BEAMR

TECH GUIDE

It's Time To Modernize Your Encoding Profiles, But How?

CONTENT ADAPTIVE TECH GUIDE

Beamr Imaging Ltd July 2016

*This content originally appeared in the 2016 Streaming Media Advanced Encoding and Formats Superguide.

If you've ever compressed a video file, you know that every video is different, and that the only way to achieve the best picture quality at the lowest bitrate is by customizing the encoding parameters for each file, which often requires trial and error to achieve the optimum result. Still, since the dawn of adaptive streaming, and Apple's Tech Note TN2224, almost all video producers have applied a single encoding ladder to all videos, irrespective of the content type.

This started to change when Netflix published their Per-Title Encode Optimization blog post (http://snip.ly/hh2Iq), which described how Netflix formulates the optimal encoding "ladder" for each video. This scheme provides Netflix not only bandwidth and storage savings, but also ensures that the highest video quality is delivered. Not surprisingly, YouTube has its own approach for customizing encoding parameters for the near- countless videos uploaded to the platform (http://snip.ly/0f53i).

While these innovations are driven by behemoths with armies of engineers and teams of PhD's, all publishers of premium video—large and small—should pursue the same cost and quality of service benefits that breaking from fixed recipe encoding can bring. This article reviews a number of approaches, including some that can be simply and inexpensively implemented by any video distributor.

First, let's address the reality that in addition to technical considerations, your monetization strategy will also impact your encoding configuration. Perhaps your marketing department says you must encode at 1080p to "be competitive," but have you ever wondered why many streaming services deliver at 720p? The reason is simple. At 720p, many devices will scale to fill the screen quite nicely, and with less than half the pixels of 1080p, encoders can deliver good quality while keeping the bitrate low enough to stream reliably.

We aren't suggesting that reducing resolution is a panacea, but that the secret to optimizing video quality while retaining streaming UX is to appropriately match bitrate and resolution, and in this regard, content adaptive techniques and tools can help. This article will describe multiple tools and techniques to achieve the highest quality possible at the lowest bitrate. First, let's define a few terms and technologies.

Encoding ladder—An encoding ladder is a set of encoding configurations used to encode a single video title (VOD or live) for adaptive streaming at various resolutions and bitrates. In general, encoding ladders are formulated to deliver an optimal range of viewing experiences to those connecting with different devices over varying bandwidths.

Constant Rate Factor (CRF) Encoding—CRF encoding is available with x264 and x265 open-source implementations of H.264 and HEVC, and with VP9. When encoding with CRF, you set a desired quality level for the file, not a data rate. While encoding, the encoder tries to maintain the desired quality, and drops data rate and video quality during motion frames since the eye perceives less loss of detail in a moving frame than a static frame. Using CRF, the data rate for the file is not known in advance, and may be relatively high for complex scenes. In addition, the same CRF value can yield different levels of perceptual quality for different content, such as animation vs. natural video. As you'll see, CRF is more often used to gauge the encoding complexity of a file than to encode files for streaming delivery.

Objective Quality Metrics—Objective quality benchmarks are mathematical algorithms that compare the compressed video with the source and render a value that predicts how viewers would rate the compressed file. There are multiple algorithms, including the Peak Signal-to-ise Ratio (PSNR), Structural Similarity Index (SSIM), and Netflix's new Video Multimethod Assessment Fusion (VMAF).

PER-TITLE ENCODING NETFLIX STYLE

Netflix's per-title encoding schema is a fully automated technique that creates a custom encoding ladder for each video file. At the heart of the analysis is a series of test encodes performed at different CRF levels and resolutions and then analyzed using the Video Multimethod Assessment Fusion (VMAF) quality metric, though Netflix used PSNR initially. Netflix uses these scores to identify the best quality resolution at each relevant data rate, producing the Convex Hull shown in **Figure 1**.

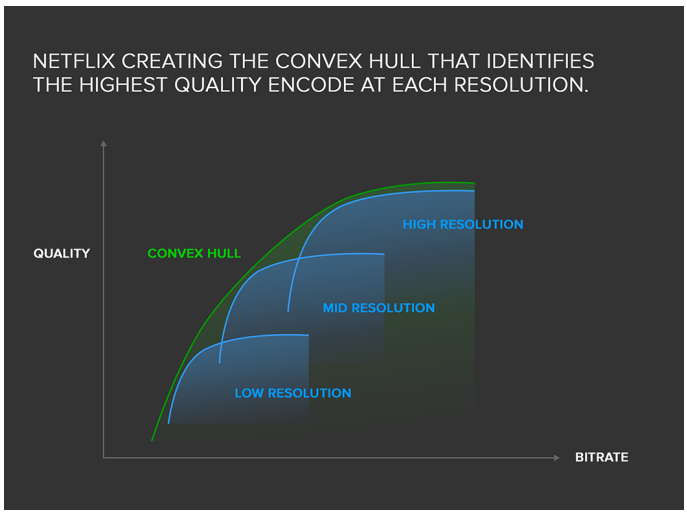

Figure 1. *Convex Hulls identify the highest quality encoding data rate at each resolution.*

Netflix encodes videos using a fixed set of encoding resolutions to ensure backwards compatibility with all supported devices. To save encoding cycles, the company doesn't run exhaustive tests at each resolution; rather, it runs a series of trial encodes over a finite set of resolutions, and then interpolates those results over the full encoding ladder. Netflix performs the initial analysis for 1080p and larger displays. To ensure optimal quality for older devices with limited display resolution, Netflix performs separate analysis for devices with maximum viewing resolutions of 480p and 720p, so each video has three encoding ladders, one for 480p, one for 720p, and one for 1080p and beyond. Netflix uses this schema to identify the optimal data rate, but performs the production encode using 2-pass VBR, not CRF.

YOUTUBE AND THE NEURAL NETWORK

Netflix's approach involves multiple trial encodes, which works well when you distribute a large, but limited set of content. YouTube on the other hand, ingests more than 400 hours of video per minute, so they require a more efficient technique (http://snip.ly/ak27u).

Complicating matters is the encoding pipeline YouTube deploys, which splits each source file into chunks, and encodes them in parallel using different encoding instances. Since communications between these instances would complicate system design and operation, YouTube designed a solution that didn't involve interinstance communications. Because YouTube deploys multiple codecs, the solution also had to be codec agnostic, which meant that it had to be driven by a single rate control parameter for each codec. For x264, YouTube uses the now familiar CRF value.

For each chunk, YouTube needs to identify the CRF value that would produce the optimal quality at the most efficient data rate, without any communication between encoding instances. YouTube couldn't run a full quality CRF pass on the video; it had to work with information gleaned from a mezzanine transcode performed on all uploaded clips before encoding. This data includes information like input bitrate, motion vector bits, resolution and frame rate.

To convert this clip-specific data into a CRF value for each chunk, YouTube deployed a neural network where Google ran a brute force computation that encoded 10,000 video clips at every CRF setting to find the ideal CRF value for that clip. Then they input all known data points for each clip, and told the system to find a correlation between the data and the ideal CRF value. YouTube learned that with this data, the system could choose the correct CRF value 65% of the time. YouTube then added in the data derived from a fast, low resolution CRF encode of the clip, which boosted accuracy to 80%.

Results are shown in **Figure 2**, where the blue line illustrates best fit as determined by a brute force calculation. The red line is the original result from the neural network, while the orange line illustrates the final result using the low resolution CRF encode. The green line represents the baseline using a preset CRF value.

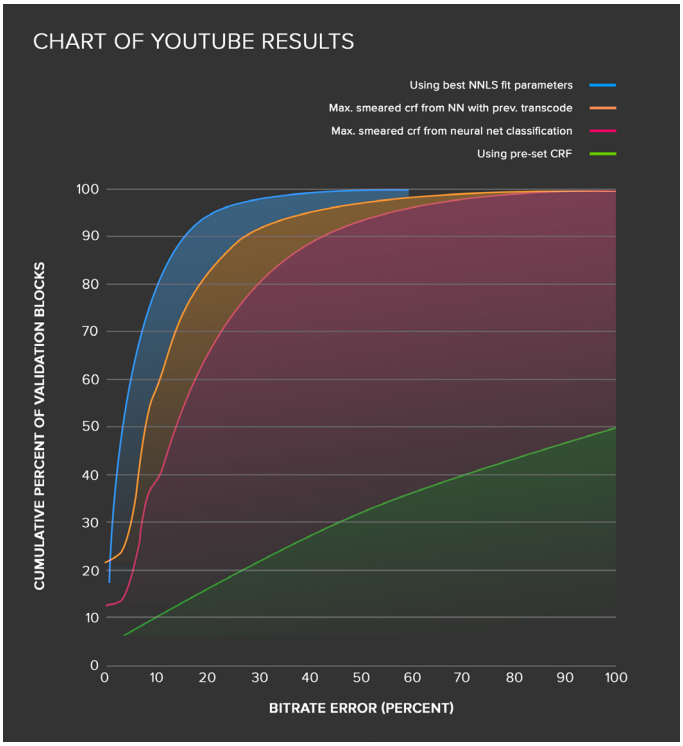

Figure 2. *Chart of YouTube neural network analysis results.*

Netflix and YouTube have the scale and budgets to invest heavily in automated per-title encoding solutions, but what about the rest of us? Following are several more affordable approaches.

CATEGORY-SPECIFIC ENCODING

Category-specific encoding works well for publishers with distinctly different content that can be easily grouped, like screencams, PowerPoint-based videos, and simple talking head videos.

Here are the steps to implementing category specific encoding:

1. Separate videos into distinct classes and identify 3-5 videos in each class that represents a cross section of the content within the class. These will serve as your test video data set.

2. Encode the videos at their highest distribution resolution using CRF encoding with a value of between 19-23. Play the videos to gauge quality, and try testing the files with a video quality metric you trust. Select a CRF value that delivers acceptable quality at the lowest possible data rate, and average the data rates in each class.

3. Encode the test files at this data rate using the technique you'll use for actual distribution, typically either constant bitrate encoding (CBR) or variable bitrate encoding (VBR). Play the files and verify that encoding quality is still acceptable. Test the file with an objective quality metric (such as PSNR), and focus on both the average quality and the lowest quality regions or frames. One caution is that data rates which work well for CRF encoding can cause transient quality issues with distribution encoding techniques, particularly CBR, so be sure to visually check all your encodes.

4. If the quality is acceptable at the rate shown in step 2, build your encoding ladder for each class of content from this resolution/ data rate.

This per-category encoding approach is generally less effective for premium movie and TV shows, since the range of encoding complexity within a class can vary significantly. As an example, though animated movies are typically easier to compress than real world videos, the range of animated techniques is highly diverse, from Mickey Mouse, to Big Buck Bunny, to Terminator and Transformers. This makes it challenging to create an encoding ladder that works across all animated content equally. Thus a weakness of the categorized content encoding approach can be seen when the variety inside a category varies widely.

Another challenge of per-category encoding is the variability between scenes in a single title. For example a sports show may include talking head in-studio shots along with fast action game shots and slow-motion recaps, each requiring different bitrate values to maximize quality while ensuring the least bits possible.

PROGRAMMATIC PER-TITLE ENCODING

The developers of enterprise-class encoding programs recognized the need for per-title encoding, and have started to implement this capability within their products. One of the first implementations is the Source Adaptive Bitrate Ladder (SABL) feature in the Cambria FTC encoder from Capella Systems.

Cambria uses a preset to encode adaptive bitrate streams for packaging into DASH, HLS, or Smooth Streaming formats. This preset contains an encoding ladder with a set number of streams along with their resolutions, data rates, and other configuration options.

Cambria's SABL is a scripted configurable process that works by running a CRF encode of the source file, during which it identifies the region with "peak complexity," essentially the hardest region to encode within the file. The script then uses the data rate produced by the CRF encode in this region to adjust the encoding ladder up or down by a percentage specified in the script.

AD HOC PER-TITLE ENCODING

Low volume publishers can use an analysis like that developed by Capella to gauge the encoding complexity of their individual files and customize their encoding ladder accordingly. Steps to implement this technique would be:

1. Identify a range of files in your library that represent a broad cross-section of encoding complexity. Ten files should be a sufficient starting point.

2. Use an FFmpeg script like the following, or similar configurations in your encoding program, to compute the data rate associated with the desired quality level, customizing the CRF value to achieve the desired quality. In the script, the `-psnr` command computes the PSNR value for each frame and for the file as a whole, which is stored in a log file via the `-report` command.

   ```
   ffmpeg -i input.mov -crf 23 -psnr output.mp4
   -report
   ```

   Which PSNR value should you target? In its per-title encoding analysis, Netflix commented that viewers cannot perceive the quality associated with PSNR values in excess of 45 dB, and that values below 35 dB often exhibit visible artifacts. For this reason, an overall PSNR value of between 42 - 44 dB is a realistic target, and you can adjust CRF values in the command line to consistently achieve this value. Or, you can use a different quality metric or tool to ascertain the desired quality level and adjust the CRF value accordingly.

3. Divide the ten files into different groups based upon data rate ranges and build an encoding ladder for each group. Three groups, low, medium, and high, is a good starting point.

4. For each new file, run the same FFmpeg command line shown above to compute the associated data rate. Use the data rate to slot the file into an existing group, and apply the associated encoding ladder.

CAPPED CRF ENCODING

Capped CRF encoding is a bitrate control technique available with some codecs, including x264, x265, and VP9. Capped CRF works like it sounds; you choose a CRF level and a maximum data rate. The encoder adjusts the data rate to deliver the specified quality level, but never exceeds the specified maximum. With x264 and FFmpeg, the maximum data rate is set with a combination of the maxrate and bufsize options, the latter of which controls the Video Buffering Verifier (VBV). As an example, consider the following command string:

```
ffmpeg -i input.mov -crf 21 -maxrate 4000k -bufsize
4000k output.mp4
```

This directs FFmpeg to encode at a CRF value of 21, but to cap the data rate at 4 mbps. In the case of a high-motion or otherwise hard to compress video file, the CRF value would likely return a data rate in excess of 4 mbps, which would be capped by the encoder, so the overall data rate would be close to 4 mbps. With easier to compress videos, the CRF value would return a data rate of less than 4 mbps, so the cap might never come into play.

Figure 3 illustrates a drawback of the capped CRF approach where locations that the bitrate cap was reached will pose the greatest challenge to VQ. Since the buffer may be quite full at this point, the quality can be severely compromised exactly at the point in the video when it will be most noticeable.

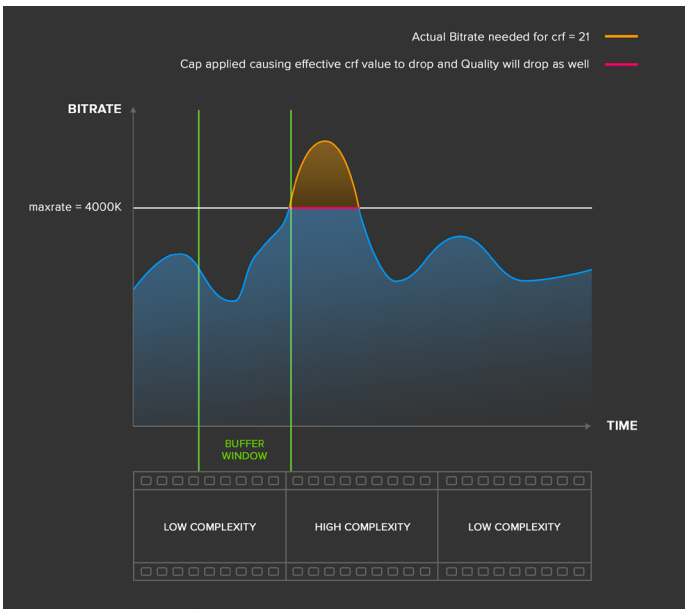

Figure 3. *Video quality compromise with capped CRF in high complexity scenes.*

BEAMR CONTENT ADAPTIVE PROCESS

Beamr is an image science and encoding technology company who has developed the world's first content adaptive optimization process built on a perceptual quality measure operating in a closed loop at the frame level. The quality measure and related technologies are backed by 19 granted International patents and over 30 pending applications. The technology is commercialized in an optimization software solution called Beamr Video that is applied post encode to compressed H.264 and HEVC streams prior to packaging. Beamr Video enables additional bitrate savings between 20% to 40% without introducing any perceptible change to the original video quality (VQ). Beamr Video is available as a Linux software application that can be deployed both on-premises and in the cloud, and is also available as a fully scalable cloud service accessible with a REST API.

Unlike other approaches, Beamr optimization works on a frame-by-frame basis to deliver the smallest possible size for each frame while ensuring the highest overall quality of the frame within the video sequence. This avoids transient quality issues that are often seen in other encoding and optimization techniques. The Beamr Quality Measure Analyzer has a higher correlation with subjective results than existing quality measures such as PSNR and SSIM, including CRF. This claim has been proven with ITU-R BT.500 testing, and by the "Golden Eyes" of Hollywood's largest studios where Beamr Video is used to preserve the quality of Blu-ray encodes while reducing file sizes 25% or more.

Figure 4 illustrates the effect of Beamr Video optimizing a live stream with just a single GOP delay, where even with an aggressive encoding recipe attractive bitrate savings are achieved.

Taking a deeper look, the Beamr optimization process is an iterative closed loop system built around the concept of determining whether the encoded frame is perceptually identical to the input frame. During this process, the original compressed frames are decoded, and then re-encoded using more aggressive compression parameters, thus delivering a more compact frame than the original. This input frame is then decompressed, and compared to the reference (source) frame using the Beamr Quality Measure Analyzer (process illustrated in **Figure 5**), which then returns a score that numerically represents whether the frame is perceptually identical to the original. The evaluation of frame quality also takes into account the temporal flow of the frame sequence, so each frame will be perceptually identical and the viewing experience fully preserved. If the score is within a certain range, Beamr Video accepts the frame and moves to the next.

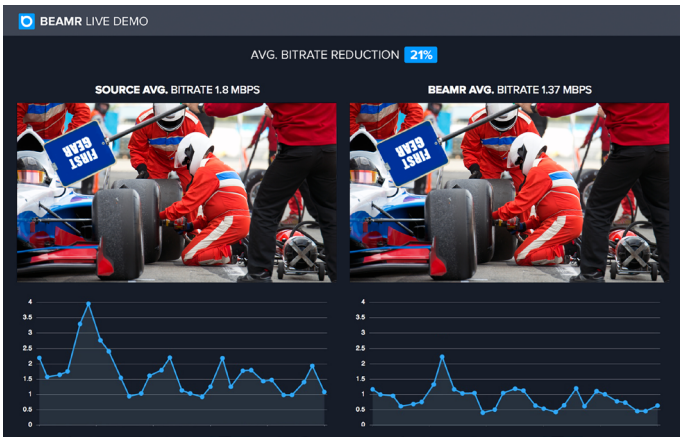

Figure 4. *Beamr's content adaptive optimization process in real world use*

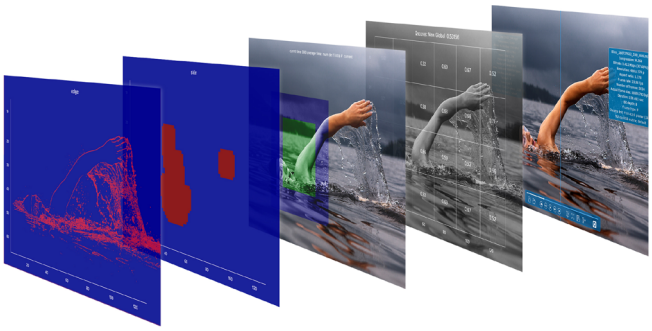

Figure 5. *Beamr Quality Measure Analyzer in action.*

If the score exceeds the quality threshold, this indicates more savings may be achievable, and Beamr Video will complete an additional encode at higher compression settings, analyze again, until the score is within the target range. Conversely, if the score is less than the quality threshold value, it indicates the frame is no longer visually identical to the original, and the optimizer will re-encode at a lower compression setting until the frame meets the quality threshold.

Beamr Video works across file-based workflows today, but is fully compatible with the requirements of low latency (real-time) encoding applications and will support live stream optimization as shown in Figure 4 by early 2017. Building on the company's market momentum, in late 2016, Beamr will introduce an end-to-end encode to optimize solution that is guaranteed to deliver the highest quality at the lowest bitrate possible, thereby delivering fully on the promise of content adaptive encoding.

BEFORE YOU START WITH CONTENT ADAPTIVE ENCODING, CONSIDER THESE FOUR THINGS

To modernize your encoding recipes, you should consider the following:

1. **Automation is preferred.** Large scale producers must reliably automate their analysis, encoding and workflow processes. Techniques that manually analyze files to determine optimal parameters should be reserved for low volume workflows and distributors.

2. **Scene and frame adaptive yields best results.** Methods that analyze the video to determine a fixed bitrate for VBR or CBR cannot provide the same level of granularity as techniques that perform this analysis at the scene or frame level. Beamr Video performs both the analysis and the selection of compression parameters on a frame basis in a closed loop, ensuring the optimal allocation of bits per frame.

3. **Perceptual quality metric driven.** While useful, CRF uses little data beyond the perceived motion in a frame to determine the required quality level. A sophisticated quality metric that is closely coupled to actual viewer perception will deliver a more accurate result across an entire library of content.

4. **Guaranteed quality must be guaranteed.** Several of the approaches described use CRF as a guide, but then encode using VBR or CBR without a post-encode quality verification step. Without the benefit of a closed loop, the final encoded file may have degraded scenes, transient quality issues, or simply be encoded at a higher data rate than is needed for the content.

The promise of content adaptive techniques to improve video quality and reduce bitrate is alluring. However many fall short in one or more key areas, whereby compromising video quality during exceptionally difficult scenes, because the encoder lacks intelligence to determine where to optimally apply bits in the frame. For this reason, only a closed loop, perceptual quality measure driven process can be 100% safe on all content, regardless of type or complexity. This makes solutions like Beamr's a highly attractive option for any video distributor, MSO, MVPD, OTT streaming platform or publisher needing to balance the vectors of bitrate and quality. Factors that are becoming competitive differentiators in a post net-neutrality world where technologies like 4k, virtual reality and augmented reality means consumer demand for higher and higher quality video, without a correspondingly high increase in bandwidth, requires encoding and optimization solutions that can go beyond the current state of the art.

CONTRIBUTED WRITING FROM:

Jan Ozer

Dror Gill

Tamar Shoham

Mark Donnigan

ABOUT BEAMR

Beamr is a global video technology leader in H.264 and HEVC codec, encoding and optimization solutions, with over 80 employees, and offices in Palo Alto, Tel Aviv and St. Petersburg, Russia. Beamr has received 20 granted patents with another 30 pending for the industry's first fully content adaptive perceptual optimization process that reduces the bitrate of video streams and files more than 20% without compromising picture quality.

Hollywood studios, MSO's and some of the world's largest OTT distributors use Beamr technologies to ensure video quality, reduce delivery costs, improve streaming user experience, and lower TCO of cDVR, VOD and TVE systems. The company's software is available across a broad range of platforms including cloud and on-premises installations.

Founded in 2009 by a team of leading image science experts, Beamr is backed by Marker LLC, Disruptive, and Alphabet (Google) Chairman Eric Schmidt's fund, Innovation Endeavors.