COMPLEXITY-BASED CONSISTENT-QUALITY ENCODING IN THE CLOUD

Jan De Cock, Zhi Li, Megha Manohara, and Anne Aaron

Netflix Inc.,100 Winchester Circle, Los Gatos, CA, United States

ABSTRACT

A cloud-based encoding pipeline which generates streams for video-on-demand distribution typically processes a wide diversity of content that exhibit varying signal characteristics. To produce the best quality video streams, the system needs to adapt the encoding to each piece of content, in an automated and scalable way. In this paper, we describe two algorithm optimizations for a distributed cloud-based encoding pipeline: (i) per-title complexity analysis for bitrate-resolution selection; and (ii) per-chunk bitrate control for consistent-quality encoding. These improvements result in a number of advantages over a simple "one-size-fits-all" encoding system, including more efficient bandwidth usage and more consistent video quality.

Index Terms— Encoding pipeline, parallel encoding, rate control

1. INTRODUCTION

Internet streaming allows video-on-demand (VOD) distributors such as Netflix to tailor their streams to the viewers' available bandwidth and viewing device capability. Streams are pre-encoded at various bitrates applying optimized encoding recipes. On the user's device, the client runs adaptive streaming algorithms which instantaneously select the best encode to maximize video quality while avoiding playback interruptions due to rebuffers.

Encoding with the best recipe is not a simple problem. For example, assuming a 1 Mbps bandwidth, should H.264/AVC video be streamed at 480p, 720p or 1080p resolution? At 480p, 1 Mbps will likely not exhibit encoding artifacts such as blocking or ringing, but if the user is watching on an HD device, the upsampled video will not be sharp. On the other hand, if we encode at 1080p we send a higher resolution video, but the bitrate may be too low such that most scenes will contain annoying encoding artifacts.

In a fixed-bitrate encoding system, codec parameters can be selected that produce the best quality trade-offs across different types of content. A set of bitrate-resolution pairs (referred to as a *bitrate ladder*), are selected such that the bitrates are sufficient to encode the stream at that resolution without significant encoding artifacts. This "one-size-fits-all" fixed bitrate ladder achieves, for most content, good quality encodes given the bitrate constraint. However, for some cases, such as scenes with high camera noise or film grain noise, a 5000 kbps stream would still exhibit blockiness in the noisy areas. On the other end, for simple content like cartoons, 5000 kbps is far more than needed to produce excellent 1080p encodes.

The titles in a VOD collection such as Netflix's have very high diversity in signal characteristics. For example, some animation titles reach very high PSNR (45 dB or more) at bitrates of 2500 kbps or less. On the other extreme, some titles with high action scenes or significant spatial texture (camera or film grain noise) require bitrates of 8000 kbps or more to achieve an acceptable PSNR of 38 dB. Given this diversity, a one-size-fits-all scheme obviously cannot provide the best video quality for a given title and user's allowable bandwidth. It can also waste storage and transmission bits because, in some cases, the allocated bitrate goes beyond what is necessary to achieve a perceptible improvement in video quality. Furthermore, even within a title, the signal characteristics can vary significantly, from simple talking head scenes to explosions and car chases.

In this paper we describe a cloud-based encoding system which selects optimized bitrate ladders *per title*. Given a selected bitrate-resolution pair, we further enhance the bitrate allocation by adapting the target bitrate of each video chunk to the complexity of that segment. The chunk-based algorithm is similar to the approach described in [1], where the authors propose multi-pass encoding to steer the bitrate of each video segment to meet maximum quality and bitrate constraints. However, for our approach we base the initial Constant Rate Factor (CRF) [2] value and the maximum bitrate of each chunk on the results of per-title complexity analysis.

This paper is structured as follows. In Section 2, we give an overview of our complexity analysis algorithm, which determines the per-title bitrate ladder. Section 3 describes enhancements to a VOD encoding pipeline which lead to per-chunk quality and rate control. In Section 4, we present results on a set of full-length titles from the Netflix catalog.

2. PER-TITLE COMPLEXITY ANALYSIS

To design the optimal per-title bitrate ladder, we select the total number of quality levels and the bitrate-resolution pair for each quality level according to several practical constraints. For example, we need backward-compatibility (streams are

978-1-4673-9961-6/16/$31.00 ©2016 IEEE

playable on all previously certified devices), so we limit the resolution selection to a finite set – e.g. 1920x1080, 1280x720, 720x480, 512x384, 384x288 and 320x240. In addition, the bitrate selection is also limited to a finite set, where the adjacent bitrates have an increment of roughly 5%. For optimality, we select the bitrate-resolution pair such that i) At a given bitrate, the produced encode should have as high quality as possible, and ii) The perceptual difference between two adjacent bitrates should fall just below one JND.

Fig. 1 shows an example where we encode a source at three different resolutions with various bitrates.

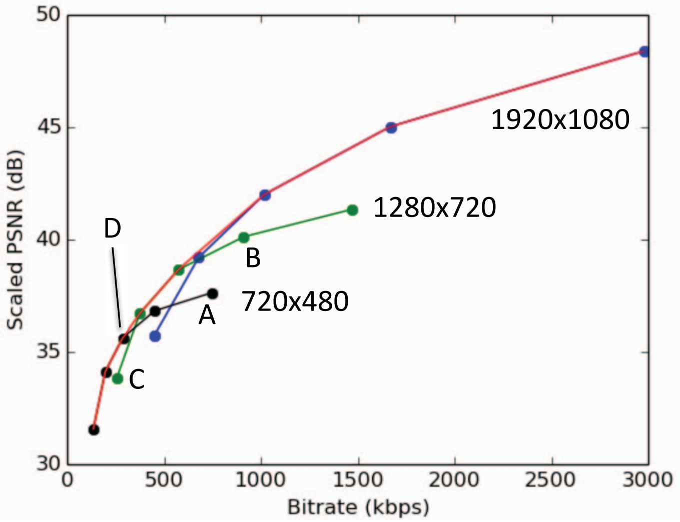


Fig. 1. Example encodes showing individual R-D curves and convex hull.

At each resolution, the quality of the encode monotonically increases with the bitrate, but the curve starts flattening out (A and B) when the bitrate goes above some threshold. On the other hand, a high-resolution encode may produce a quality lower than the one produced by encoding at the same bitrate but at a lower resolution (see C and D). This is because encoding more pixels with lower precision can produce a worse picture than encoding less pixels at higher precision combined with upsampling and interpolation. Furthermore, at very low bitrates the encoding overhead associated with every fixed-size coding block starts to dominate in the bitrate consumption, leaving very few bits for encoding the actual signal. Encoding at high resolution at insufficient bitrate would produce artifacts such as blocking, ringing and contouring.

We can see that each resolution has a bitrate region in which it outperforms other resolutions. If we collect all these regions from all the resolutions available, they collectively form the convex hull. Ideally, we want to operate exactly at the convex hull, but due to practical constraints (for example, we can only select from a finite number of resolutions), we would like to select bitrate-resolution pairs that are as close to the convex hull as possible.

It is practically infeasible to construct the full bitrate-quality graphs spanning the entire quality region for each title in a VOD catalogue. To implement a practical solution in production, we perform trial encodings at different CRF values, over a finite set of resolutions. The CRF values are chosen such that they are one JND apart. For each trial encode, we measure the bitrate and quality. By interpolating curves based on the sample points, we produce bitrate-quality curves at each candidate resolution. The final per-title bitrate ladder is then derived by selecting points closest to the convex hull.

In practice, a movie or TV show is composed of scenes of varying complexity. To account for this in the generation of the optimal bitrate ladder, we only utilize segments of the video that are at the high complexity end of the title. This guarantees optimal quality for the high complexity scenes but may over-allocate bits for the simple segments of the video.

3. ENCODING PROCESS

3.1. Per-title encoding

Once the complexity of the title has been analyzed and a per-title resolution-bitrate ladder has been constructed, the encoding process is launched. For each resolution-bitrate pair, a video encode is generated in the cloud-based video encoding pipeline as follows [3,4]: The video source is divided into fixed-length chunks and each chunk is independently encoded in the parallel encoding pipeline. After all the encode chunks are completed, a video assembler stitches the bitstreams together to produce the full bitstream of the title. The target bitrate is achieved by using two-pass bitrate-based rate control on each of the chunks, with the same target average bitrate for all the chunks.

3.2. Per-chunk bitrate setting and encoding

We enhance the encoding pipeline to support per-chunk bitrate variation. For each encode chunk, we select the bitrate such that it adapts to the complexity of the video for that specific segment. As mentioned above, the complexity analysis results in optimal resolution-bitrate pairs for that title. In addition, each resolution-bitrate pair (R_i, B_i) corresponds to a specific CRF value, C_i that was used to generate the trial encoding. This CRF number represents the *consistent quality* target for the title given the ladder point i. The objective of the per-chunk bitrate adaptation is to encode each chunk at resolution R_i with quality C_i and capped at bitrate B_i. Since the resolution-bitrate pairs for the title were chosen using the complex segments of the title, per-chunk adaptation results in an average bitrate across the title of less than B_i.

In particular, we apply multi-pass encoding. For each chunk n, the first pass uses CRF rate control at the desired CRF C_i, and the size of the resulting encode determines the chunk bitrate $B_{i,n}$. Based on this bitrate, two options are possible: (i) When the bitrate $B_{i,n}$ does not exceed the maximum bitrate B_i, $B_{i,n}$ is passed on as the rate target for the second pass. The second pass is bitrate-controlled and uses the per-frame statistics from the first pass. Compared to the one-pass CRF encode, the additional second pass allows for improved bit allocation and buffer compliance while still maintaining a consistent quality target across chunks. (ii) If the per-chunk bitrate exceeds the maximum bitrate, a regular two-pass encode is started with rate target B_i, leading to three passes overall for this chunk.

4. RESULTS

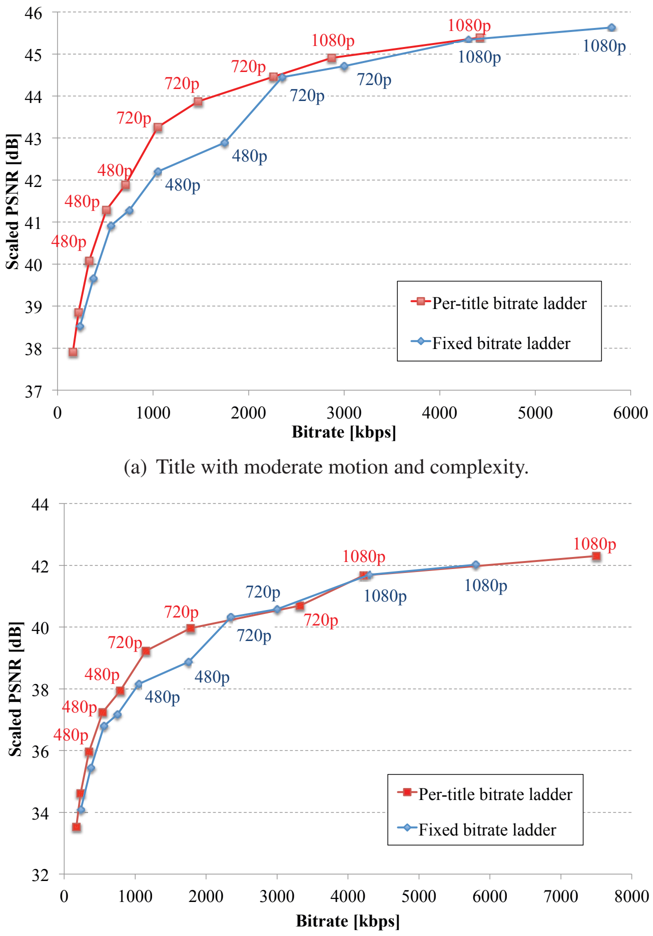
In this section, we present results of the complexity analysis and encoding process for a set of VOD titles in the Netflix catalog. Each title went through a cloud-based encoding pipeline consisting of source inspection, complexity analysis, multi-pass chunk encoding, and assembly [4]. To measure the quality of the encodes, we use both scaled PSNR (measured at 1080p resolution) and the Video Multi-method Assessment Fusion (VMAF) metric [3]. VMAF estimates quality by combining scores from multiple quality assessment algorithms, including Anti-Noise SNR, Detail Loss Measure [5], Visual Information Fidelity [6], and motion information, and shows a higher correlation with subjective quality scores than e.g. PSNR and SSIM.

4.1. Comparison between per-title encoding and fixed-ladder encoding

The per-title encoding scheme has the advantage that client devices can switch between resolutions at a bitrate more appropriate for each individual title. The gains in quality become apparent from the example rate-distortion graph in Fig. 2(a), where we show the fixed-rate and per-title R-D curves for a full-length episode of a drama series with moderate spatio-temporal complexity. R-D points for different resolutions are plotted in this graph, with their scaled PSNR values shown in the ordinate. When looking at the combined R-D points for a certain resolution (indicated for 480p, 720p and 1080p), the R-D curve for that particular resolution can be distinguished.

The red per-title curve shows how we are now encoding at the convex hull encompassing the individual per-resolution R-D curves. In this example, the transitions between resolutions occur at lower bitrates for the per-title ladder (e.g. 720p resolution is enabled at 1050 kbps instead of 2350 kbps). In particular at these transition points, we are obtaining substantial quality gains over the fixed-ladder approach.

Fig. 2(b) shows the RD curve for a second example title, further illustrating the benefit of encoding at the convex hull. For this example, there are clear gains at low resolutions, but for e.g. 1080p there is little benefit in switching to this resolution at a lower bitrate. We are, however, increasing the highest bitrate point to 7500 kbps. Visual inspection shows that encoding at this higher bitrate point better preserves the film grain present in this show.



(a) Title with moderate motion and complexity.

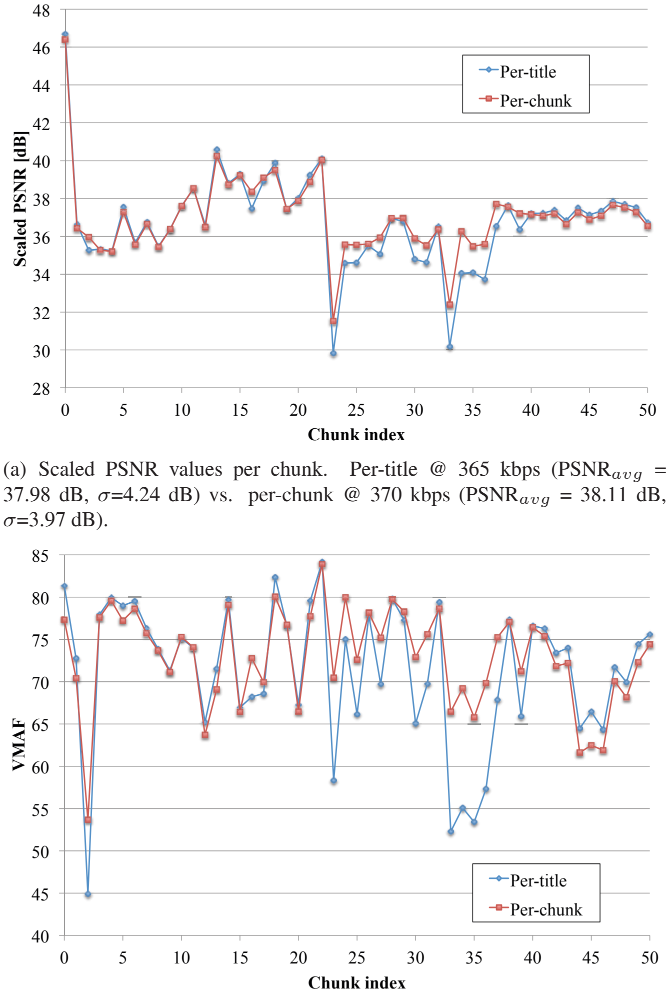
(b) Complex title containing film grain.

Fig. 2. Example R-D curves for fixed and per-title bitrate ladders.

4.2. Comparison between per-chunk and per-title encoding

Per-chunk encoding leads to a more consistent video quality across the entire title. On average, we obtain PSNR and VMAF values which are similar for both per-title and per-chunk encoding. The benefit of using per-chunk encoding (based on multi-pass rate control as described in Section 3), however, lies in the reduction of the *quality variation*, and in the increase of the minimum quality.

The per-chunk PSNR and VMAF values for the first 500 seconds of a drama series episode are plotted in Fig. 3. In these graphs, each point represents a 10-second chunk of the episode. On average for the whole episode, the quality scores are close to each other (e.g. a PSNR of 38.11 dB for per-chunk encoding vs. 37.98 dB for per-title encoding), but we can see significant differences in the per-chunk quality. This is more apparent from the VMAF quality scores. By using the per-chunk encoding approach, we are able to reduce the variation, and to limit the drops in quality. For this example, the standard deviation σ of the per-frame quality is reduced from 4.24 dB to 3.97 dB (PSNR), and from 10.91 to 9.16 (VMAF).



(a) Scaled PSNR values per chunk. Per-title @ 365 kbps (PSNR$_{avg}$ = 37.98 dB, σ=4.24 dB) vs. per-chunk @ 370 kbps (PSNR$_{avg}$ = 38.11 dB, σ=3.97 dB).

(b) VMAF score per chunk. Per-title @ 365 kbps (VMAF$_{avg}$ = 71.26, σ=10.91) vs. per-chunk @ 370 kbps (VMAF$_{avg}$ = 72.29, σ=9.16).

Fig. 3. Scaled PSNR and VMAF scores for the first 500 seconds of a drama series episode.

To aggregate the results over multiple titles, Fig. 4 shows the cumulative distribution function (CDF) of the quality variation for per-title and per-chunk encoding, where $\Delta\text{PSNR} = \text{PSNR} - \text{PSNR}_{avg}$. These statistics were collected on 10 full-length titles with varying spatio-temporal characteristics, representing more than 500,000 frames. The graph shows that the CDF for per-chunk encoding rises more sharply around zero, indicating a reduced quality variation.

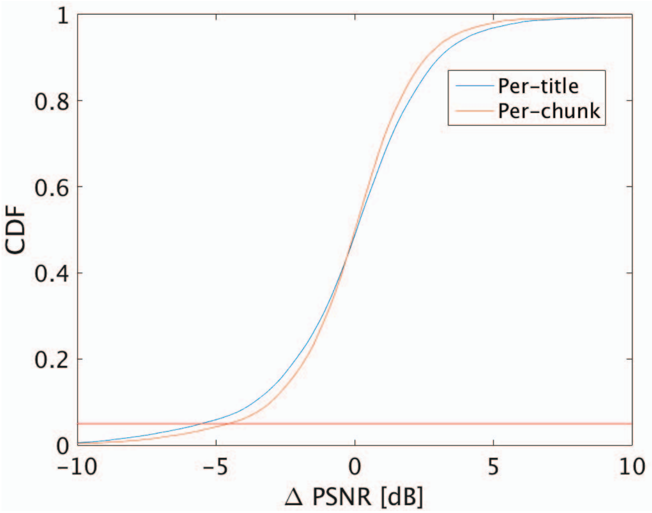


Fig. 4. Cumulative distribution function for ΔPSNR for per-title and per-chunk encoding (aggregated for 10 titles).

To improve quality of experience, we are interested in maximizing the minimal quality in the encoded streams. This is particularly important for the lower resolution encodes, where the quality variation is the highest. The CDF for per-chunk encoding indicates that we are effectively increasing the quality on the lower end. When looking at the fifth percentile of the frames, we improve the quality, as indicated by the red line in Fig. 4, and as detailed in Table 1. For the 480p encodes of the test set, we obtain gains of 0.5 dB for the lowest resolution, and a VMAF improvement of 3.15. The gains decrease for higher resolutions to about 0.3 dB (PSNR) and 0.45 (VMAF) for the 1080p encodes.

Table 1. Quality (scaled PSNR and VMAF) for fifth percentile of frames.

	Per-title		Per-chunk	
Resolution	PSNR [dB]	VMAF	PSNR [dB]	VMAF
480p	32.85	50.61	33.35	53.76
720p	36.32	64.77	36.74	66.06
1080p	37.77	72.87	38.05	73.32

5. CONCLUSIONS

In this paper, we gave an overview of two improvements that can be implemented in a VOD encoding pipeline. Based on a complexity analysis step, we are able to determine resolution-bitrate pairs which closely reflect the convex hull of the R-D curves at different resolutions. Furthermore, we are determining the bitrate of individual chunks, by using a CRF-based multi-pass encoding process. Per-chunk encoding was shown to lead to more consistent quality across the individual chunks, and to improve the minimal quality in the video streams.

6. REFERENCES

[1] Y.-C. Lin, H. Denman, and A. Kokaram, "Multipass encoding for reducing pulsing artifacts in cloud based video transcoding," in *IEEE International Conference on Image Processing (ICIP)*. IEEE, 2015, pp. 907–911.

[2] L. Merritt and R. Vanam, "Improved rate control and motion estimation for h.264 encoder," in *IEEE International Conference on Image Processing (ICIP)*. IEEE, 2007, pp. 309–312.

[3] A. Aaron, Z. Li, M. Manohara, J.Y. Lin, E.C.-H. Wu, and C.-C.J. Kuo, "Challenges in cloud based ingest and encoding for high quality streaming media," in *IEEE International Conference on Image Processing (ICIP)*, 2015, pp. 1732–1736.

[4] A. Aaron and D. Ronca, "High quality video encoding at scale," Netflix Tech Blog, http://techblog.netflix.com/2015/12/high-quality-video-encoding-at-scale.html, December 9, 2015.

[5] S. Li, F. Zhang, L. Ma, and K.N. Ngan, "Image quality assessment by separately evaluating detail losses and additive impairments," *IEEE Transactions on Multimedia*, vol. 13, no. 5, pp. 935–949, Oct 2011.

[6] H.R. Sheikh and A.C. Bovik, "Image information and visual quality," *IEEE Transactions on Image Processing*, vol. 15, no. 2, pp. 430–444, Feb 2006.