English version is derived from the [x264](https://www.nazorip.site/archives/63) x265 Ultimate Tutorial by iAvoe/iiAvoe

|  |  |
| --- | --- |
| [**LigH**](http://www.mediafire.com/?6lfp2jlygogwa) | .hevc GCC10 [single .exe 8-10-12bit] w/ x86 w/ libx265.dll |
| [**Rigaya**](https://drive.google.com/drive/u/0/folders/0BzA4dIFteM2dWEpvWGZXV3ZhdTA) | .hevc GCC 9.3 [8-10-12bit] w/ x86 |
| [**Patman**](http://www.mediafire.com/folder/arv5xmdqyiczc) | .hevc GCC 11+MSVC1925 [8-10-12bit] |
| [**ShortKatz**](https://forum.doom9.org/showthread.php?p=1937773#post1937773) | arm64~64e with x86 ? [?] macOS compiling needed |
| **[DJATOM-aMod](https://github.com/DJATOM/x265-aMod/releases/)** | Intel, AMD zen1~2 [10bit], zen3 [10-12bit] GCC 10.2.1+GCC10.3 |
| **[MeteorRain-yuuki](https://down.7086.in/)** | lsmash.mkv/mp4或.hevc [lavf isn't as reliable as pipe acc. rumor] GCC 9.3+ICC 1900+MSVC 1916 [8][10][12bit]+[8-10-12bit] |
| [**ffmpeg**](http://ffmpeg.org/download.html) all OS compatible. backup link: ottverse.com/ffmpeg-builds | |
| [**mpv**](https://mpv.io/installation/) **player** a small sized opensource video player with no color issues afaik | |
| [**x265GuiEx**](https://drive.google.com/drive/folders/0BzA4dIFteM2dRkRzWXZMT0lkM2M) **(Rigaya)** 日本語, compiles by auto-setup, [link](https://aviutl.info/x265guiex/#toc4) for tutorial | |
| [**Voukoder**; **V-Connector**](https://www.voukoder.org/)free Premiere/Vegas/AE/Davinci Studio export plugin with ffmpeg's internal encoders, and good encoding presets to choose | |
| [**ffprobe**](http://ffmpeg.org/download.html)metadata & media format reader in CLI from the same origin as ffmpeg (within the ffmpeg download zip file), see [tutorial page](https://nazorip.site/archives/169/) (with webpage translate) | |

### x265.exe command line for new users

[Download ffmpeg, ffprobe/MediaInfo & x265 to a memorable path] Here they are under D:\

[Open CMD/PowerShell or Linux/MacOS Bash/Terminal, input path\to\ffmpeg, ffprobe, x265, then press enter] i.e., here they are D:\x265-10bit.exe -V and D:\ffmpeg.exe

[Check ffmpeg build ver.] C:\folder\ffmpeg.exe; [x265 build ver.] C:\folder\x265.exe -V

[Auto-filling] Write PATH/filename partially, and hit [Tab]

### Get source video metadata w/ ffprobe:

ffprobe.exe -i ".\video.mp4" -select\_streams v:0 -v error -hide\_banner -show\_streams -show\_frames -read\_intervals "%+#1" -show\_entries frame=top\_field\_first:stream=codec\_long\_name,width,coded\_width,height,coded\_height,pix\_fmt,color\_range,field\_order,r\_frame\_rate,avg\_frame\_rate,nb\_frames -of ini

### Variable framerate:

Used on mobile devices to save battery, causing compatibility issues. Add ffmpeg option -vsync cfr to convert to cfr

### Rectangular pixel:

old & unsupported lossy compression. Swap src video if possible

### Encoding duration:

number of frames÷encoding speed (fps)=required time(second)

### ffmpeg-pipe-x265 example:

D:\ffmpeg.exe -i F:\video.mov -an -pix\_fmt yuv420p10 -f yuv4mpegpipe -strict unofficial - | D:\x265-10bit.exe --preset slow --me umh --subme 5 --merange 48 --weightb --aq-mode 4 --bframes 5 --ref 3 --hash 2 --allow-non-conformance --qg-size 16 --rd 3 --limit-modes --limit-refs 1 --rskip 1 --splitrd-skip --no-sao --tskip --colorprim bt2020 --colormatrix bt2020nc --transfer smpte2084 --y4m - --output F:\done.hevc 2>D:\Desktop\ffmpeg\_or\_x265\_error\_logs.txt

### Get the correct -pix\_fmt value:

Get source video metadata with MediaInfo by dragging the file onto its program window (first-time use may require selecting [View]-[Tree]), find [Colorspace], [Color Sampling] & [Bit Depth]. (see ffprobe's method above). Usually they are [YUV], [4:2:0] & [8bit], which corresponds to yuv420p from the list below, x265 has a smaller set of supported formats than ffmpeg, they are:

yuv420p, yuv422p, yuv444p, yuv420p10le, yuv420p12le, yuv422p10le, yuv422p12le, yuv444p10le, yuv444p12le, yuv444p10le, yuv444p12le, gray, gray10le, gray12le, nv12, nv16

### Select bit depth for x264/5:

One single x265.exe contain 8-10-12bit (Check with x265.exe -V) are set with option -D, such as -D 10 to encode in 10bit; or the downloaded zip contains separate x265-8bit.exe, x265-10bit.exe, simply call the corresponding executable

### Triangle rule of encoding:



The methodical way to develop encoding strategy & hardware choice. A dedicated solution delivers an unbalanced encoding result with heavy compromise, a general purposed solution in contrast provides a balanced solution that works for most applications. Faster hardware fits with general purposed, slower often fits dedicated solution

### ffmpeg, VS, avs2yuv pipe

ffmpeg -i video\_in.mp4 -an -f yuv4mpegpipe -strict unofficial - | x265 --y4m --input - --output

ffmpeg -i video\_in.mp4 -an -f rawvideo - | x265.exe --input-res <WxH> --fps <int/flo/frac> --input - --output

-format, -an bypass audio, -strict unofficial lift std. restrictions, --y4m stands for "YUV for MPEG", both "-" passes stream through the pipeline

VSpipe.exe [script].vpy --y4m - | x265.exe --y4m --input - --output

VSpipe/avs2yuv [script].vpy - | x265.exe --input-res [WxH] --fps [] --input - --output

avs2yuv.exe [script].avs -raw - | x265.exe --input-res [WxH] --fps [] --input - --output

### ffmpeg built-in scaling:

-sws\_flags bicubic bitexact gauss neighbor bicublin lanczos spline +full\_chroma\_int +full\_chroma\_inp +accurate\_rnd

Example:

-sws\_flags bitexact+full\_chroma\_int+full\_chroma\_inp+accurate\_rnd)

### ffmpeg multiplex all tracks (container format depends on output extension)

* ffmpeg.exe -i ".\v\_in.hevc" -an -c:v copy -i ".\audio\_in.aac" -c:a copy -i ".\subtitle\_in.srt" -c:s copy "mux\_out.mkv"
* ffmpeg.exe -i ".\v\_in.hevc" -an -c:v copy -i ".\audio1.aac" -c:a copy -i ".\aud2.aac" -c:a copy -i ".\sub1.ass" -c:s copy -i ".\sub2.ass" -c:s copy "mux\_out.mkv"

### Subtitle support of different container formats:

[Wikipedia - Subtitle formats support](https://en.wikipedia.org/wiki/Comparison_of_video_container_formats)

### QAAC audio encoding [tutorial](https://www.nazorip.site/archives/44/) or [Github](https://github.com/iAvoe/QAAC-Tutorial-Standalone/blob/master/%E6%95%99%E7%A8%8B.md)

(use webpage translation)

### 

### ffmpeg replace audio track, itoffset±seconds to align:

* ffmpeg.exe -i ".\mux\_in.mov" -i ".\new\_audio.aac" -c:v copy -map 0:v:0 -map 1:a:0 -c:a copy -itsoffset 0 ".\new\_mux\_out.mov"

**ffmpeg: small thread\_queue\_size warning:**

* -thread\_queue\_size<(avg src bitrate kbps+1000)/usable CPU core count>

### Batch: resume CMD prompt on finish:

cmd /k

### + show windows build version:

cmd -k

### x265 HDR settings:

### HDR Tags

**Color**

Primaries

--master-display <manually tagging for instruct video players or decoders to correctly play HDR sources

DCI-P3: G(13250,34500)B(7500,3000)R(34000,16000)WP(15635,16450)L(maxCLL×10000,1)

bt709: G(15000,30000)B(7500,3000)R(32000,16500)WP(15635,16450)L(maxCLL×10000,1)

bt2020: G(8500,39850)B(6550,2300)R(35400,14600)WP(15635,16450)L(maxCLL×10000,1)

* Check HDR source's metadata for color space，then copy the corresponding settings above as param value
* max for L has no standards，which means every video could be different, check your source stream

DCI-P3: G(x0.265, y0.690), B(x0.150, y0.060), R(x0.680, y0.320), WP(x0.3127, y0.329)

bt709: G(x0.30, y0.60), B(x0.150, y0.060), R(x0.640, y0.330), WP(x0.3127,y0.329)

bt2020: G(x0.170, y0.797), B(x0.131, y0.046), R(x0.708, y0.292), WP(x0.3127,y0.329)>

--max-cll <maxCLL,maxFALL>max, average pel intensity. Skip if MediaInfo doesn't get those values out

--colormatrix <as src, e.g.: gbr bt709 fcc bt470bg smpte170m YCgCo bt2020nc bt2020c smpte2084 ictcp>

--transfer <as source, e.g.: gbr bt709 fcc bt470bg smpte170m YCgCo bt2020nc bt2020c smpte2084 ictcp>

**Dolby vision**

: DV-MEL (BL+RPU) & DV-FEL (BL+EL+RPU), x265 support 3 profiles of DV-MEL

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Profile | Codec | BL:EL resolution | x265 supported | Gamma | Space |
| 4 | 10bit hevc | 1:1/4 |  | SDR | YCbCr |
| 5 | BL only (DV-MEL) | √ |  | ICtCp |
| 7 | 4K=1:1/4; 1920x1080=1:1 |  | UHD BluRay | YCbCr |
| 8.1 | BL only (DV-MEL) | √ | HDR10 |
| 8.2 | √ | SDR |
| 8.4 |  | HLG |
| 9 | 8bit avc | BL only (DV-MEL) |  | SDR | YCbCr |

--dolby-vision-profile

<select 5/8.1 (HDR10)/8.2>8.1 needs --master-display & --hdr10-opt

--dolby-vision-rpu

<path>specify path to input RPU binary (.bin)

### Target bit-depth

ffmpeg has yuv-for-mpeg pipe and raw pipe to send video frames to downstream x265 for encoding. The problem is that the raw pipe does not send video frame metadata, and not all x265.exe may read it, causing the loss of bit-depth information, and potentially miss the quality target this tutorial could offer.

For libx265, since output depth is a CLI-ONLY parameter, and ffmpeg is controlling this library, simply setting the -pix-fmt option would do.

The process obtaining pixel bit depth is the same as pix-fmt. See procedures written above: Get the correct -pix\_fmt value.

### x265 pipe input parameter change

x265 v4.0 has introduced Multiview Encoding (multiple video streams from different angles), therefore it changed the pipe specifier from (old way since x264) "-" to "--input -". We are no longer skipping the --input option specifier now.

Note: Encoding speed reference content is temporarily removed due to change in settings

# Gen-Purpose·Simple

Minimal configurable options for simplicity

**splt-trans**

**me-mc**

**adpt quant**

**rate control**

**io**

**bit depth**

**multi node**

**others**

**color space**

--preset slow

--me umh --subme 5 --merange 48 --weightb

--aq-mode 4

--bframes 5 --ref 3

--hash 2 --allow-non-conformance

-D 8/10/12 <Manual specifying required for x265.exe supporting multiple bit depths, default 8, do not cover low-to-high depth, covert high-to-low with --dither>

--pools ,,,, (e.g.: "-,+"states a 2-node computer & use node 2, don't use >1 node per encode)

crop: --display-window < integer "←,↑,→,↓" pixels >, ≥22 core cpu opt.:--pme, interlaced: --field, pixel depth reduction quality+:--dither, begin; ending frame:--seek; --frames, crf/abr resist noise factor:--rc-grain

ffmpeg -pix\_fmt yuv420p / yuv422p / yuv444p / yuv420p10 / yuv422p10 / yuv444p10…

α——(ffmpeg pipe) x265 CLI parameters

* ffmpeg.exe -loglevel 16 -hwaccel auto -y -hide\_banner -i ".\导入.mp4" -an -f yuv4mpegpipe -strict unofficial -pix\_fmt 〇 - | x265.exe -D 〇 --preset slow --me umh --subme 5 --merange 48 --weightb --aq-mode 4 --bframes 5 --ref 3 --hash 2 --allow-non-conformance --y4m --input - --output ".\v\_out.hevc"

β——libx265 CLI, copy audio & multiplex to MP4

* ffmpeg.exe -loglevel 16 -hwaccel auto -y -hide\_banner -i ".\v\_in.mp4" -c:v libx265 -pix\_fmt 〇 -x265-params "preset=slow:me=umh:subme=5:merange=48:weightb=1:bframes=5:ref=3:hash=2:allow-non-conformance=1" -fps\_mode passthrough -c:a copy ".\v\_out.mp4"

### γ——Libkvazaar CLI (in dev, crf mode missing)

* ffmpeg.exe -loglevel 16 -hwaccel auto -y -hide\_banner -i ".\v\_in.mp4" -c:v libkvazaar -pix\_fmt 〇 -kvazaar-params "limit-tu=1:tr-depth-intra=2:pu-depth-intra=4:pu-depth-inter=3:smp=1:amp=1:bipred=1:me=tz:subme=4:merange=48:me-early-termination=off:max-merge=2:ref=3:open-gop=0:period=360:gop=16:transform-skip=1:qp=16:fast-residual-cost=1:early-skip=1:max-merge=5:rd=3:mv-rdo=1:rdoq-skip=1:intra-rdo-et=1:sao=edge:hash=checksum" -fps\_mode passthrough -c:a copy ".\v\_out.mp4"

# Standard.

Lots of custom options for optimizations

--tu-intra-depth 3 --tu-inter-depth 3 --limit-tu 1 --rdpenalty 1 --rect

--me umh --subme <24fps=3, 48fps=4, 60fps=5, 100fps=6> --merange <1920:1080=48, 2560:1440=52, 3840:2160=56> --weightb

--ref 3 --max-merge <2fast, 3, 4slow> --early-skip --no-open-gop --min-keyint 5 --keyint <9×fps> --fades --bframes 8 --b-adapt 2 --radl 3 <sharp source: --pbratio 1.2>

<fast: --fast-intra / slow: --b-intra / very slow & may cause artifacts: --constrained-intra>

--crf <18~20 HQ 19 ~22 HD> --crqpoffs -3 --cbqpoffs -1

--rdoq-level <1fast，2slow>

<anime source: --hevc-aq, remove aq-mode> --aq-mode 4 --aq-strength <flat=0.8, edgy=1>

--rd 3 --limit-modes --limit-refs 1 --rskip <2fast, 1mid, 0slow> --rc-lookahead <3×fps, greater than bframes> --tskip-fast --rect <veryslow: --amp>

--psy-rd <film=1.6，anime=0.6, +0.6 if ctu=64, -0.6 if ctu=16> --splitrd-skip <EXP: --qp-adaptation-range 3>

--limit-sao --sao-non-deblock --deblock 0:-1

--hash 2 --allow-non-conformance <NAS streaming: --idr-recovery-sei>

-D 8/10/12 <Manual specifying required for x265.exe supporting multiple bit depths, default 8, do not cover low-to-high depth, covert high-to-low with --dither>

--pools ,,,, (e.g.: "-,+"states a 2-node computer & use node 2, don't use >1 node per encode)

ffmpeg -pix\_fmt yuv420p / yuv422p / yuv444p / yuv420p10 / yuv422p10 / yuv444p10…

**splt-trans**

**me-mc**

**ref-rateol**

**intra coding**

**quantization**

**rdoq**

**adapt quant**

**md decision**

**rdo**

**deblock-sao**

**input output**

**bit depth**

**multi node**

**others**

α——(ffmpeg pipe) x265 CLI parameters

* ffmpeg.exe -loglevel 16 -hwaccel auto -y -hide\_banner -i ".\v\_in.mp4" -an -f yuv4mpegpipe -strict unofficial -pix\_fmt 〇 - | x265.exe -D 〇 --ctu 〇 --min-cu-size 16 --tu-intra-depth 3 --tu-inter-depth 3 --limit-tu 1 --rdpenalty 1 --me umh --subme 〇 --merange 〇 --weightb --ref 3 --max-merge 〇 --early-skip --no-open-gop --min-keyint 5 --fades --bframes 8 --b-adapt 2 --radl 3 --pbratio 1.2 --fast-intra --b-intra --crf 〇 --crqpoffs -3 --cbqpoffs -1 --rdoq-level 〇 --aq-mode 4 --aq-strength 〇 --rd 3 --limit-modes --limit-refs 1 --rskip 〇 --rc-lookahead 〇 --tskip-fast --rect --amp --psy-rd 〇 --splitrd-skip --qp-adaptation-range 4 --limit-sao --sao-non-deblock --deblock 0:-1 --hash 2 --allow-non-conformance --y4m --input - --output ".\v\_out.hevc"

β——libx265 CLI, copy audio & multiplex to MP4

* ffmpeg.exe -loglevel 16 -hwaccel auto -y -hide\_banner -i ".\v\_in.mp4" -c:v libx265 -pix\_fmt 〇 -x265-params "ctu=〇:min-cu-size=16:tu-intra-depth=3:tu-inter-depth=3:limit-tu=1:rdpenalty=1:me=umh:subme=〇:merange=〇:weightb=1:ref=3:max-merge=〇:early-skip=1:open-gop=0:min-keyint=5:fades=1:bframes=8:b-adapt=2:radl=3:pbratio=1.2:fast-intra=1:b-intra=1:crf=〇:crqpoffs=-3:cbqpoffs=-1:rdoq-level=〇:aq-mode=4:aq-strength=〇:rd=3:limit-modes=1:limit-refs=1:rskip=〇:rc-lookahead=〇:tskip-fast=1:rect=1:amp=1:psy-rd=〇:splitrd-skip=1:qp-adaptation-range=4:limit-sao=1:sao-non-deblock=1:deblock=0,-1:hash=2:allow-non-conformance=1" -fps\_mode passthrough -c:a copy ".\v\_out.mp4"

High Compression·Film

--ctu 64 --tu-intra-depth 4 --tu-inter-depth 4 --limit-tu 1

--me star --subme <24fps=3, 48fps=4, 60fps=5, 100fps=6> --merange 56 --weightb

--ref 3 --max-merge 5 --no-open-gop --min-keyint 3 --keyint <9×fps> --fades --bframes 8 --b-adapt 2 --radl 3 --analyze-src-pics

--b-intra < very slow & may cause artifacts: --constrained-intra>

--crf 21.8 --crqpoffs -3 --ipratio 1.2 --pbratio 1.5

--rdoq-level 2

--aq-mode 4 --aq-strength <clean source=0.8，film=1> --qg-size 8

--rd 5 --limit-refs 0 --rskip 0 --rc-lookahead <1.8×fps, greater than bframes> --rect --amp

--psy-rd <film=1.6, animation=0.6, +0.6 if ctu=64, -0.6 if ctu=16> <EXP: --qp-adaptation-range 3>

--deblock 0:0

--limit-sao --sao-non-deblock --selective-sao 3

--hash 2 --allow-non-conformance --nr-inter 8 <NAS streaming: --idr-recovery-sei>

-D 8/10/12 <Manual specifying required for x265.exe supporting multiple bit depths, default 8, do not cover low-to-high depth, covert high-to-low with --dither>

--pools ,,,, (e.g.: "-,+"states a 2-node computer & use node 2, don't use >1 node per encode)

ffmpeg -pix\_fmt yuv420p / yuv422p / yuv444p / yuv420p10 / yuv422p10 / yuv444p10…

**splt-trans**

**me-mc**

**ref-rateol**

**intra coding**

**quantization**

**rdoq**

**adapt.quant**

**md decision**

**rdo**

**deblock**

**sao**

**io**

**bit depth**

**multi node**

**color space**

α——(ffmpeg pipe) x265 CLI parameters

* ffmpeg.exe -loglevel 16 -hwaccel auto -y -hide\_banner -i ".\v\_in.mp4" -an -f yuv4mpegpipe -strict unofficial -pix\_fmt 〇 - | x265.exe -D 〇 --ctu 64 --tu-intra-depth 4 --tu-inter-depth 4 --limit-tu 1 --me star --subme 〇 --merange 〇 --weightb --ref 3 --max-merge 5 --no-open-gop --min-keyint 3 --keyint 〇 --fades --bframes 8 --b-adapt 2 --radl 3 --analyze-src-pics --b-intra --crf 21.8 --crqpoffs -3 --ipratio 1.2 --pbratio 1.5 --rdoq-level 2 --aq-mode 4 --aq-strength 〇 --qg-size 8 --rd 5 --limit-refs 0 --rskip 0 --rc-lookahead 〇 --rect --amp --psy-rd 〇 --qp-adaptation-range 3 --deblock 0:-1 --limit-sao --sao-non-deblock --selective-sao 3 --hash 2 --allow-non-conformance --y4m --input - --output ".\v\_out.hevc"

β——libx265 CLI, copy audio & multiplex to MP4

* ffmpeg.exe -loglevel 16 -hwaccel auto -y -hide\_banner -i ".\v\_in.mp4" -c:v libx265 -pix\_fmt 〇 -x265-params "ctu=64:tu-intra-depth=4:tu-inter-depth=4:limit-tu=1:me=star:subme=〇:merange=〇:weightb=1:ref=3:max-merge=5:open-gop=0:min-keyint=3:keyint=〇:fades=1:bframes=8:b-adapt=2:radl=3:analyze-src-pics=1:b-intra=1:crf=21.8:crqpoffs=-3:ipratio=1.2:pbratio=1.5:rdoq-level=2:aq-mode=4:aq-strength=〇:qg-size=8:rd=5:limit-refs=0:rskip=0:rc-lookahead=〇:rect=1:amp=1:psy-rd=〇:qp-adaptation-range=3:deblock=0,-1:limit-sao=1:sao-non-deblock=1:selective-sao=3:hash=2:allow-non-conformance=1" -fps\_mode passthrough -c:a copy ".\v\_out.mp4"

Stock Footage·Render & Reuse

--ctu 32

--me star --subme <24fps=3, 48fps=4, 60fps=5, 100fps=6> --merange <1920:1080=48, 2560:1440=52, 3840:2160=56> --analyze-src-pics

--max-merge 5 --early-skip --b-intra

--no-open-gop --min-keyint 1 --keyint <7×fps>--ref 3 --fades --bframes 7 --b-adapt 2

--crf 17 --crqpoffs -3 --cbqpoffs -2

--rd 3 --limit-modes --limit-refs 1 --rskip 1 --rc-lookahead <4×fps, , greater than bframes>

--splitrd-skip

--deblock -1:-1

--hash 2 --allow-non-conformance

--tune grain

-D 8/10/12 <Manual specifying required for x265.exe supporting multiple bit depths, default 8, do not cover low-to-high depth, covert high-to-low with --dither>

ffmpeg -pix\_fmt yuv420p / yuv422p / yuv444p / yuv420p10 / yuv422p10 / yuv444p10…

**block/unit spitting**

**motion est.&cmp**

**intraframe search**

**rate control**

**quantization**

**mode decision**

**R-D optimization**

**deblock**

**input output**

**tuning**

**bit depth**

**color space**

α——(ffmpeg pipe) x265 CLI parameters

* ffmpeg.exe -loglevel 16 -hwaccel auto -y -hide\_banner -i ".\v\_in.mp4" -an -f yuv4mpegpipe -strict unofficial -pix\_fmt 〇 - | x265.exe -D 〇 --ctu 32 --me star --subme 〇 --merange 〇 --analyze-src-pics --max-merge 5 --early-skip --b-intra --no-open-gop --min-keyint 1 --keyint 〇 --ref 3 --fades --bframes 7 --b-adapt 2 --crf 17 --crqpoffs -3 --cbqpoffs -2 --rd 3 --limit-modes --limit-refs 1 --rskip 1 --rc-lookahead 〇 --splitrd-skip --deblock -1:-1 --hash 2 --allow-non-conformance --tune grain --y4m --input - --output ".\v\_out.hevc"

β——libx265 CLI, copy audio & multiplex to MP4

* ffmpeg.exe -loglevel 16 -hwaccel auto -y -hide\_banner -i ".\v\_in.mp4" -c:v libx265 -pix\_fmt 〇 -x265-params "ctu=32:me=star:subme=〇:merange=〇:analyze-src-pics=1:max-merge=5:early-skip=1:open-gop=0:min-keyint=1:keyint=〇:ref=3:fades=1:bframes=7:b-adapt=2:radl=3:b-intra=1:crf=17:crqpoffs =-3:cbqpoffs=-2:rd=3:limit-modes=1:limit-refs=1:rskip=1:rc-lookahead=〇:splitrd-skip=1:deblock=-1,-1:hash=2:allow-non-conformance=1:tune=grain" -fps\_mode passthrough -c:a copy ".\v\_out.mp4"

Anime·High Compression·Subtitle Groups

--tu-intra-depth 4 --tu-inter-depth 4 --max-tu-size 16

--me umh --subme <24fps=3, 48fps=4, 60fps=5, 100fps=6> --merange <1920:1080=48, 2560:1440=52, 3840:2160=56> --weightb --max-merge 5 --early-skip

--ref 3 --no-open-gop --min-keyint 5 --keyint <12×fps> --fades --bframes 16 --b-adapt 2 --radl 3 --bframe-bias 20

--b-intra <very slow & may cause artifacts: --constrained-intra>

--crf 22 --crqpoffs -4 --cbqpoffs -2 --ipratio 1.6 --pbratio 1.3 --cu-lossless --tskip

--psy-rdoq 2.3 --rdoq-level 2

--hevc-aq --aq-strength 0.9 --qg-size 8

--rd 3 --limit-modes --limit-refs 1 --rskip 1 --rc-lookahead <2.5×fps, greater than bframes> --rect --amp

--psy-rd 1.5 --splitrd-skip --rdpenalty 2 <EXP: --qp-adaptation-range 4>

--deblock 0:-1 --limit-sao --sao-non-deblock

--hash 2 --allow-non-conformance --single-sei <NAS streaming: --idr-recovery-sei>

-D 8/10/12 <Manual specifying required for x265.exe supporting multiple bit depths, default 8, do not cover low-to-high depth, covert high-to-low with --dither>

--pools ,,,, (e.g.: "-,+"states a 2-node computer & use node 2, don't use >1 node per encode)

ffmpeg -pix\_fmt yuv420p / yuv422p / yuv444p / yuv420p10 / yuv422p10 / yuv444p10…

**splt-trans**

**me-mc**

**ref-rateol**

**intra coding**

**quantization**

**rdoq**

**aq**

**md**

**rdo**

**deblock -sao**

**input output**

**bit depth**

**multi nodes**

**color space**

α——(ffmpeg pipe) x265 CLI parameters

* ffmpeg.exe -loglevel 16 -hwaccel auto -y -hide\_banner -i ".\v\_in.mp4" -an -f yuv4mpegpipe -strict unofficial -pix\_fmt 〇 - | x265.exe -D 〇 --tu-intra-depth 4 --tu-inter-depth 4 --max-tu-size 16 --me umh --subme 〇 --merange 〇 --weightb --max-merge 5 --early-skip --ref 3 --no-open-gop --min-keyint 5 --keyint 〇 --fades --bframes 16 --b-adapt 2 --radl 3 --bframe-bias 20 --b-intra --crf 22 --crqpoffs -4 --cbqpoffs -2 --ipratio 1.6 --pbratio 1.3 --cu-lossless --tskip --psy-rdoq 2.3 --rdoq-level 2 --hevc-aq --aq-strength 0.9 --qg-size 8 --rd 3 --limit-modes --limit-refs 1 --rskip 1 --rc-lookahead 〇 --rect --amp --psy-rd 1.5 --splitrd-skip --rdpenalty 2 --qp-adaptation-range 4 --deblock -1:0 --limit-sao --sao-non-deblock --hash 2 --allow-non-conformance --y4m --input - --output ".\v\_out.hevc"

β——libx265 CLI, copy audio & multiplex to MP4

* ffmpeg.exe -loglevel 16 -hwaccel auto -y -hide\_banner -i ".\v\_in.mp4" -c:v libx265 -pix\_fmt 〇 -x265-params "tu-intra-depth=4:tu-inter-depth=4:max-tu-size=16:me=umh:subme=〇:merange=〇:weightb=1:max-merge=5:early-skip=1:ref=3:open-gop=0:min-keyint=5:keyint=〇:fades=1:bframes=16:b-adapt=2:radl=3:bframe-bias=20:b-intra=1:crf=22:crqpoffs=-4:cbqpoffs=-2:ipratio=1.6:pbratio=1.3:cu-lossless=1:tskip=1:psy-rdoq=2.3:rdoq-level=2:hevc-aq=1:aq-strength=0.9:qg-size=8:rd=3:limit-modes=1:limit-refs=1:rskip=1:rc-lookahead=〇:rect=1:amp=1:psy-rd=1.5:splitrd-skip=1:rdpenalty=2:qp-adaptation-range=4:deblock=-1,0:limit-sao=1:sao-non-deblock=1:hash=2:allow-non-conformance=1" -fps\_mode passthrough -c:a copy ".\v\_out.mp4"

Anime·HEDT+HQ Src Only

Slower, less efficient and larger files compared to Anime-Subtitle-Groups’ case, keeping over sharing

--tu-intra-depth 4 --tu-inter-depth 4 --max-tu-size 4 --limit-tu 1 --rect --amp

--me star --subme <24fps=3, 48fps=4, 60fps=5, 100fps=6> --merange <1920:1080=52, 2560:1440=56, 3840:2160=64> --analyze-src-pics --weightb --max-merge 5

--ref 3 --no-open-gop --min-keyint 1 --keyint <12×fps> --fades --bframes 16 --b-adapt 2 --radl 2

--b-intra

--crf 17.1 --crqpoffs -5 --cbqpoffs -2 --ipratio 1.67 --pbratio 1.33 --cu-lossless

--psy-rdoq 2.5 --rdoq-level 2

<Normal: --hevc-aq --aq-strength 1.4; Jpsdr mod: --aq-auto 10 --aq-bias-strength 1.3 --aq-strength-edge 1.4 --aq-bias-strength 1.1> --qg-size 8

--rd 5 --limit-refs 0 --rskip 2 --rskip-edge-threshold 3 --rc-lookahead <2.5×fps, greater than bframes> --no-cutree

--psy-rd 1.5 --rdpenalty 2 <EXP: --qp-adaptation-range 5>

--deblock -2:-2

--limit-sao --sao-non-deblock --selective-sao 1

--hash 2 --allow-non-conformance --single-sei <NAS streaming: --idr-recovery-sei>

-D 8/10/12 <Manual specifying required for x265.exe supporting multiple bit depths, default 8, do not cover low-to-high depth, covert high-to-low with --dither>

**splt-trans**

**me-mc**

**ref-rateol**

**intra coding**

**quantization**

**rdoq**

**aq**

**md**

**rdo**

**deblock**

**sao**

**input output**

**bit depth**

α——(ffmpeg pipe) std. x265 CLI parameters

* ffmpeg.exe -loglevel 16 -hwaccel auto -y -hide\_banner -i ".\v\_in.mp4" -an -f yuv4mpegpipe -strict unofficial -pix\_fmt 〇 - | x265.exe -D 〇 --tu-intra-depth 4 --tu-inter-depth 4 --max-tu-size 4 --limit-tu 1 --rect --amp --me star --subme 〇 --merange 〇 --analyze-src-pics --weightb --max-merge 5 --ref 3 --no-open-gop --min-keyint 1 --keyint 〇 --fades --bframes 16 --b-adapt 2 --radl 2 --b-intra --crf 17.1 --crqpoffs -5 --cbqpoffs -2 --ipratio 1.67 --pbratio 1.33 --cu-lossless --psy-rdoq 2.5 --rdoq-level 2 --hevc-aq --aq-strength 1.4 --qg-size 8 --rd 5 --limit-refs 0 --rskip 2 --rskip-edge-threshold 3 --rc-lookahead 〇--no-cutree --psy-rd 1.5 --rdpenalty 2 --qp-adaptation-range 5 --deblock -2:-2 --limit-sao --sao-non-deblock --selective-sao 1 --hash 2 --allow-non-conformance --y4m --input - --output ".\v\_out.hevc"

β——(ffmpeg pipe) jpsdr mod x265 CLI parameters

* ffmpeg.exe -loglevel 16 -hwaccel auto -y -hide\_banner -i ".\v\_in.mp4" -an -f yuv4mpegpipe -strict unofficial -pix\_fmt 〇 - | x265.exe -D 〇 --tu-intra-depth 4 --tu-inter-depth 4 --max-tu-size 4 --limit-tu 1 --rect --amp --me star --subme 〇 --merange 〇 --analyze-src-pics --weightb --max-merge 5 --ref 3 --no-open-gop --min-keyint 1 --keyint 〇 --fades --bframes 16 --b-adapt 2 --radl 2 --b-intra --crf 17.1 --crqpoffs -5 --cbqpoffs -2 --ipratio 1.67 --pbratio 1.33 --cu-lossless --psy-rdoq 2.5 --rdoq-level 2 --aq-auto 10 --aq-bias-strength 1.3 --aq-strength-edge 1.4 --aq-bias-strength 1.1 --qg-size 8 --rd 5 --limit-refs 0 --rskip 2 --rskip-edge-threshold 3 --rc-lookahead 〇--no-cutree --psy-rd 1.5 --rdpenalty 2 --qp-adaptation-range 5 --deblock -2:-2 --limit-sao --sao-non-deblock --selective-sao 1 --hash 2 --allow-non-conformance --y4m --input - --output ".\v\_out.hevc"

γ——Standard libx265 CLI, copy audio & multiplex to MP4

* ffmpeg.exe -loglevel 16 -hwaccel auto -y -hide\_banner -i ".\v\_in.mp4" -c:v libx265 -pix\_fmt 〇 -x265-params "tu-intra-depth=4:tu-inter-depth=4:max-tu-size=4:limit-tu=1:rect=1:amp=1:me=star:subme=〇:merange=64:analyze-src-pics=1:weightb=1:max-merge=5:mcstf=1:ref=3:open-gop=0:min-keyint=1:keyint=〇:fades=1:bframes=16:b-adapt=2:radl=2:b-intra=1:crf=17.1:crqpoffs=-5:cbqpoffs=-2:ipratio=1.6:pbratio=1.33:cu-lossless=1:psy-rdoq=2.5:rdoq-level=2:hevc-aq=1:aq-strength=1.4:qg-size=8:rd=5:limit-refs=0:rskip=2:rskip-edge-threshold=3:rc-lookahead=〇:cutree=0:psy-rd=1.5:rdpenalty=2:qp-adaptation-range=5:deblock=-2:-2:limit-sao=1:sao-non-deblock=1:selective-sao=1:hash=2:allow-non-conformance=1" -fps\_mode passthrough -c:a copy ".\v\_out.mp4"