What is Tuning?

- Disable features that:
  - Improve subjective video quality but
  - Degrade objective scores
- Example: adaptive quantization – changes bit allocation over frame depending upon complexity
  - Improves visual quality
  - Looks like “error” to metrics like PSNR/VMAF

What is Tuning?

- Switches in encoding string that enables tuning (and disables these features)

```
ffmpeg -input.mp4 -c:v libx264 -tune psnr output.mp4
```

- With x264, this disables adaptive quantization and psychovisual optimizations

Why So Important

- Major point of contention:
  - “If you’re running a test with x264 or x265, and you wish to publish PSNR or SSIM scores, you MUST use –tune PSNR or –tune SSIM, or your results will be completely invalid.”
  - http://x265.org/compare-video-encoders/
- Absolutely critical when comparing codecs because some may or may not enable these adjustments
- You don’t have to tune in your tests; but you should address the issue and explain why you either did or didn’t

Does Impact Scores

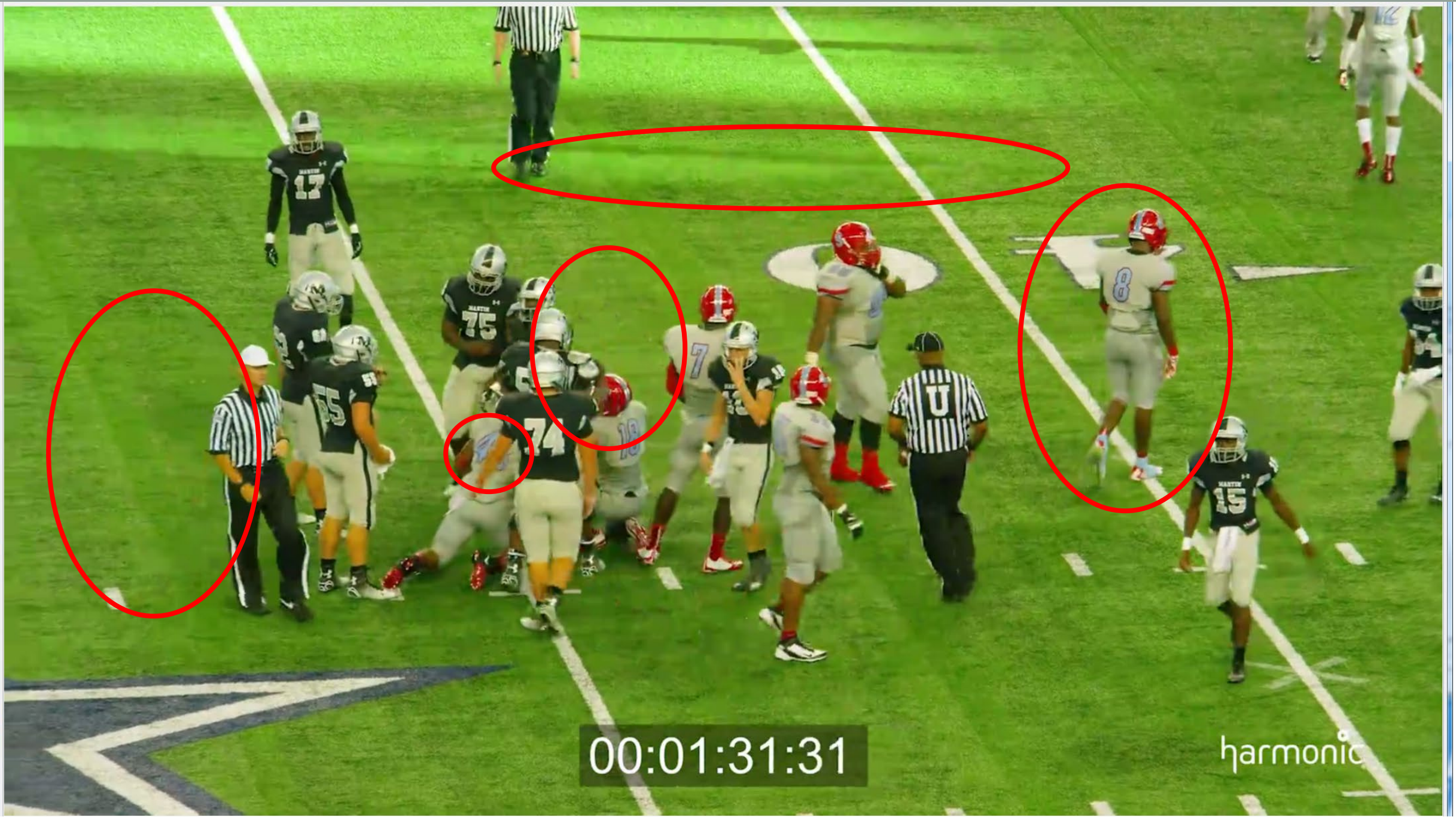
- 3 mbps football (high motion, lots of detail)
- PSNR
  - No tuning – 32.00 dB
  - Tuning – 32.58 dB
  - .58 dB
- VMAF
  - No tuning – 71.79
  - Tuning – 75.01
  - Difference – over 3 VMAF points
    - 6 is JND, so not a huge deal
    - But if inconsistent between test parameters, could incorrectly show one codec (or encoding configuration) as better than the other

VQMT VMAF Graph

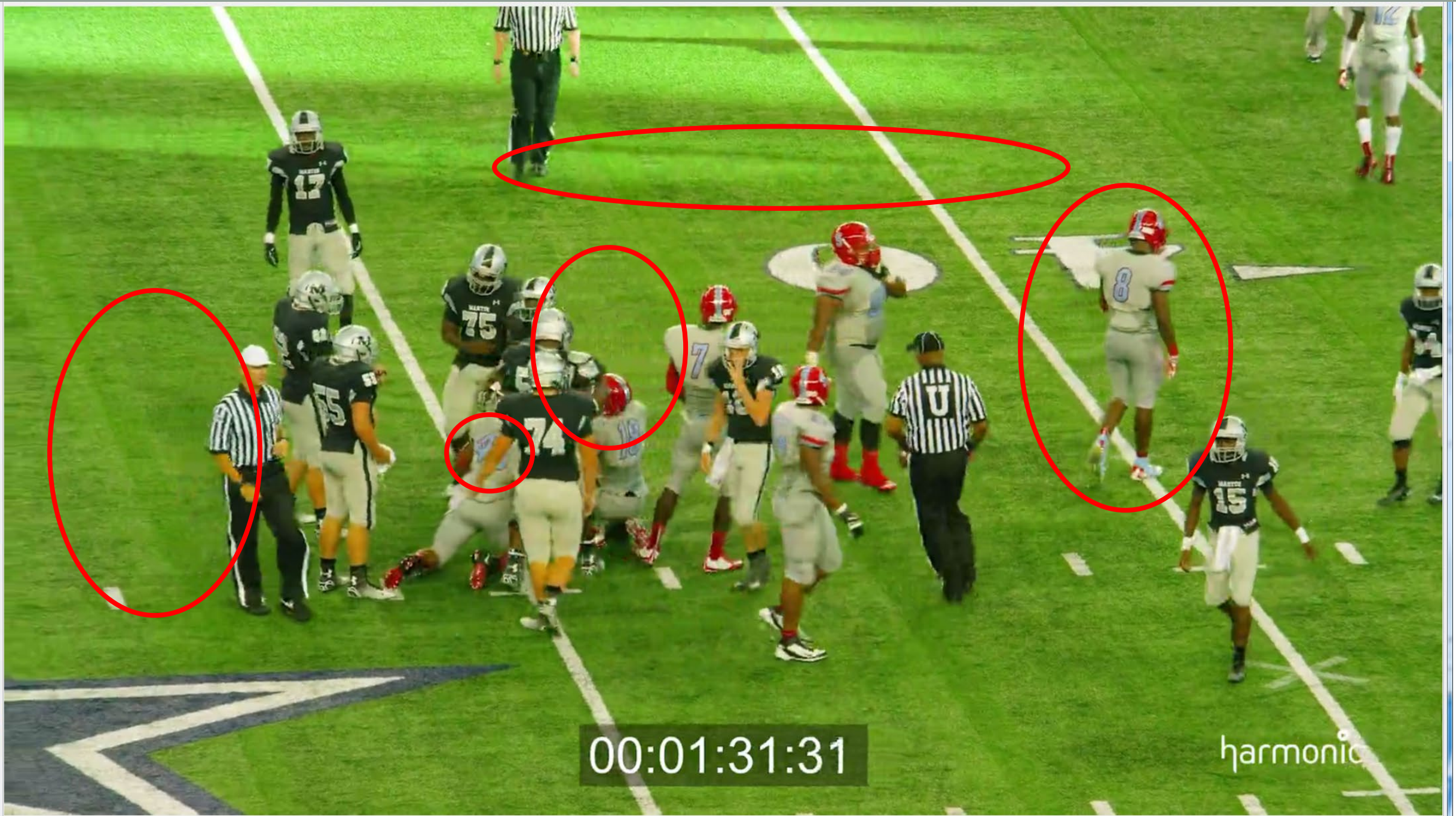
Red – tuned
Green – not tuned

Multiple frames with
3-4-point differentials

Tuned



Not tuned



Observations

- Tuning
  - Produces more blurry areas
  - Reduces detail
  - Reduces artifacts
- Without tuning
  - More detail
  - Slightly more artifacts
  - Looks more accurate and “better” to my eye

- Key point:
  - When comparing encoders and codecs with visual quality metrics, **be consistent**
    - If tuning for one, tune for all
- When comparing encoding parameters with the same codec, not so critical
  - Tuning or not tuning should have the same efffect

Most Academic Comparisons Tend to Tune

- Coding efficiency comparison of AV1/VP9, H.265/MPEG-HEVC, and H.264/MPEG-AVC encoders
  - bit.ly/Grois_AV1

TABLE I. SELECTED SETTINGS FOR THE AOM/AV1 ENCODER

CODEC	AOM/AV1
Version	AOMedia Project AV1 Encoder, Version: b6724815f22876ca88f43b57dba09a555ef4e1b0
Recommended settings	--best --psnr --tune=psnr --end-usage=q --passes=2 --tile-columns=0 --arnr-strength=5 --min-q=$QP --max-q=$QP --cq-level=$QP

TABLE II. SELECTED SETTINGS FOR THE X265 ENCODER

CODEC	x265
Version	VideoLAN Project x265 Encoder, Version: 2.0
Recommended settings	--profile=main –p=placebo --psnr --tune=psnr --pools none --no-pmode --no-pme --no-allow-non-conformance --rd=6 --rect --amp –qp=$QP --keyint=$IntraPeriod --min-keyint=$IntraPeriod --pass=2

Moscow State University

- MSU Codec Comparison 2018
  - bit.ly/MSU_HEVC_18
  - Tuned whenever possible

Universal Encoding	x264 --preset slower --me hex --keyint infinite --tune ssim --pass 1 --bitrate %BITRATE_KBPS% %SOURCE_FILE% --input-res %WIDTH%x%HEIGHT% --fps %FPS% -o NUL x264 --preset slower --me hex --keyint infinite --tune ssim --pass 2 --bitrate %BITRATE_KBPS% %SOURCE_FILE% --input-res %WIDTH%x%HEIGHT% --fps %FPS% -o %TARGET_FILE%

Universal Encoding	x265.exe --input %SOURCE_FILE% --input-res %WIDTH%x%HEIGHT% --fps %FPS% -p medium --bitrate %BITRATE_KBPS% --psnr --ssim --tune=ssim -o %TARGET_FILE% --bframes 4 --max-merge 3 --ref 3 --b-intra --limit-ref 1 --early-skip

Practitioners Are Mixed

- Facebook
  - AV1 beats x264 and libvpx-vp9 in practical use cases
    - bit.ly/FB_AV1_VP9
  - Two encoding cases, ***neither tuned***

Codec	CRF/QP mode	ABR mode
AV1	<INPUT> --i420 -y --codec=av1 --cpu-used=1 --threads=0 --profile=0 --lag-in-frames=19 --min-q=0 --max-q=63 --auto-alt-ref=1 --kf-max-dist=60 --kf-min-dist=60 --drop-frame=0 --static-thresh=0 --bias-pct=50 --minsection-pct=0 --maxsection-pct=2000 --arnr-maxframes=7 --arnr-strength=5 --sharpness=0 --undershoot-pct=100 --overshoot-pct=100 --tile-columns=0 --frame-parallel=0 --test-decode=warn -v --end-usage=q --cq-level=<CRF> --webm -o <OUTPUT>	<INPUT> --i420 -y --codec=av1 --cpu-used=1 --threads=0 --profile=0 --lag-in-frames=19 --min-q=0 --max-q=63 --auto-alt-ref=1 --passes=<PASS> --kf-max-dist=60 --kf-min-dist=60 --drop-frame=0 --static-thresh=0 --bias-pct=50 --minsection-pct=0 --maxsection-pct=2000 --arnr-maxframes=7 --arnr-strength=5 --sharpness=0 --undershoot-pct=100 --overshoot-pct=100 --tile-columns=0 --frame-parallel=0 --test-decode=warn -v --end-usage=vbr --target-bitrate=<BITRATE> --webm -o <OUTPUT>
x264 Main Profile	-i <INPUT> -c:v libx264 -pix_fmt yuv420p -profile:v main -preset veryslow -crf <CRF> -refs 5 -g 60 -keyint_min 60 -sc_threshold 0 -f mp4 <OUTPUT>	-i <INPUT> -c:v libx264 -pix_fmt yuv420p -profile:v main -preset veryslow -b:v <BITRATE> -refs 5 -g 60 -keyint_min 60 -sc_threshold 0 -pass <PASS> -f mp4 <OUTPUT>
x264 High Profile	-i <INPUT> -c:v libx264 -pix_fmt yuv420p -profile:v high -preset veryslow -crf <CRF> -refs 5 -g 60 -keyint_min 60 -sc_threshold 0 -f mp4 <OUTPUT>	-i <INPUT> -c:v libx264 -pix_fmt yuv420p -profile:v high -preset veryslow -b:v <BITRATE> -refs 5 -g 60 -keyint_min 60 -sc_threshold 0 -pass <PASS> -f mp4 <OUTPUT>
libvpx-vp9	-i <INPUT> -c:v libvpx-vp9 -pix_fmt yuv420p -crf <CRF> -b:v 0 -speed 1 -tile-columns 0 -frame-parallel 0 -auto-alt-ref 1 -lag-in-frames 25 -keyint_min 60 -g 60 -f webm <OUTPUT>	-i <INPUT> -c:v libvpx-vp9 -pix_fmt yuv420p -b:v <BITRATE> -speed 1 -tile-columns 0 -frame-parallel 0 -auto-alt-ref 1 -lag-in-frames 25 -keyint_min 60 -g 60 -pass <PASS> -f webm <OUTPUT>

Practitioners Are Mixed

- Netflix – Doesn't Tune
  - Standardization bodies tend to use test conditions that let them compare one tool to another, often maximizing a particular objective metric and reducing variability over different experiments. For example, rate-control and visual tunings are generally disabled, to focus on the effectiveness of core coding tools.
  - Netflix encoding recipes focus on achieving the best quality, enabling the available encoder tools that boost visual appearance, and thus, giving less weight to indicators like speed or encoder footprint that are crucial in other applications.
  - bit.ly/NF_codecs

Netflix on Tuning

- Best Practices for Netflix's VMAF Metric
  - bit.ly/VMAF_bestp
- On tuning for VMAF
  - "Since VMAF partially captures the benefit of perceptual optimization, and if at the end of the day you will be encoding with these settings on, we still recommend turning them on."

General Rules

- When VQ metrics accurately mimic human perception, there will be no need to tune
- Until then:
  - Be consistent – either tune for all or don't tune for any
- If testing for publication, detail what you did and why
  - This decision will make or break public perception of your work
    - Detail what you did and why
- If producing for inhouse use:
  - Test using actual production parameters unless this introduces an obvious bias

Implementing Tuning

- Tuning varies by codec
  - x264/x265 – can tune for PSNR/SSIM
  - Intel SVT-AV1 – can tune for PSNR/VMAF/Visual quality
  - NGCodec (others) – Must manually disable adaptive quantization
- Before getting started:
  - Check codec documentation
  - Spend an hour checking other published comparisons to see what they did