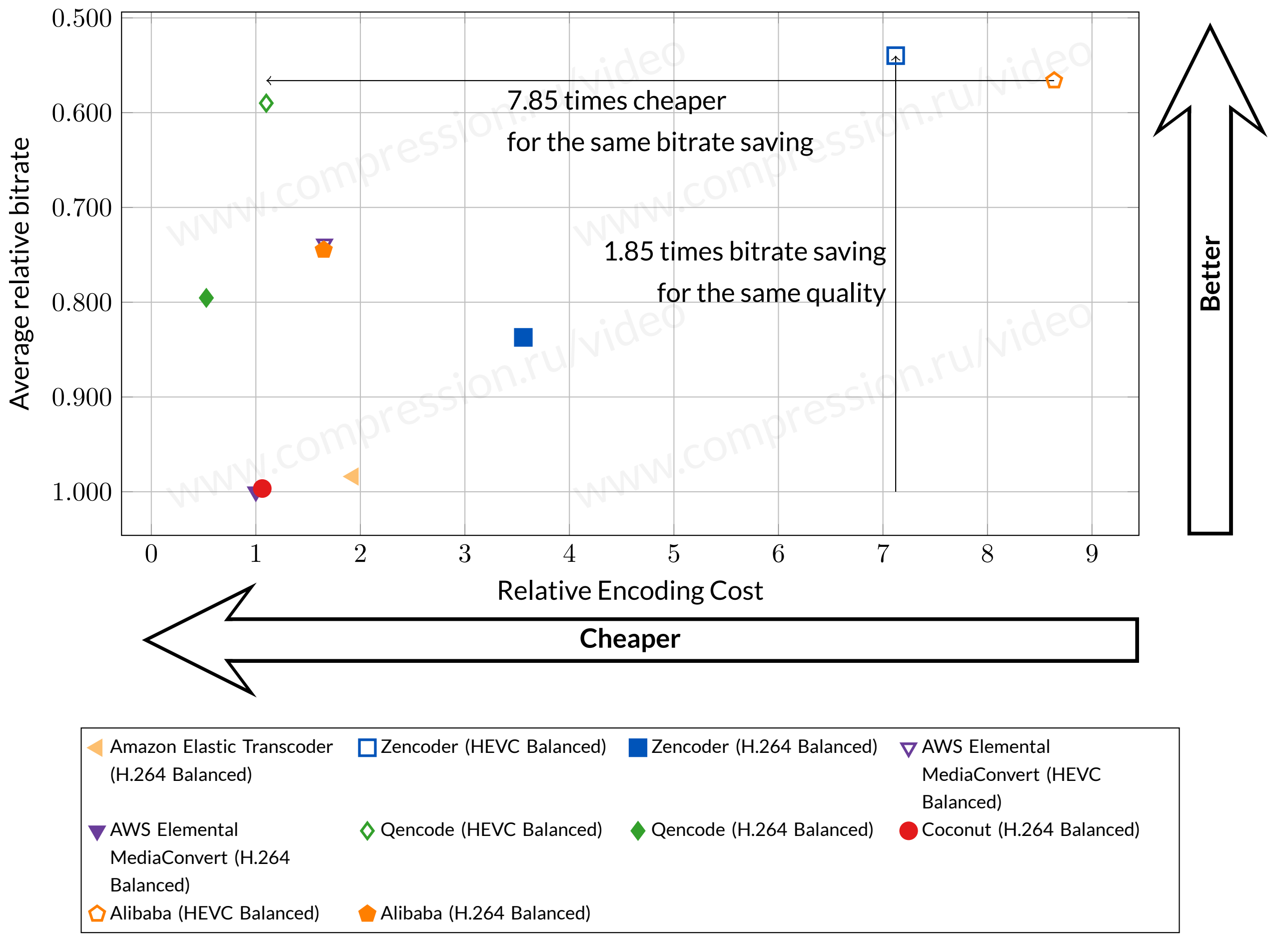


Figure 1: Cost/Quality Trade-Off—use case "Similar encoding options," all sequences, YUV-SSIM metric.

At similar bitrates, Alibaba costs nearly eight times as much as Qencode. At the same time, the output file for AWS Elemental MediaConvert is twice as large as for the leaders.

3.1.2. Bitrate Handling

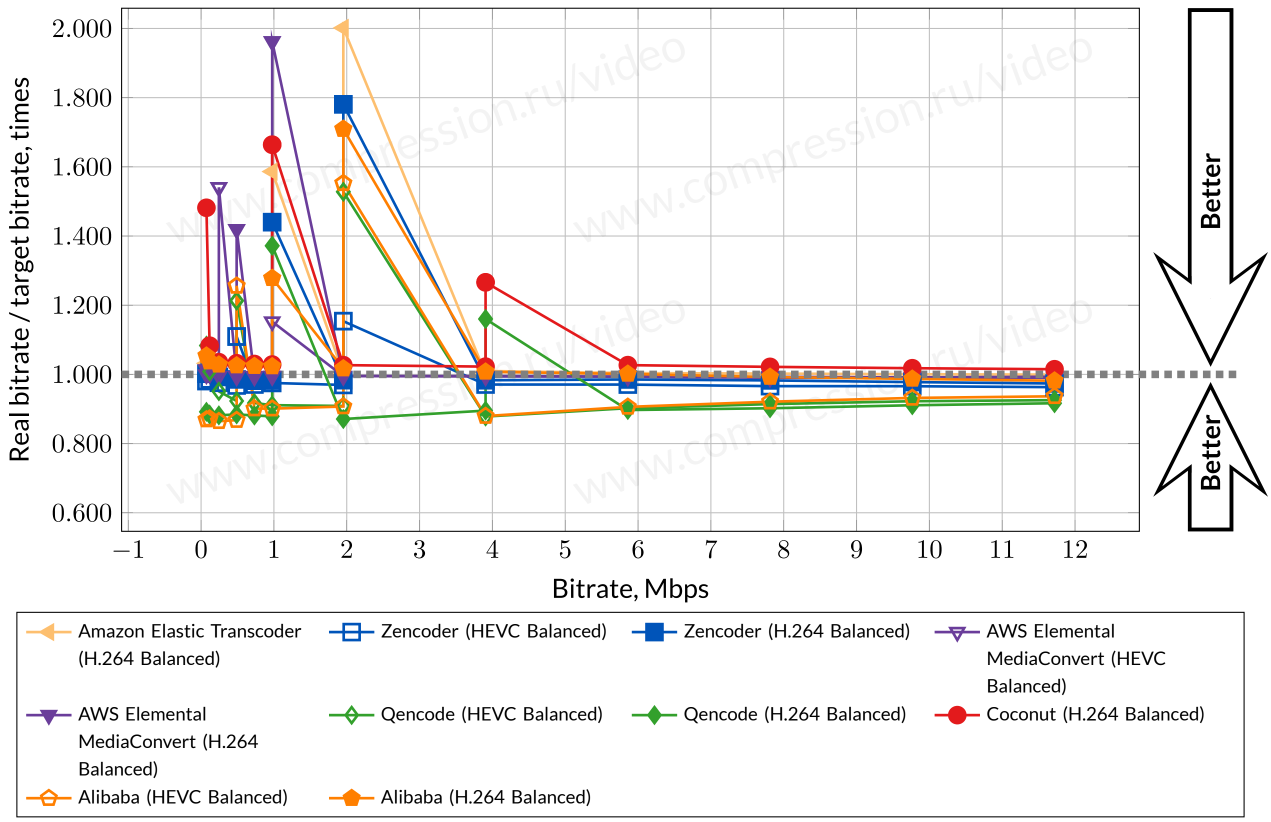


Figure 2: Bitrate handling—use case "Similar encoding options," *epson* sequence.

The explanation of measuring and comparison method is presented in Section D.4.

A distinguishing feature of a good video codec is its ability to maintain a given bitrate. On some sequences, however, it may increase the bitrate:

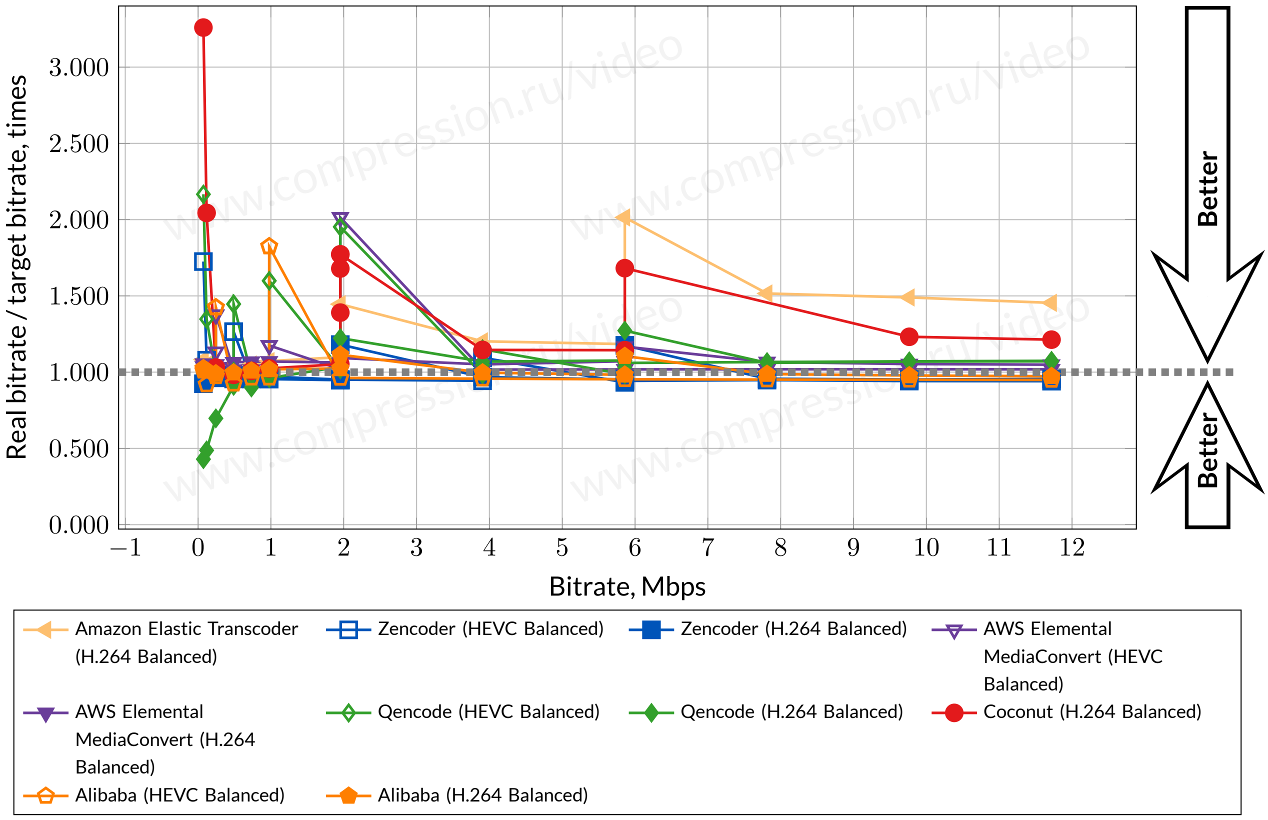


Figure 3: Bitrate handling—use case "Similar encoding options," *fountains* sequence.

The explanation of measuring and comparison method is presented in Section D.4.

The reverse situation, which is also undesirable (a codec could improve the quality while consuming the entire allocated bitrate), can occur as well:

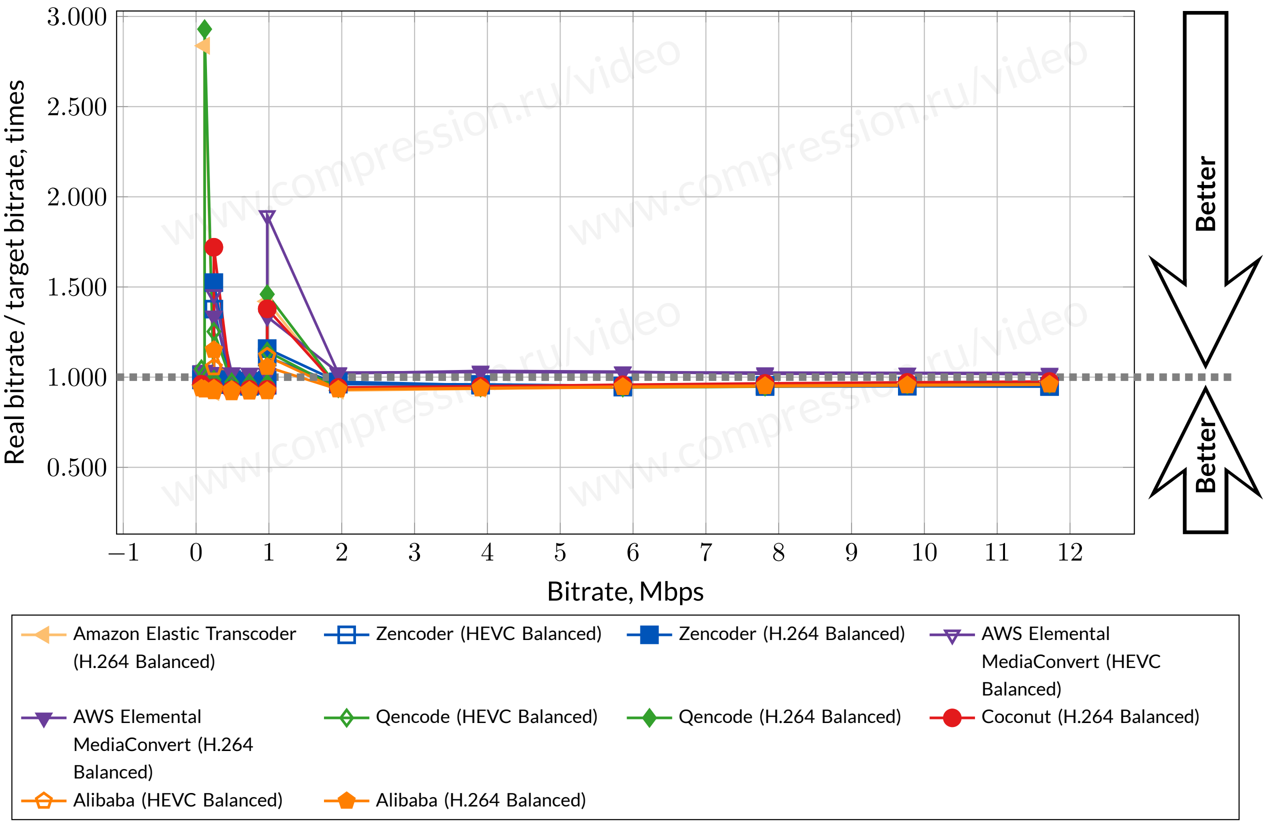


Figure 4: Bitrate handling—use case "Similar encoding options," *hawk* sequence.

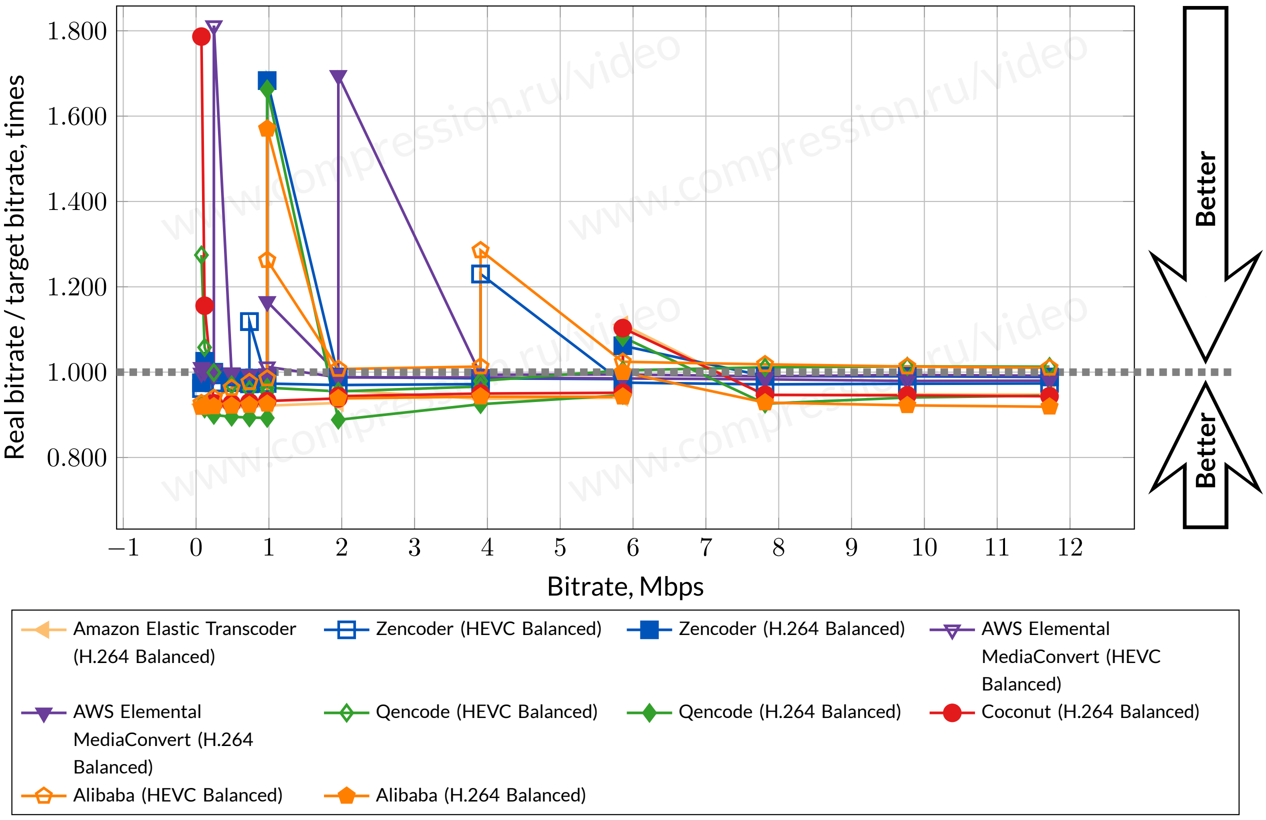


Figure 5: Bitrate handling—use case "Similar encoding options," *hockey* sequence.

The explanation of measuring and comparison method is presented in Section D.4.

3.1.3. Relative Quality Analysis

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

	Amazon Elastic Transcoder (H.264 Balanced)	Zencoder (HEVC Balanced)	Zencoder (H.264 Balanced)	AWS Elemental MediaConvert (HEVC Balanced)	AWS Elemental MediaConvert (H.264 Balanced)	Qencode (HEVC Balanced)	Qencode (H.264 Balanced)	Coconut (H.264 Balanced)	Alibaba (HEVC Balanced)	Alibaba (H.264 Balanced)
Amazon Elastic Transcoder (H.264 Balanced)	100.0%	177.0%	114.0%	131.0%	98.0%	164.0%	121.0%	97.0%	169.0%	129.0%
Zencoder (HEVC Balanced)	58.0%	100.0%	64.0%	74.0%	54.0%	93.0%	67.0%	56.0%	97.0%	73.0%
Zencoder (H.264 Balanced)	92.0%	157.0%	100.0%	113.0%	84.0%	145.0%	104.0%	89.0%	150.0%	113.0%
AWS Elemental MediaConvert (HEVC Balanced)	84.0%	139.0%	89.0%	100.0%	74.0%	129.0%	93.0%	82.0%	133.0%	100.0%
AWS Elemental MediaConvert (H.264 Balanced)	117.0%	199.0%	123.0%	137.0%	100.0%	182.0%	128.0%	114.0%	187.0%	139.0%
Qencode (HEVC Balanced)	62.0%	108.0%	70.0%	81.0%	59.0%	100.0%	73.0%	60.0%	105.0%	79.0%
Qencode (H.264 Balanced)	89.0%	154.0%	96.0%	109.0%	80.0%	141.0%	100.0%	86.0%	146.0%	108.0%
Coconut (H.264 Balanced)	103.0%	182.0%	117.0%	132.0%	100.0%	168.0%	123.0%	100.0%	173.0%	131.0%
Alibaba (HEVC Balanced)	61.0%	103.0%	67.0%	77.0%	57.0%	96.0%	70.0%	59.0%	100.0%	76.0%
Alibaba (H.264 Balanced)	81.0%	138.0%	89.0%	101.0%	74.0%	128.0%	93.0%	79.0%	133.0%	100.0%

Table 3: Average bitrate ratio for a fixed quality—use case "Similar encoding options," all sequences, YUV-SSIM metric.

Table explanation is presented in Section D.5.

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

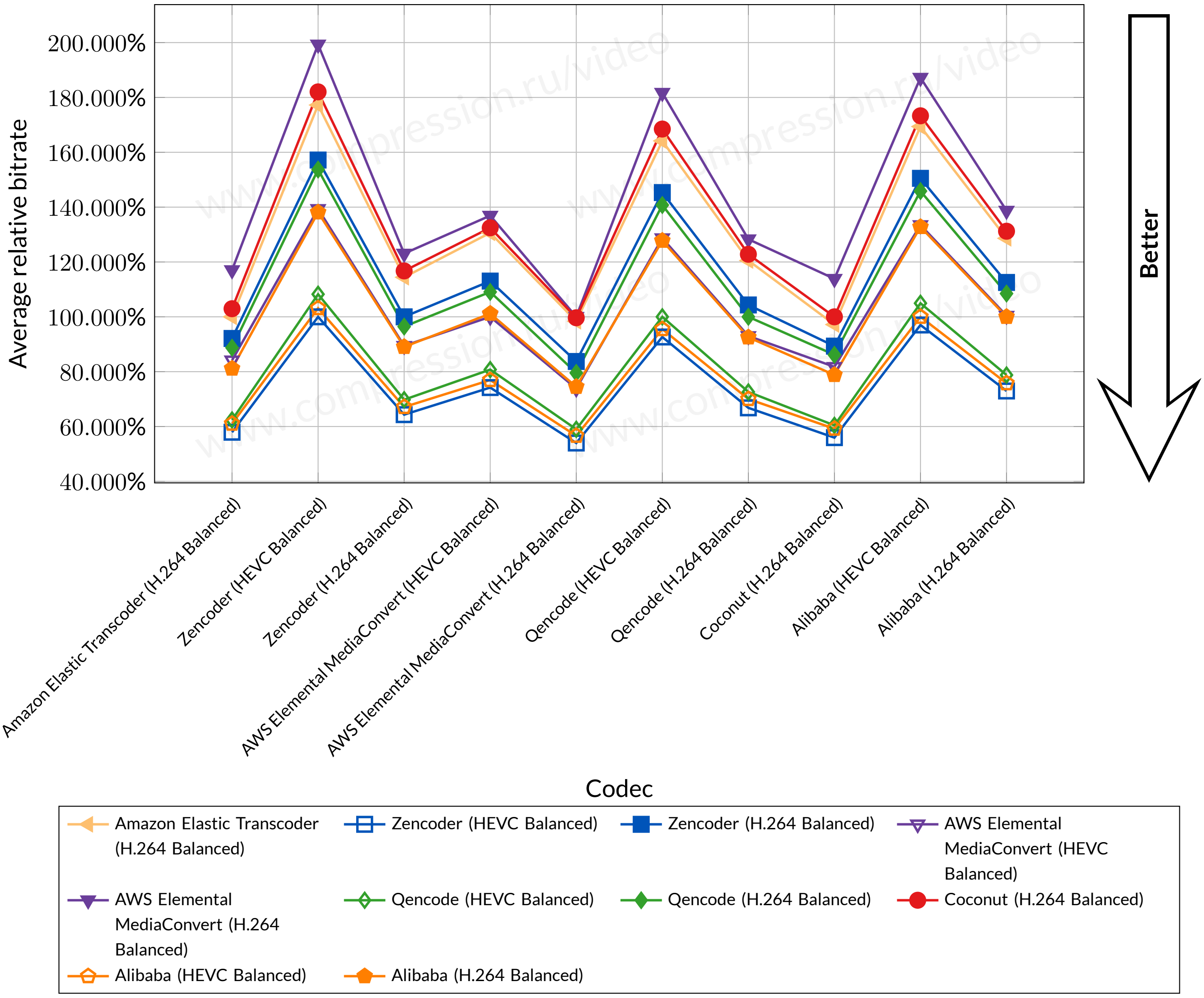


Figure 6: Average bitrate ratio for a fixed quality—use case "Similar encoding options," all sequences, YUV-SSIM metric.

3.2. Default encoding options

To simulate the simplest case for these services, we used the default settings.

Amazon Elastic Transcoder has no default options, so we omitted it from this use case.

3.2.1. Cost/Quality Trade-Off

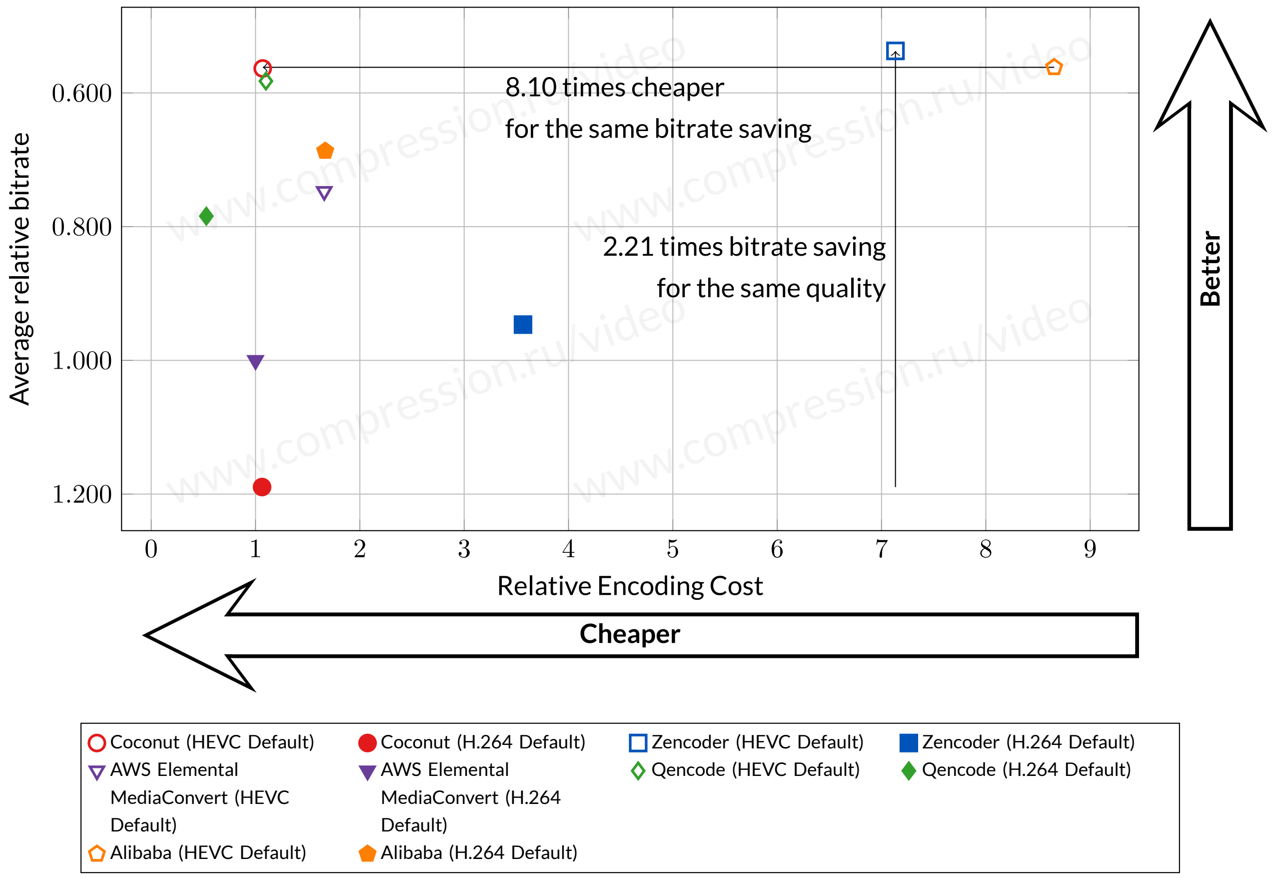


Figure 7: Cost/Quality Trade-Off—use case "Default encoding options," all sequences, YUV-SSIM metric.

The bitrate leaders, Zencoder and Alibaba, are the most expensive among all competitors. Qencode and Coconut, which follow them in bitrate, offer low prices (7–8 times cheaper, on average, than Zencoder), which is why they are leaders in the cost/quality tradeoff. Despite Coconut's success on HEVC, its H.264 results are extremely poor.

3.3. Qencode with Tuned H.264 Options

One of the projects in our lab is dedicated to video encoders parameters optimization for increasing encoding performance (`https://evt.guru`). As a part of this comparison, we applied current investigations to search for good encoding parameters that can be provided to a cloud encoder. Qencode was chosen for this part of comparison as it is the only service that provides an ability to set a big number of encoding parameters.

For each video from this comparison, we searched for a good x264 encoding configuration among all encoding options available in Qencode. The service allows varying 11 options which influence encoding performance: trellis, me_method, partitions, b_strategy, me_range, i_qfactor, qcomp, refs, directpred, sc_threshold, subq. The best configuration found for these parameters was used to encode each of four test videos on the service platform.

The results of our tuning are presented on the following graph:

3.3.1. Cost/Quality Trade-Off

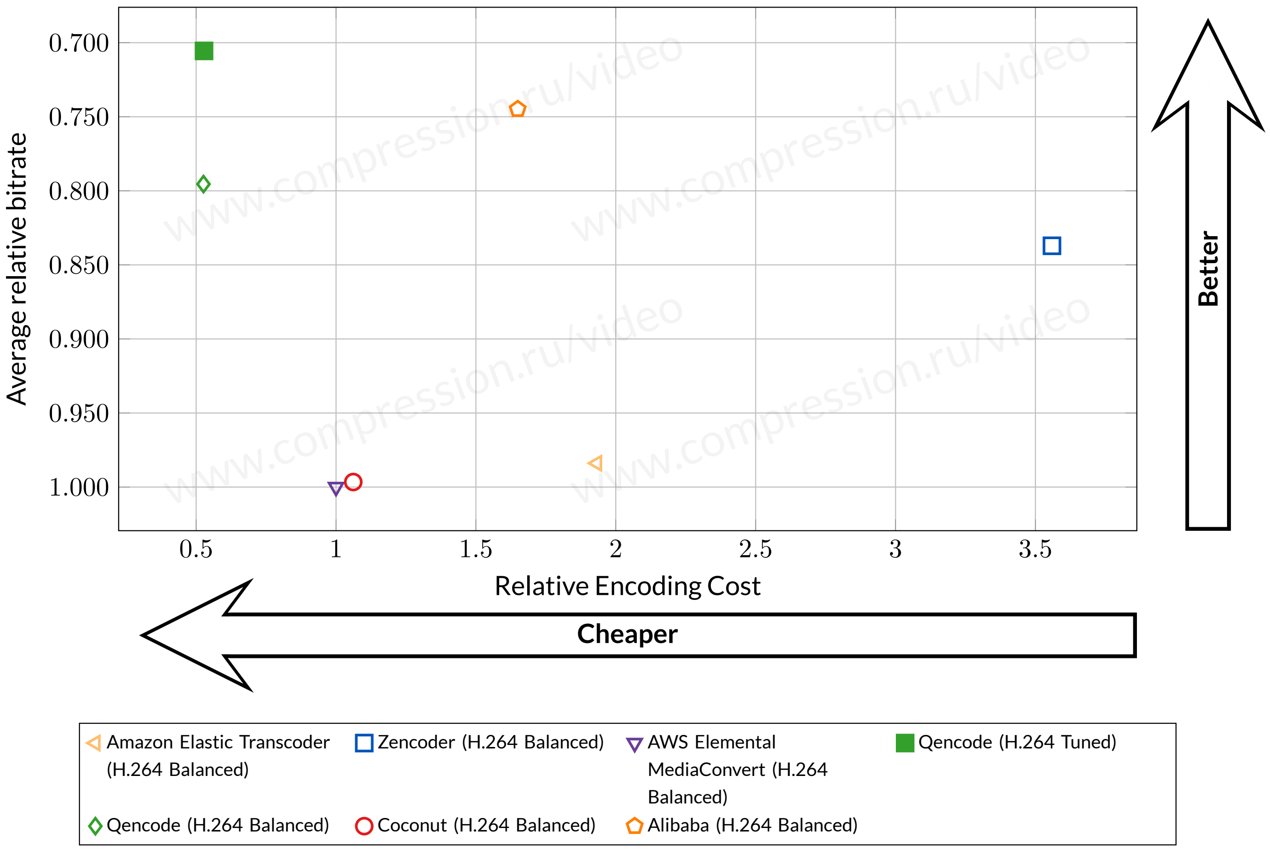


Figure 8: Cost/Quality Trade-Off—use case “Qencode with Tuned H.264 Options,” all sequences, YUV-SSIM metric.

Thanks to the large number of options available, we were able to find effective presets for Qencode for each sequence. At the same cost, we were able to improve the compression ratio by 13%. Ultimately, Tuned Qencode outperformed Alibaba and Zencoder in price and compression ratio.

4. SERVICE FEATURES

When choosing a cloud-based video-coding service, the main factor is price per minute of output video. Slightly less important is the configurability of global encoding parameters such as profile, level and GOP. In addition, the service should have a convenient way to upload and download videos.

The smaller the rank — the better.

4.1. Encoding price

Criterion	Alibaba Cloud	Amazon Elastic Transcoder	AWS Elemental MediaConvert	Coconut	Qencode	Zencoder
H.264-encoding (SD)	0.0125$	0.015$	0.0075$	0.015$	0.004$	0.05$
H.264-encoding (HD)	0.025$	0.03$	0.015$	0.015$	0.008$	0.05$
HEVC-encoding (SD)	0.0625$	Unsupported	0.012$	0.015$	0.008$	0.1$
HEVC-encoding (HD)	0.125$	Unsupported	0.024$	0.015$	0.016$	0.1$
Average cost	0.05625$	0.025$	0.014625$	0.015$	0.009$	0.075$

Table 4: Encoding cost
The prices in the table are per minute of output video

4.2. H.264 encoding options support

Criterion	Alibaba Cloud	Amazon Elastic Transcoder	AWS Elemental MediaConvert	Coconut	Qencode	Zencoder
Codec	No info	x264	No info	x264	x264	No info
Profile	Yes	Yes	Yes	Yes	Yes	Yes
Level	No	Yes	Yes	Yes	Yes	Yes
GOP	Yes	Yes	Yes	No	Yes	Yes
Rank	3	1	1	2	1	1

Table 5: H.264 encoding options

4.3. HEVC encoding options support

Criterion	Alibaba Cloud	Amazon Elastic Transcoder	AWS Elemental MediaConvert	Coconut	Qencode	Zencoder
Codec	No info	Unsupported	No info	x265	x265	No info
Profile	No	Unsupported	Yes	No	No	Yes
Level	No	Unsupported	Yes	No	No	Yes
GOP	Yes	Unsupported	Yes	No	Yes	Yes
Rank	2	–	1	3	2	1

Table 6: HEVC encoding options

4.4. Options for storing and transferring video

Criterion	Alibaba Cloud	Amazon Elastic Transcoder	AWS Elemental MediaConvert	Coconut	Qencode	Zencoder
FTP/SFTP	No	No	No	Yes	Yes	Yes
Amazon S3	No	Yes	Yes	Yes	Yes	Yes
Rank	3	2	2	1	1	1

Table 7: Storing and transferring options

4.5. Overall Conclusion about Available Features

- Cheapest service: Qencode
- Most expensive service: Zencoder
- Support for HEVC options almost universally poor
- Amazon Elastic Transcoder only implements H.264
- Alibaba Cloud only works with its own storage (incompatible with S3), complicating its use outside the Alibaba Cloud infrastructure
- Amazon Elastic Transcoder and AWS Elemental MediaConvert only run Amazon S3

5. CONCLUSION

5.1. Merged resolutions

The plot below shows overall quality scores for the encoders in our comparison (see Section D for a description of the integral-score computation method). First place in the quality competition goes to **Zencoder (HEVC Balanced)**, second place goes to **Alibaba (HEVC Balanced)**, and third place to **Qencode (HEVC Balanced)**.

5.1.1. Similar encoding options

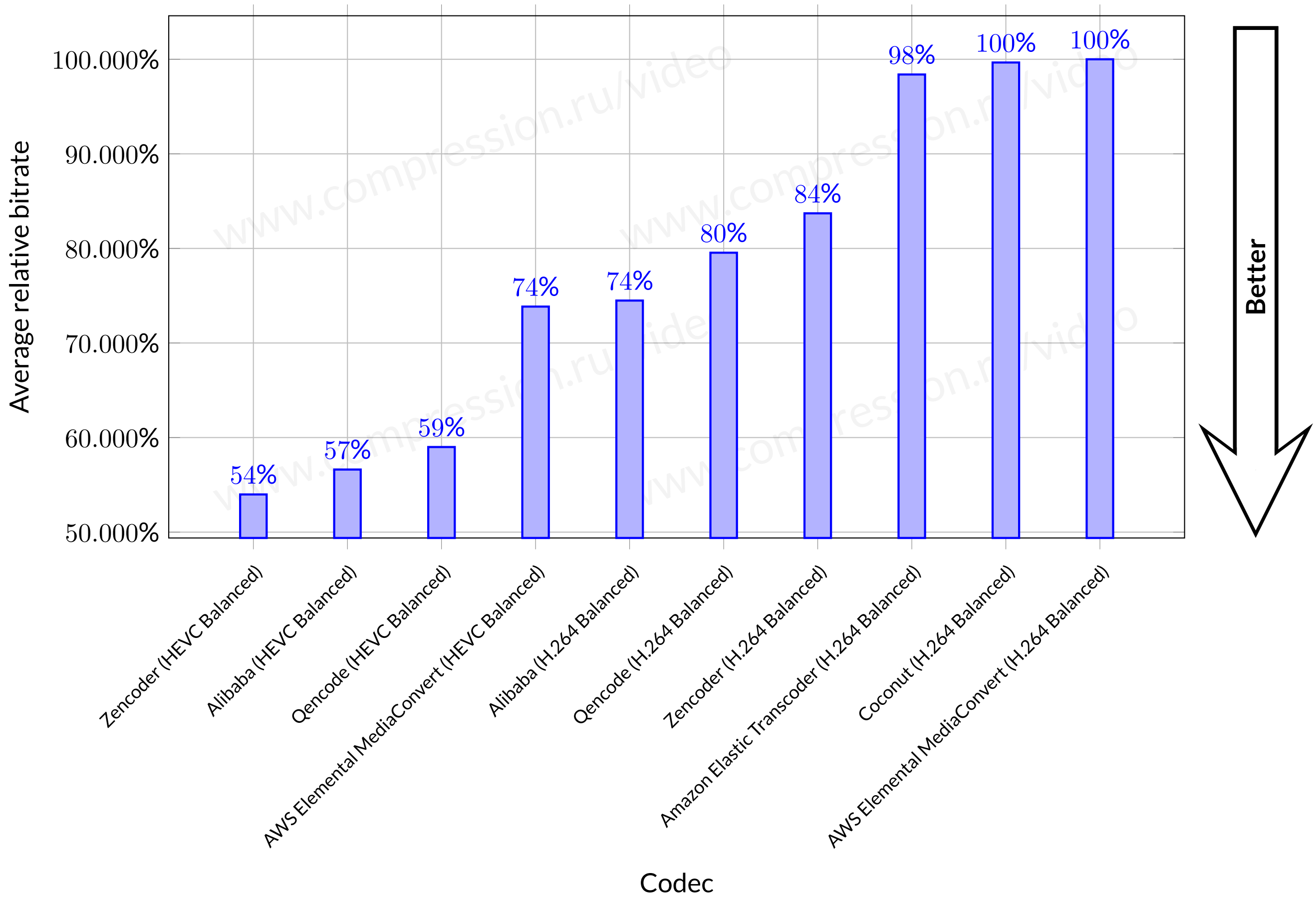


Figure 9: Average bitrate ratio for a fixed quality—use case "Similar encoding options," all sequences, YUV-SSIM metric.

A more recent compression standard — HEVC — exhibits better results. Zencoder works great with HEVC, but with H.264 it loses to its closest competitors. Alibaba and Qencode, however, are close behind Zencoder in HEVC coding.

Owing to the specifics of cloud coding services, considering quality without considering price is pointless.

The plot below shows overall quality scores for the encoders in our comparison (see Section D for a description of the integral-score computation method). First place in the quality competition goes to **Qencode (H.264 Tuned)**, second place goes to **Alibaba (H.264 Balanced)**, and third place to **Qencode (H.264 Balanced)**.

5.1.2. Qencode with Tuned H.264 Options

This part of the comparison involved pre-measurements to find good encoding parameters for each video, so using Qencode in this way requires more time. However, our latest results proved the ability of good encoding configurations predicting without additional encoder launches, and we hope to include them in the next comparison.

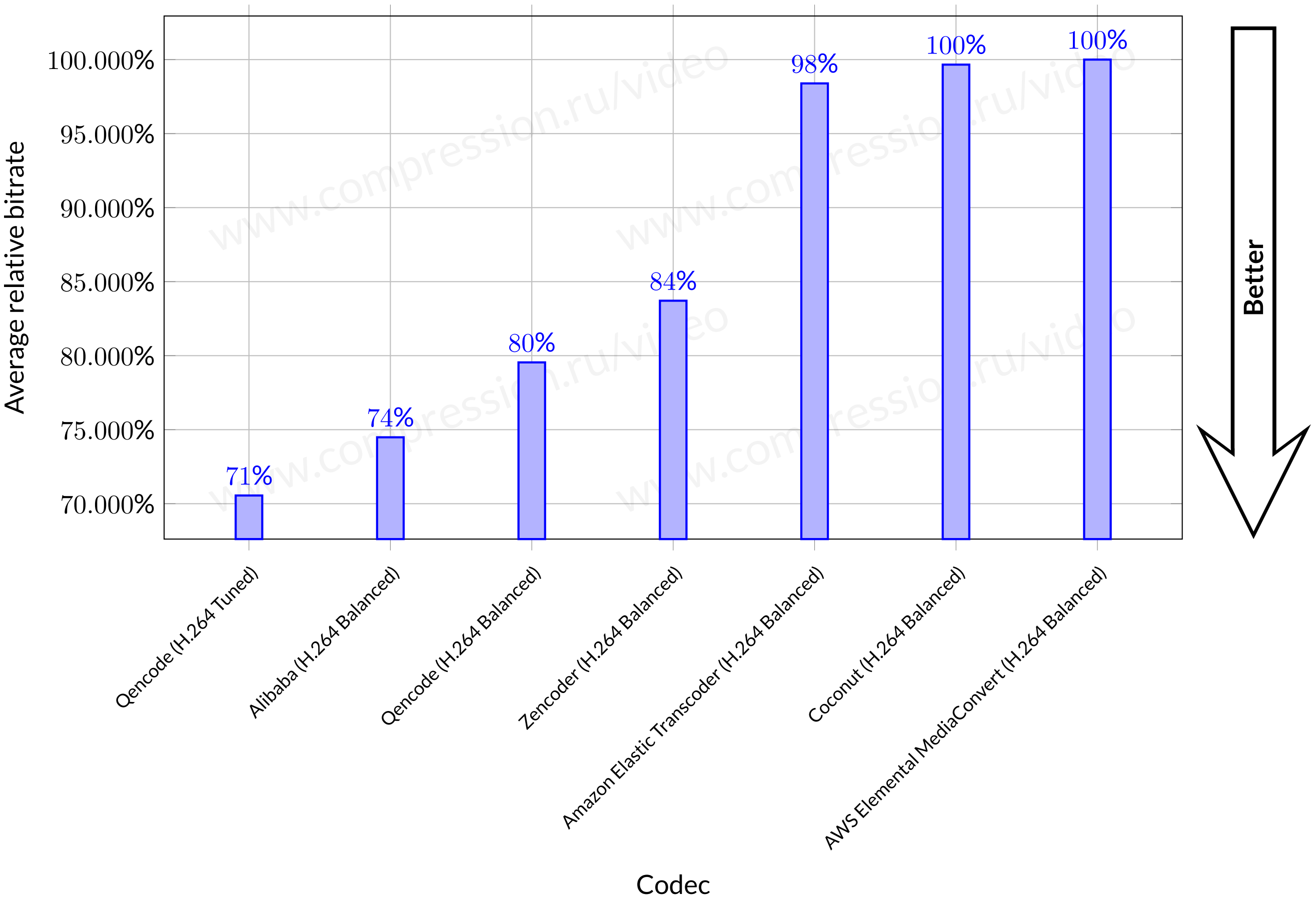


Figure 10: Average bitrate ratio for a fixed quality—use case "Qencode with Tuned H.264 Options," all sequences, YUV-SSIM metric.

5.2. Overall

Criterion	Alibaba Cloud	Amazon Elastic Transcoder	AWS Elemental MediaConvert	Coconut	Qencode	Zencoder
H.264 options	3	1	1	2	1	1
HEVC options	2	4 (unsupported)	1	3	2	1
Storage	3	2	2	1	1	1
Cost/Quality H.264	1	4	6	3	2	5
Cost/Quality HEVC	3	5 (unsupported)	4	1	1	2
Bitrate handling	1	1	2	3	2	1
Average rank	2.16	2.83	2.66	2.16	**1.5**	1.83

Table 8: Average rank for each service

For real-world projects, the most interesting chart is the one that shows the cost/quality tradeoff (see Section 3.1.1).

- Boasting high compression efficiency and low price, Qencode and Coconut HEVC are the winners.
- Zencoder and Alibaba cloud encoding services shows the best results considering only compression efficiency.
- H.264 encoding is cheaper, but the loss of compression efficiency becomes noticeable.

A. SEQUENCES

A.1. epson

Sequence title	epson
Resolution	1920×1080
Number of frames	599
Color space	YV12
Frames per second	24
Source	https://vimeo.com/120309899
Source resolution	FullHD

Epson printer commercial, frequently changing short scenes.

Figure 11: epson sequence, frame 648

A.2. fountains

Sequence title	fountains
Resolution	1920×1080
Number of frames	264
Color space	YV12
Frames per second	25
Source	https://vimeo.com/92772987
Source resolution	FullHD

Static camera, several fountains in front of which people walk.

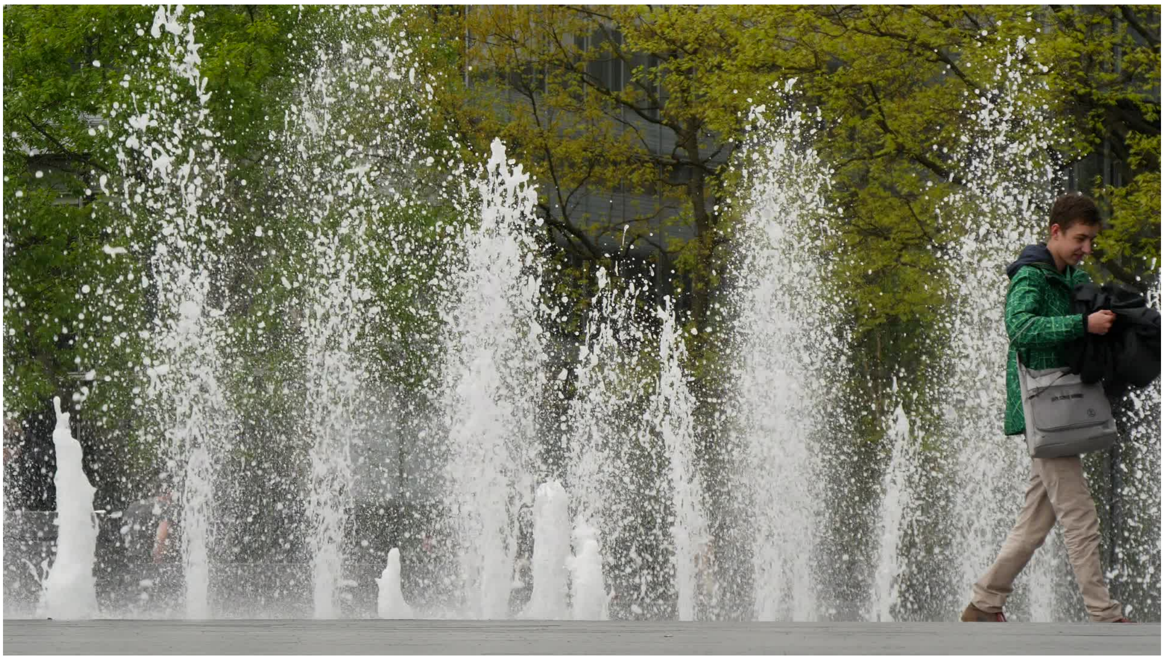


Figure 12: fountains sequence, frame 0

A.3. hawk

Sequence title	hawk
Resolution	1920×1080
Number of frames	276
Color space	YV12
Frames per second	30
Source	https://vimeo.com/109321182
Source resolution	FullHD

Hawk sitting on a branch against the background of the sea.

Figure 13: hawk sequence, frame 1505

A.4. hockey

Sequence title	hockey
Resolution	1920×1080
Number of frames	557
Color space	YV12
Frames per second	24
Source	https://vimeo.com/117014819
Source resolution	FullHD

Video about the hockey team, contains dynamic and slow scenes.

Figure 14: hockey sequence, frame 79

B. SERVICES

B.1. Alibaba

Service title	Alibaba
Test date	12.03.2019
Developed by	Alibaba

Encoding cost:

- H.264 (SD) — 0.0125$
- H.264 (HD) — 0.025$
- HEVC (SD) — 0.0625$
- HEVC (HD) — 0.125$

Features:

- High default settings
- Native OSS storage is incompatible with S3
- Depends heavily on Alibaba Cloud infrastructure

Preset name	Encoder parameters
H.264 Balanced	`Codec=H.264, Video.Profile=main, Video.GOP=250`
H.264 Default	`Codec=H.264`
H.264 Balanced FullHD	`Codec=H.264, Height=1080, Width=1920, Video.Profile=main, Video.GOP=250`
H.264 Default FullHD	`Codec=H.264, Height=1080, Width=1920`
H.264 Balanced SD	`Codec=H.264, Height=480, Width=854, Video.Profile=main, Video.GOP=250`
H.264 Default SD	`Codec=H.264, Height=480, Width=854`

H.264 Balanced HD	`Codec=H.264, Height=720, Width=1280, Video.Profile=main, Video.GOP=250`
H.264 Default HD	`Codec=H.264, Height=720, Width=1280`
HEVC Balanced	`Codec=H.265, Video.Profile=main, Video.GOP=250`
HEVC Default	`Codec=H.265`
HEVC Balanced FullHD	`Codec=H.265, Height=1080, Width=1920, Video.Profile=main, Video.GOP=250`
HEVC Default FullHD	`Codec=H.265, Height=1080, Width=1920`
HEVC Balanced SD	`Codec=H.265, Height=480, Width=854, Video.Profile=main, Video.GOP=250`
HEVC Default SD	`Codec=H.265, Height=480, Width=854`
HEVC Balanced HD	`Codec=H.265, Height=720, Width=1280, Video.Profile=main, Video.GOP=250`
HEVC Default HD	`Codec=H.265, Height=720, Width=1280`

B.2. Amazon Elastic Transcoder

Service title	Amazon Elastic Transcoder
Test date	14.03.2019
Developed by	Amazon

Encoding cost:

- H.264 (SD) — 0.015$
- H.264 (HD) — 0.03$

Features:

- Lacks HEVC support
- More expensive than AWS Elemental MediaConvert
- Inconvenient preset creation
- Depends heavily on AWS infrastructure

Preset name	Encoder parameters
H.264 Balanced	`Codec=H.264, KeyframesMaxDist=250, FixedGOP=true, CodecOptions.Level=4.1, CodecOptions.Profile=main`
H.264 Balanced FullHD	`Codec=H.264, Resolution=1920x1080, KeyframesMaxDist=250, FixedGOP=true, CodecOptions.Level=4.1, CodecOptions.Profile=main`
H.264 Balanced SD	`Codec=H.264, Resolution=854x480, KeyframesMaxDist=250, FixedGOP=true, CodecOptions.Level=4.1, CodecOptions.Profile=main`
H.264 Balanced HD	`Codec=H.264, Resolution=1280x720, KeyframesMaxDist=250, FixedGOP=true, CodecOptions.Level=4.1, CodecOptions.Profile=main`

B.3. AWS Elemental MediaConvert

Service title	AWS Elemental MediaConvert
Test date	28.02.2019
Developed by	Amazon

Encoding cost:

- H.264 (SD) — 0.0075$
- H.264 (HD) — 0.015$
- HEVC (SD) — 0.012$
- HEVC (HD) — 0.024$

Features:

- Doesn't accept Lossless H.264 High 4:4:4
- Convenient preset configurator in console
- Cheaper than Amazon Elastic Transcoder
- Depends heavily on AWS infrastructure

Preset name	Encoder parameters
H.264 Balanced	`Codec=H_264, GopSize=250, CodecLevel=LEVEL_4_1, CodecProfile=MAIN`
H.264 Default	`Codec=H_264`
H.264 Balanced FullHD	`Codec=H_264, Height=1080, Width=1920, GopSize=250, CodecLevel=LEVEL_4_1, CodecProfile=MAIN`
H.264 Default FullHD	`Codec=H_264, Height=1080, Width=1920`
H.264 Balanced SD	`Codec=H_264, Height=480, Width=854, GopSize=250, CodecLevel=LEVEL_4_1, CodecProfile=MAIN`
H.264 Default SD	`Codec=H_264, Height=480, Width=854`
H.264 Balanced HD	`Codec=H_264, Height=720, Width=1280, GopSize=250, CodecLevel=LEVEL_4_1, CodecProfile=MAIN`

H.264 Default HD	`Codec=H_264, Height=720, Width=1280`
HEVC Balanced	`Codec=H_265, GopSize=250, CodecLevel=LEVEL_5, CodecProfile=MAIN_MAIN`
HEVC Default	`Codec=H_265`
HEVC Balanced FullHD	`Codec=H_265, Height=1080, Width=1920, GopSize=250, CodecLevel=LEVEL_5, CodecProfile=MAIN_MAIN`
HEVC Default FullHD	`Codec=H_265, Height=1080, Width=1920`
HEVC Balanced SD	`Codec=H_265, Height=480, Width=854, GopSize=250, CodecLevel=LEVEL_5, CodecProfile=MAIN_MAIN`
HEVC Default SD	`Codec=H_265, Height=480, Width=854`
HEVC Balanced HD	`Codec=H_265, Height=720, Width=1280, GopSize=250, CodecLevel=LEVEL_5, CodecProfile=MAIN_MAIN`
HEVC Default HD	`Codec=H_265, Height=720, Width=1280`

B.4. Coconut

Service title	Coconut
Test date	26.02.2019
Developed by	Coconut

Encoding cost:

- Any encoding — 0.015$

Features:

- The simplest service
- Tasks may freeze indefinitely
- Technical support is nonresponsive
- Poor bitrate handling

Preset name	Encoder parameters
H.264 Balanced	`->mp4:h264, x264_vprofile=main, x264_level=41`
H.264 Default	`->mp4:h264`
H.264 Balanced FullHD	`->mp4:h264_1080p, x264_vprofile=main, x264_level=41`
H.264 Default FullHD	`->mp4:h264_1080p`
H.264 Balanced SD	`->mp4:h264_480p, x264_vprofile=main, x264_level=41`
H.264 Default SD	`->mp4:h264_480p`
H.264 Balanced HD	`->mp4:h264_720p, x264_vprofile=main, x264_level=41`
H.264 Default HD	`->mp4:h264_720p`
HEVC Default	`->mp4:hevc`

HEVC Default FullHD	`->mp4:hevc_1080p`
HEVC Default SD	`->mp4:hevc_480p`
HEVC Default HD	`->mp4:hevc_720p`

B.5. Qencode

Service title	Qencode
Test date	27.02.2019
Developed by	Qencode

Encoding cost:

- H.264 (SD) — 0.004$
- H.264 (HD) — 0.008$
- HEVC (SD) — 0.008$
- HEVC (HD) — 0.016$

Features:

- Numerous options from FFmpeg
- Occasional poor bitrate handling

Preset name	Encoder parameters
H.264 Balanced	`video_codec=libx264, keyframes=250, profile=main, video_codec_parameters.level=41`
H.264 Default	`video_codec=libx264`
H.264 Balanced FullHD	`video_codec=libx264, height=1080, keyframes=250, profile=main, video_codec_parameters.level=41`
H.264 Default FullHD	`video_codec=libx264, height=1080`
H.264 Balanced SD	`video_codec=libx264, height=480, keyframes=250, profile=main, video_codec_parameters.level=41`
H.264 Default SD	`video_codec=libx264, height=480`
H.264 Balanced HD	`video_codec=libx264, height=720, keyframes=250, profile=main, video_codec_parameters.level=41`
H.264 Default HD	`video_codec=libx264, height=720`

HEVC Balanced	`video_codec=libx265, keyframes=250, profile=main, video_codec_parameters.level=50`
HEVC Default	`video_codec=libx265`
HEVC Balanced FullHD	`video_codec=libx265, height=1080, keyframes=250, profile=main, video_codec_parameters.level=50`
HEVC Default FullHD	`video_codec=libx265, height=1080`
HEVC Balanced SD	`video_codec=libx265, height=480, keyframes=250, profile=main, video_codec_parameters.level=50`
HEVC Default SD	`video_codec=libx265, height=480`
HEVC Balanced HD	`video_codec=libx265, height=720, keyframes=250, profile=main, video_codec_parameters.level=50`
HEVC Default HD	`video_codec=libx265, height=720`

B.6. Zencoder

Service title	Zencoder
Test date	12.03.2019
Developed by	Zencoder

Encoding cost:

- H.264 (SD) — 0.05$
- H.264 (HD) — 0.05$
- HEVC (SD) — 0.1$
- HEVC (HD) — 0.1$

Features:

- Convenient control panel

Preset name	Encoder parameters
H.264 Balanced	`codec=h264, profile=main, level=4.1`
H.264 Default	`codec=h264`
H.264 Balanced FullHD	`codec=h264, size=1920x1080, profile=main, level=4.1`
H.264 Default FullHD	`codec=h264, size=1920x1080`
H.264 Balanced SD	`codec=h264, size=854x480, profile=main, level=4.1`
H.264 Default SD	`codec=h264, size=854x480`
H.264 Balanced HD	`codec=h264, size=1280x720, profile=main, level=4.1`
H.264 Default HD	`codec=h264, size=1280x720`
HEVC Balanced	`codec=hevc, profile=main, level=5.0`

HEVC Default	`codec=hevc`
HEVC Balanced FullHD	`codec=hevc, size=1920x1080, profile=main, level=5.0`
HEVC Default FullHD	`codec=hevc, size=1920x1080`
HEVC Balanced SD	`codec=hevc, size=854x480, profile=main, level=5.0`
HEVC Default SD	`codec=hevc, size=854x480`
HEVC Balanced HD	`codec=hevc, size=1280x720, profile=main, level=5.0`
HEVC Default HD	`codec=hevc, size=1280x720`

C. MEASUREMENT METHOD

Our comparison used three video-sequence resolutions:

- 1080p (1,920x1,080) — the original size of the video sequence
- 720p (1,280x720)
- 480p (854x480)

Calculation of the metrics for 720p and 480p employed the Lanczos scaling algorithm in the FFmpeg implementation. The size of the encoded sequence is the size of the encoded stream extracted from the container. For each resolution, we selected a ladder of target bitrates (seven per resolution).

Each service has its own default settings, and most of them are fixed. To compare hidden options, we introduced a similar options use case in which the settings available for most services are aligned: profile, level and GOP. There is also a default use case in which all options remain at their default values.

From the output we obtained an RD curve (quality dependence on bitrate) from seven points for each resolution, standard, use case, sequence and service. To produce the overall picture, we combined the curves for different resolutions, choosing the points with the highest quality at a similar bitrate.

D. FIGURE EXPLANATION

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

D.1. RD Curves

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

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

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

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

D.3. Graph Example

Figure 15 shows a situation where these graphs can be useful. In the top-left graph, the "Green" codec clearly produces better quality than the "Black" codec. On the other hand, the top-right graph shows that the "Green" codec is slightly slower. Relative-bitrate/relative-time graphs can be useful in precisely these situations: the bottom graph clearly shows that one codec is slower but yields higher visual quality, whereas the other codec is faster but yields lower visual quality.

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

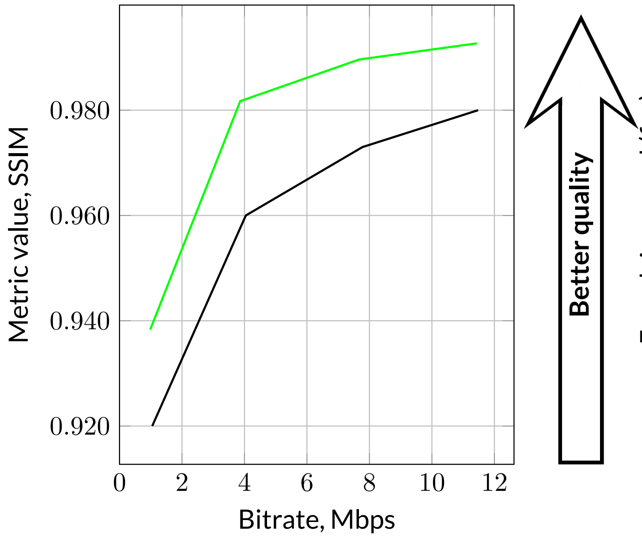
A more detailed description of how we prepared these graphs appears below.

D.4. Bitrate Ratio for the Same Quality

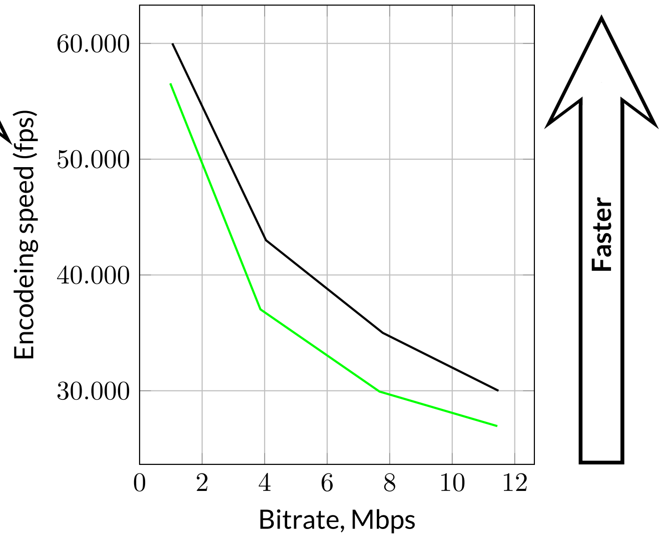
The first step in computing the average bitrate ratio for a fixed quality is to invert the axes of the bitrate/quality graph (see Figure 16b). All further computations use the inverted graph.

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

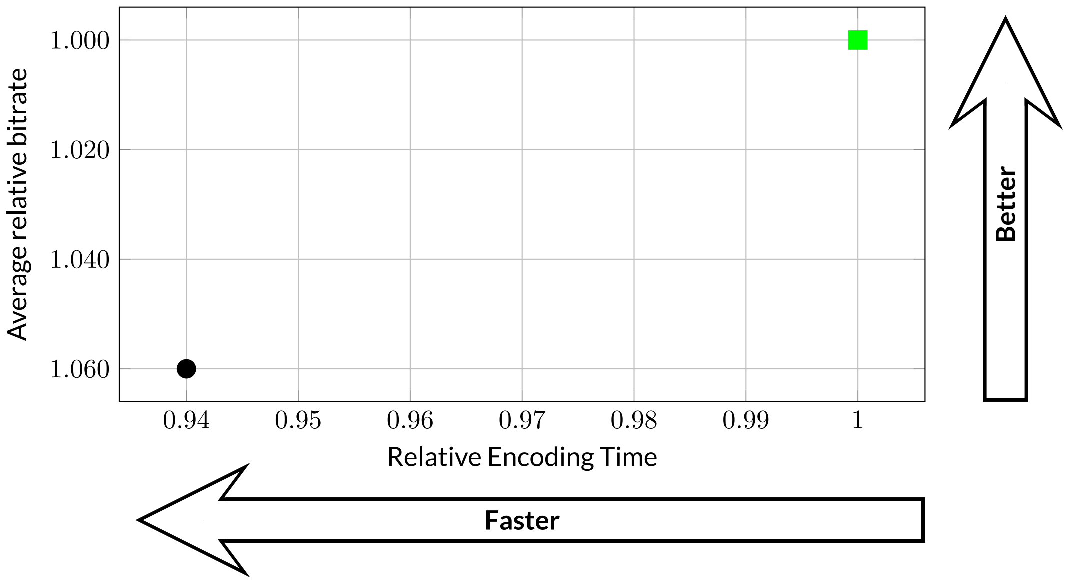
The final step is calculation of the area under the curves in the chosen interpolation segment and determination of their ratio (see Figure 16c). This result is an average bitrate ratio at a fixed quality for the two codecs. When



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

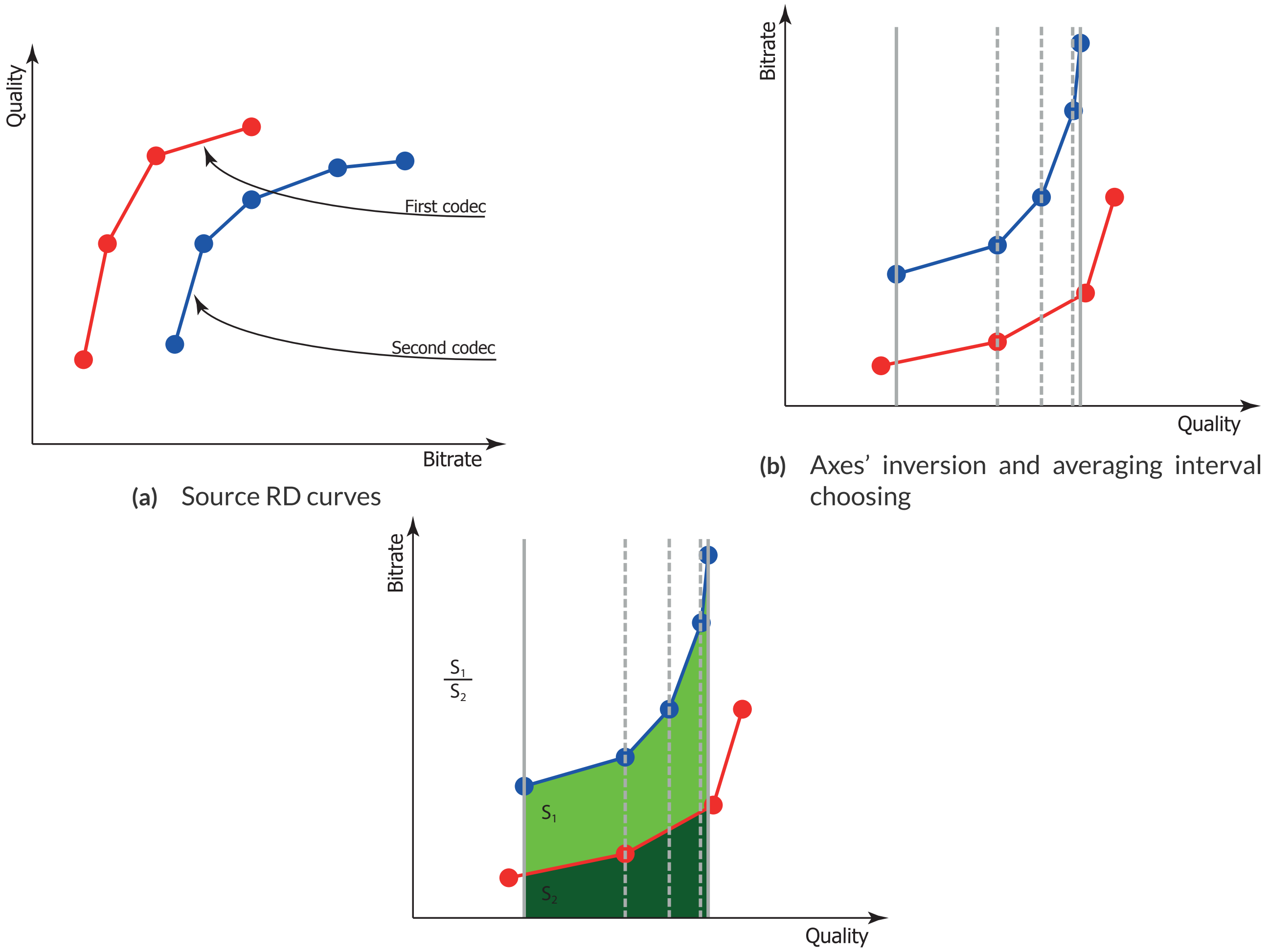


(b) Encoding speed (frames per second). "Green" codec is slower!



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

Figure 15: Speed/Quality trade-off example



(a) Source RD curves

(b) Axes' inversion and averaging interval choosing

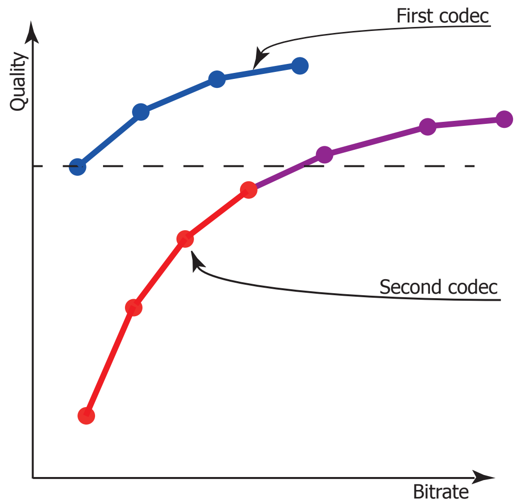
(c) Areas under curves ratio

Figure 16: Average bitrate ratio computation

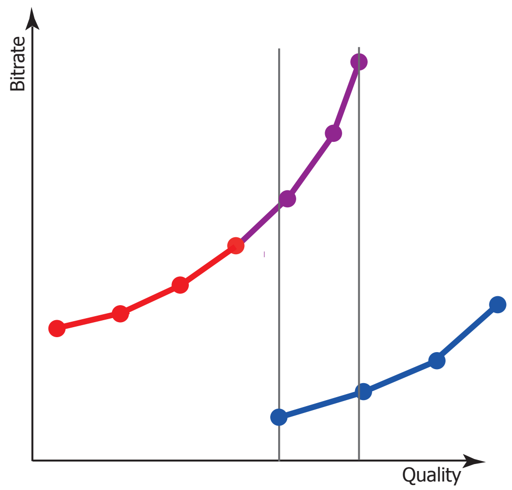
considering more than two codecs, one of is defined as a reference codec, and the quality of the others is compared with that of the reference.

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

If no segment exists for which two codecs both produce encoding results, we measured the results for additional higher and/or lower bitrates. The schematic example (Figure 17) shows that the results for these extra bitrates (purple) cross with codec two and enable a comparison with codec one.



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

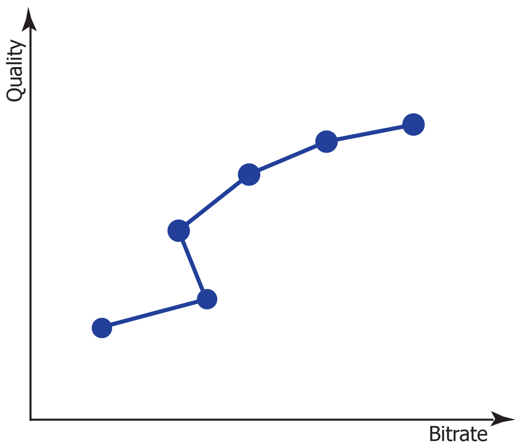


(b) Axes' inversion and averaging interval choosing

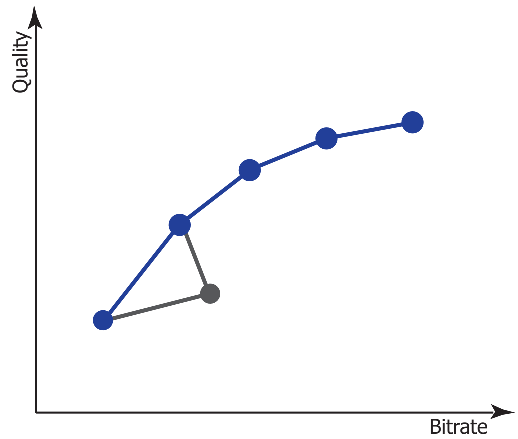
Figure 17: Measuring codec on additional bitrates to make it cross with other codecs over the quality axis.

D.4.2. When RD Curves Are Non-monotonic

Sometimes, especially on complex videos, the encoding results for neighboring bitrates vary greatly owing to the codec's operating characteristics. This situation leads to a non-monotone RD curve, which we process as follows: for each point, use the next point at the target bitrate that has greater or equal quality. This technique yields the reduced monotonic curve, which appears in the example of Figure 18.



(a) Non-monotonic RD-curve.



(b) Points that were used to calculate integral.

Figure 18: Processing non-monotonic RD-curves.

D.5. Relative Quality Analysis

Although most figures in this report provide codec scores relative to a reference encoder (i.e., x264), the "Relative Quality Analysis" sections provide the bitrate ratio at a fixed quality score (see Section D.4) for each codec pair. This approach may be useful when comparing codec A relative with codec B only.

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

	A	B
A	100% ☺	75% ☹
B	134% ☹	100% ☺

Table 15: Example of average bitrate ratio for a fixed quality table

Consider column "B", row "A" of the table, which contains the value 75%. This number should be interpreted in the following way: the average bitrate for Codec B at a fixed quality is 75% less than that for codec A. The icon in the cell depicts the confidence of this estimate. If projections of RD curves on the quality axis (see Figure 16) have large common areas, the cell contains a happy icon. If this overlapping area is small, and thus the bitrate-score calculation is unreliable, the cell contains a sad icon.

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

E. OBJECTIVE-QUALITY METRIC DESCRIPTION

E.1. SSIM (Structural Similarity)

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

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

Below is a brief description of SSIM computation.

E.1.1. Brief Description

Wang, et al.[1] published the original paper on SSIM. This paper available at http://ieeexplore.ieee.org/iel5/83/28667/01284395.pdf. The SSIM author homepage is http://www.cns.nyu.edu/~lcv/ssim/

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

- Luminance
- Contrast
- Structure

The final formula, after combining these comparisons, is

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

where

$$\mu_x = \sum_{i=1}^{N} \omega_i x_i, \quad (3)$$

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

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

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

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

For our implementation, one SSIM value corresponds to two sequences. The value is in the range $[-1, 1]$, with higher values being more desirable (a value of 1 corresponds to identical frames). One advantage of the SSIM

[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.

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

E.1.2. Examples

Figure 19 shows an example SSIM result for an original and processed (compressed with lossy compression) image. The value of 0.9 demonstrates that the two images are very similar.

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

Figure 19: SSIM example for compressed image

Figure 20 depicts various distortions applied to the original image, and Figure 21 shows SSIM values for these distortions.

(a) Original image

(b) Image with added noise

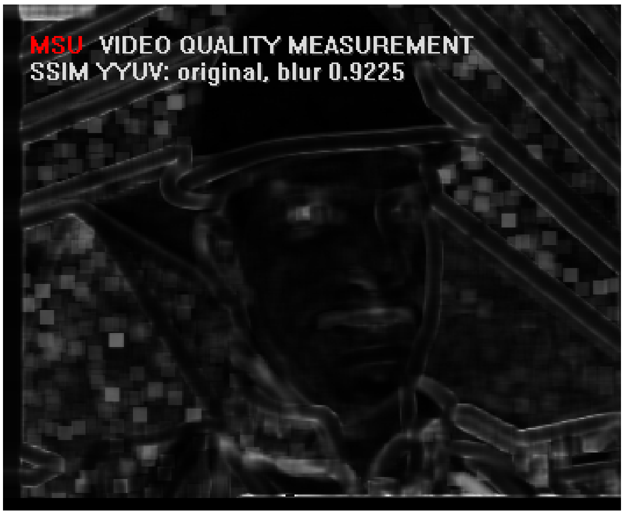
(c) Blurred image

(d) Sharpen image

Figure 20: Examples of processed images

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

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



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

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

Figure 21: SSIM values for original and processed images

E.1.3. Measurement method

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

Run the command

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

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

F. ABOUT THE GRAPHICS & MEDIA LAB VIDEO GROUP

Graphics & Media Lab
Video Group

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

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

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

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

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

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

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

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

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