Encoding AV1 video with ffmpeg

What is AV1?

- Open, royalty-free video coding format
- Succeeds VP9 video codec
- Founded by Amazon, Cisco, Google, Intel, Microsoft, Mozilla, Netflix

Neat things about AV1

- Film grain synthesis
- Warped / global motion vectors (3D space)
- T shaped partitions
- Upscaling

rav1e encoder copypasta

ffmpeg -i input.mkv -c:v librav1e -qp 80 -speed 4 -tile-columns 2 -tile-rows 2 -an -sn rav1e.mkv

- -qp 80: Lower qp value = bigger file size and better quality (values 0-255)
- -speed 4: Lower speed value = better quality, longer encode times (values 0-10)
- -tile-columns 2: More columns = slightly lower quality, faster encode times, faster decode times (values 0+)
- -an: Don’t encode audio (can be replaced with “-c:a libopus -ac 2” for stereo opus sound)
- -sn: Don’t encode subtitles (can be replaced with “-c:s copy” to copy subtitle stream over)
- https://www.ffmpeg.org/ffmpeg-all.html#librav1e for more options

aom encoder copypasta

```
ffmpeg -i input.mkv -c:v libaom-av1 -cpu-used 4 -crf 30 -b:v 0 -pix_fmt yuv420p10le -g 240 -aom-params enable-dnl-denoising=0:film-grain-table=1920x1080-BT2020-ISO6400 -an -sn aom.mkv
```

- -cpu-used 4: Lower cpu-used value = better quality, longer encode times (values 0-8)
- -crf 30: Lower crf value = bigger file size and better quality (values 0-63)
- -b:v 0 -pix_fmt yuv420p10le: enables 10bit colour, reduces colour banding, smaller file size, same quality, delete this for 8bit colour. Best to stick with 10bit – minimal drawbacks.
- -g 240: Insert a key-frame every 240 frames, without this seeking aom encoded videos can be very taxing. Suggest to have a key frame every 10 seconds
- -aom-params enable-dnl-denoising=0:film-grain-table=1920x1080-BT2020-ISO6400: aom's noise synthesis sucks for videos longer than 8 minutes, use pre-made photon noise tables. BT2020=10bit colour, SRGB=8bit colour, higher ISO=more noise. Omit this if you don't care about noise synthesis.
- https://www.ffmpeg.org/ffmpeg-all.html#libaom_002dav1 for more options
- http://rhubarb.us.to/av1/Photon-Noise-Tables.tar.xz for pre-computed noise tables

svt-av1 encoder copypasta

```
ffmpeg -i input.mkv -c:v libsvtav1 -preset 4 -qp 30 -sc_detection true -pix_fmt yuv420p10le -g 240 -an -sn svt-av1.mkv
```

- -preset 4: Lower preset value = better quality, longer encode times (values 0-13)
- -qp 30: Lower qp value = bigger file size and better quality (values 0-63)
- -sc_detection true: Enables scene change detection (better key-frame placement?)
- -pix_fmt yuv420p10le: enables 10bit colour, reduces colour banding, smaller file size, same quality, delete this for 8bit colour. Best to stick with 10bit – minimal drawbacks.
- -g 240: Insert a key-frame every 240 frames, without this seeking aom encoded videos can be very taxing. Suggest to have a key frame every 10 seconds
- https://www.ffmpeg.org/ffmpeg-all.html#libsvtav1 for more options

svt-av1 annoying bits

- Late February 2022 ffmpeg git introduces “--svtav1-params”
- Which means, currently fancy svt-av1 stuff isn’t in ffmpeg.
- Which means, currently it’s a lot more ugly encode svt-av1 videos in ffmpeg for fancy stuff.

svt-av1 encoder ugly copypasta

- Create raw AV1 bitstream:

  ffmpeg -i input.mkv -pix_fmt yuv420p10le -f yuv4mpegpipe -strict -1 - | SvtAv1EncApp -i stdin --preset 2 --keyint 240 --input-depth 10 --crf 30 --film-grain 20 --output svt-av1.ivf

- Put raw AV1 bitstream in to a suitable video container:

  ffmpeg -i svt-av1.ivf -i input.mkv -map 0:v -map 1:a:0 -map 1:s:0 -c:v copy -an -sn svt-av1.mkv

- --keyint 240: Insert a key-frame every 240 frames, Suggest to have a key frame every 10 seconds
- --crf 30: Lower crf value = bigger file size and better quality (values 1-63)
- --film-grain 20: More film-grain = more noise (values 0-50)
- https://github.com/AOMediaCodec/SVT-AV1/blob/master/Docs/svt-av1_encoder_user_guide.md for more SvtAv1EncApp options

vmaf – video quality metric

ffmpeg -i compressed.mkv -i original.mkv -filter_complex libvmaf -f null -

- Spits out computed vmaf value of compressed.mkv video, higher value=better quality, if you compare original with original, vmaf≈98
- https://www.ffmpeg.org/ffmpeg-all.html#libvmaf for more options

AV1 playback

- All web browsers except safari
- VLC, mpv, K-Lite codec pack, windows 10/11 support requires add-on from ms store
- Linux / BSD has great support, as does ChromeOS, modern Android and Windows
- No support on macOS and iOS (can use vlc, chromium / firefox though)
- Hardware acceleration with nvidia 3000 series GPUs, most AMD 6000 series GPUs, intel 11th gen+ integrated GPUs, mandated (?) for Android 12+ (?) phones and TVs.
- Encoded AV1 tiles allow for multi-threaded playback on CPUs

Containers

- mkv can hold any video / audio / subtitle codec, great for home video collection on PC

  mkvtoolnix is a great tool to mux videos together

- webm allows easy video playback on firefox / chromium-based web browsers

avif images

avifenc --help

- Supports transparency (with compressed alpha channel)
- Supports all encoder flags (including noise, qp, crf, speed, bit depth, etc)
- Can use rav1e / aom / svt-av1 encoder
- Gimp allows exporting images in avif format
- Full Chromium and Firefox support, no macOS / iOS / Safari support
- Excellent support on Linux, Windows, Android 12+

Mad Max: Fury road 1min sample

file name	codec	speed	noise	bit depth	encoding s	file size	vmaf
aom_cpu1_noise6400	aom	1	6400	8		7.5	92.894918
aom_cpu2_noise20	aom	2	20	8		7.4	91.793266
aom_cpu4_10bit	aom	4	0	10		7.5	93.282105
aom_cpu4_2pass_crf25_noise6400_key240_10bit	aom	4	6400	10		16.7	94.543443
aom_cpu4_2pass_crf27_noise6400_key240_10bit	aom	4	6400	10		13.5	94.130374
aom_cpu4_2pass_crf30_noise6400	aom	4	6400	8		9.5	93.324334
aom_cpu4_crf30_noise6400_10bit	aom	4	6400	10		9.4	93.4376
aom_cpu4_crf30_noise6400_key240_10bit	aom	4	6400	10		10	93.454281
aom_cpu4_noise20	aom	4	20	8		7.6	91.69929
aom_cpu4_noise6400	aom	4	6400	8		7.6	92.692259
input.mkv	h264		0	8		177.5	98.831822
new_svt-av1_qp30_speed2_10bit	svt-av1	2	0	10		12.9	95.433
new_svt-av1_qp30_speed4_10bit	svt-av1	4	0	10		13	95.305128
new_svt-av1_qp30_speed6_10bit	svt-av1	6	0	10		12.2	95.200209
new_svt-av1_qp30_speed8_10bit	svt-av1	8	0	10		13.9	94.896908
rav1e	rav1e		0	8		10.6	91.626023
rav1e_0tiles	rav1e		0	8		10.4	91.654112
rav1e_2x2_qp40_speed4	rav1e	4	0	8		56.4	96.132527
rav1e_2x2_qp80_speed4	rav1e	4	0	8		17.6	93.435392
rav1e_8tiles	rav1e		0	8		10.6	91.626023
svt-av1_qp30_noise20_speed0_10bit	svt-av1	0	20	10		9.3	92.663373
svt-av1_qp30_noise20_speed2	svt-av1	2	20	8		12.7	92.51167
svt-av1_qp30_noise20_speed2_10bit	svt-av1	2	20	10		9.4	92.614055
svt-av1_qp30_speed0	svt-av1	0	0	8		16.4	94.555599
svt-av1_qp30_speed1	svt-av1	1	0	8		16.3	94.516653
svt-av1_qp30_speed2	svt-av1	2	0	8		16.3	94.460944
svt-av1_qp30_speed2_10bit	svt-av1	2	0	10	0.018	13.3	94.443591
x264	x264		0	8		29.1	93.805389

Deep Space 9 full episode sample

file name	codec	speed	noise	bit depth	file size	vmaf
new_aom_cpu4_noise20_10bit	aom	4	20	10	51.3	76.849315
new_aom_cpu4_noise20_key240_10bit	aom	4	20	10	52.9	76.866572
new_svt-av1_qp30_speed6_10bit	svt-av1	6	0	10	78.3	78.817002
aom_cpu4_noise20.mkv	aom	4	20	8	85.4	86.77642
rav1e_8tiles.mkv	rav1e		0	8	85.9	87.173676
rav1e.mkv	rav1e		0	8	86.2	87.173676
rav1e_0tiles.mkv	rav1e		0	8	83.1	87.194068
svt-av1_qp30_noise20_speed4.mkv	svt-av1	4	20	8	118.2	89.030908
aom_cpu4_noise10.mkv	aom	4	10	8	90.7	89.174667
svt-av1_qp30_noise20_speed4_10bit.mkv	svt-av1	4	20	10	100.8	89.406603
rav1e_2x2_qp80_speed4.mkv	rav1e	4	0	8	115	89.534055
aom_cpu4_noise6400.mkv	aom	4	6400	8	84	90.004104
svt-av1_qp30_noise10_speed6_10bit.mkv	svt-av1	6	10	10	106.6	90.7045
x264.mkv	x264		0	8	228.8	91.396208
input.mkv	mpeg2		0	8	1300	98.759634

Useful info

- Doom9’s VP9 and AV1 forum:
  https://forum.doom9.org/forumdisplay.php?f=84
- Reddit’s AV1 threads:
  https://www.reddit.com/r/AV1/
- My temporary AV1 info:
  https://rhubarb.us.to/av1/