Netflix like Encoding Optimization

Expose Master Thesis

Vijaykumar Rana

AVT: Prof. Dr.-Ing. Alexander Raake Alexander Raake, M.Sc. Steve Göring

February 23, 2019

1 Introduction

YouTube and Netflix are the most popular video hosting websites in the world right now. They have evolved their encoding schemes in a way such that the end user would receive highest possible quality of videos corresponding to the bitrates they can afford. This has been made possible using schemes like segment based encoding where a video is chopped into several chunks and then encoded individually. Meanwhile Netflix employs shot based encoding, where a video is divided into chunks based on the content of a shot. [MM]

Now to provide users with high quality content Netflix developed Per-Title Encode approach. [Net] The idea behind this approach is that every video has different content and therefore they have to be encoded differently unlike using traditional approach where every video stream is encoded with preset parameters which employs "one-size-fits-all" fixed bitrate ladder because of the bitrate constraints.

The approach of shot based encodes is about encoding individual shots with all possible encoding schemes(shots being the chunks of the video based on the content), followed by the analysis of quality to choose highest achievable quality for specific bitrate. In order to achieve this the encoding parameters have to be optimized. This is done by analyzing convex-hull curve obtained from rate-quality (or rate-distortion) function generated using the quality obtained using a metric called VMAF(Video Multimethod Assessment Fusion)[Zhi].

As it can be seen in the Figure 1, Lets assume a point at 1500 kbps, there are two points that can be obtained from curve B and curve C as it can be seen that for the same bitrate of 1500kbps curve C offers more quality with higher resolution. Another scenario could be at bitrate 500kbps, three points from curve A, curve B and curve C can be observed respectively. C provides higher resolution compared to curve A and curve B but the quality (as shown by PSNR value on y axis) of curve C at that point has the lowest quality. Next A has better quality and finally B has the highest. So for 500kbps, the point on curve B is considered.

This then leads to optimization on an elementary level. This granular optimization of the encodes within a video stream is achieved using a framework called Dynamic Optimizer.[Ioa] This approach has resulted in more efficient encoded videos with lower bitrates compared to conventional encoding schemes using H.264/AVC or H.265/HEVC. The results are shown and compared by the authors from [KG18].

2 Motivation and Goals

To summarize the procedure used by Netflix would be as follows: a video is divided into chunks called as shots based on the content (e.g, shots with constant background or shots with action scenes where there is camera noise or film grain noise); then each of these shots are analyzed with all possible settings to find best quality for different bitrates as shown in Figure 1. So, basically the points in the

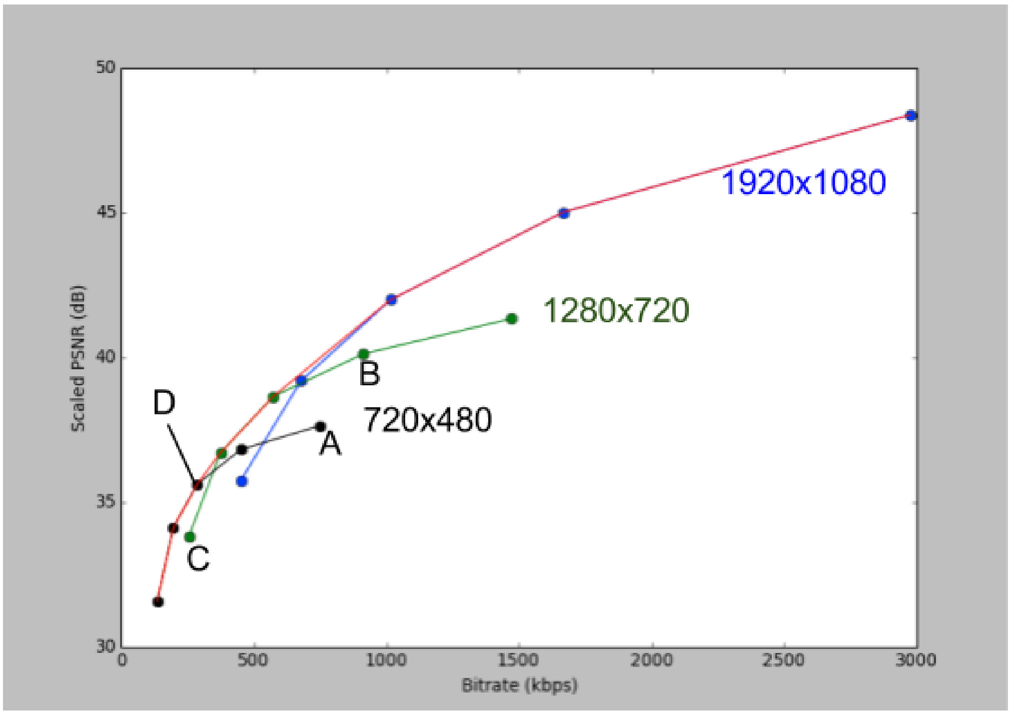

Figure 1: Encoding at three resolutions and various bitrates. Blue marker depicts encoding point and the red curve indicates the PSNR-bitrate convex hull. [Net]

R-D(Rate-Distortion or R-Q) curve obtained is from several individual settings which is a bruteforce technique to do so. The granular optimization is achieved after applying brute force technique to obtain the convex hull of R-D(R-Q) curve and then decide for the hightest quality to be selected for specific bit rate. The number of overall computation steps for each resolution for several bitrates would be quite high.

The motivation here is to do the computation for less number of bitrates and try to obtain other points by the means of any approximation method. Lets take a look at the Figure 2 which depicts a R-Q curve, if we compute quality at just the three red points (bitrates) for a given resolution and try to approximate the points in between using certain interpolation scheme or any mathematical formulation method(e.g., interpolation techniques like B-Spline Interpolation [Bsp] method) then the overall number of encoding steps would be reduced. The next step would be to compare the resulting points after approximation with points from the brute-force approach. Further analysis may provide error between points obtained using approximation approach and points from the brute-force technique. Based on the error the plausibility of the approach to minimize the overall encoding steps can be analyzed. The smaller steps for resolution will be considered and corresponding to each resolution, the computation of the quality parameters(QP) for all possible bitrate will be done.

3 First Ideas and Possible Approaches

3.1 Brute-force encoding Pipeline

Initially, the research about the video processing techniques are to be done. Then further getting into encoding pipeline. The pipeline to be implemented will be responsible for encoding for each video sample. Next the quality parameters are computed for encoded videos of different resolutions at all possible bitrates. The quality parameters will be computed using VMAF. Later on diving into the concept of VMAF and applying to encoded video clips to obtain quality parameters(QP). There are two VMAF scoring used for quality analysis, $\text{VMAF}_{\text{linear}}$(based on arithmetic-mean averaging) and $\text{VMAF}_{\text{harmonic}}$

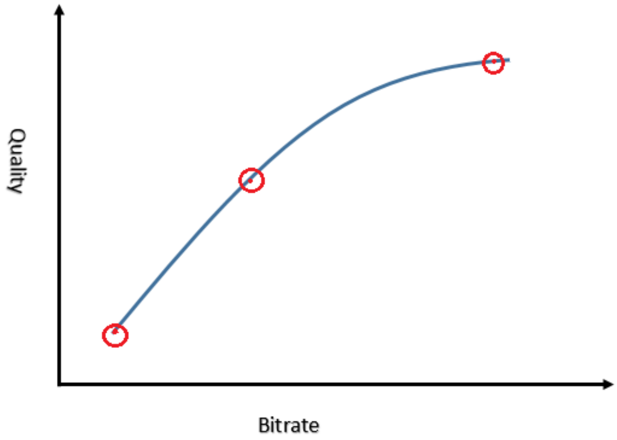

Figure 2: Sample Rate vs Quality curve.

(based on harmonic-mean averaging). These points are to be plotted to generate R-Q curve for all bitrates corresponding to each of the resolutions.

3.2 Analysis of Videos and Approximation method Development

As the approach used by Netflix is based on per-title encode, so each video is encoded and analyzed separately. Based on the Quality Parameter (QP) points obtained R-Q curve for all them of has to be analyzed as well. Next step would include developing a methodology to approximate the brute-force approach using interpolation techniques or mathematical modelling. Selecting a few number of points from the R-Q curve(points may include extreme points for a quality resolution and one or more points in between) and then trying to generate the points in between based on interpolation technique like B-spline interpolation.

After the new points are generated in between selected points from old R-Q curve obtained using brute-force approach, these new points are then compared with the old points for the given video stream. This procedure of generating points for brute-force R-Q curve and then generating points from R-Q curve of approximation approach and then comparison of points from the approaches is done for each video stream. A video stream to be evaluated would be 10-20 seconds long.

3.3 Tools and frameworks

The tool to be used for the implementation will be Python 3.x. It provides framework for various quality metrics and analysis tools for both spatial and temporal properties. The framework is available from Scikit. It provides various open-source sub-modules like Scikit-learn [Sle], Scikit-image [Sim], Scikit-video [Svi] etc.

Scikit-learn provides the tools for efficient data-analysis and data-mining. Scikit-image provides algorithms for analysis in image processing. Scikit-video provides tools for video processing which allows the users for manipulation of video and its properties.

4 Time-plan

Below is the ideal plan for different milestones to be achieved with specified time duration.

- ▷ Early April: Initial research about the topic.

- ▷ Mid April: Design of pipeline for brute-force technique for encoding sample videos.
- ▷ Mid May: Apply video to the pipeline implemented to generate encoded videos.
- ▷ Late May: Obtain the quality parameters using VMAF.
- ▷ Early June: Setup approximation algorithm and implementation.
- ▷ Mid June: Write initial chapters for Introduction and Related Work.
- ▷ Early July: Investigating the results from approximation techniques.
- ▷ Early August: Analysis and validation of the outcome from the results of investigation.
- ▷ Mid August: Write chapters for techniques used; description regarding observed results; Abstract description about approximation methodology.
- ▷ Mid August: Testing and debugging of the algorithm to further refine it to improve the results.
- ▷ Late August: Gather results and evaluate the results.
- ▷ Early September: Write chapters on observation and its implications.
- ▷ Mid September: Finishing Writing and submission.
- ▷ Defend the thesis.

References

[Bsp] B-spline. URL: https://en.wikipedia.org/wiki/B-spline (visited on 02/18/2019).

[KG18] Ioannis Katsavounidis and Liwei Guo. "Video codec comparison using the dynamic optimizer framework". In: Applications of Digital Image Processing XLI. Vol. 10752. International Society for Optics and Photonics. 2018, 107520Q.

[MM] Jan De Cock Ioannis Katsavounidis Anne Aaron Netflix Technology Blog Megha Manohara Anush Moorthy. Optimized shot-based encodes: Now Streaming! URL: https://medium.com/netflix-techblog/optimized-shot-based-encodes-now-streaming-4b9464204830 (visited on 02/15/2019).

[Sim] Scikit-image: Image Processing in Python. URL: https://scikit-image.org/ (visited on 02/22/2019).

[Sle] Scikit-learn: Machine Learning in Python. URL: https://scikit-learn.org/stable/# (visited on 02/22/2019).

[Svi] Scikit-video: Video Processing in Python. URL: https://scikit-video.org/stable/ (visited on 02/22/2019).

[Ioa] Ioannis Katsavounidis, Sr. Research Scientist, Video Algorithms - Netflix Technology Blog. Dynamic optimizer — a perceptual video encoding optimization framework. URL: https://medium.com/netflix-techblog/dynamic-optimizer-a-perceptual-video-encoding-optimization-framework-e19f1e3a277f (visited on 02/18/2019).

[Net] Netflix Technology Blog. Per-Title Encode Optimization. URL: https://medium.com/netflix-techblog/per-title-encode-optimization-7e99442b62a2 (visited on 02/16/2019).

[Zhi] Zhi Li, Anne Aaron, Ioannis Katsavounidis, Anush Moorthy, Megha Manohara - Netflix Technology Blog. Toward A Practical Perceptual Video Quality Metric. URL: https://medium.com/netflix-techblog/toward-a-practical-perceptual-video-quality-metric-653f208b9652 (visited on 02/18/2019).