Katsenou, A. V., Zhang, F., Swanson, K., Afonso, M., Sole, J., & Bull, D. R. (2021). *VMAF-based Bitrate Ladder Estimation for Adaptive Streaming*. https://arxiv.org/abs/2103.07564

Early version, also known as pre-print

# VMAF-based Bitrate Ladder Estimation for Adaptive Streaming

Angeliki V. Katsenou*, Fan Zhang*, Kyle Swanson†, Mariana Afonso†, Joel Sole†, and David R. Bull*

*Visual Information Lab, University of Bristol, Bristol, BS1 5DD, UK

†Netflix, Inc, Los Gatos, California, USA

*Abstract*—In HTTP Adaptive Streaming, video content is conventionally encoded by adapting its spatial resolution and quantization level to best match the prevailing network state and display characteristics. It is well known that the traditional solution, of using a fixed bitrate ladder, does not result in the highest quality of experience for the user. Hence, in this paper, we consider a content-driven approach for estimating the bitrate ladder, based on spatio-temporal features extracted from the uncompressed content. The method implements a content-driven interpolation. It uses the extracted features to train a machine learning model to infer the curvature points of the Rate-VMAF curves in order to guide a set of initial encodings. We employ the VMAF quality metric as a means of perceptually conditioning the estimation. When compared to exhaustive encoding that produces the reference ladder, the estimated ladder is composed by 74.3% of identical Rate-VMAF points with the reference ladder. The proposed method offers a significant reduction of the number of encodes required, 77.4%, at a small average Bjøntegaard Delta Rate cost, 1.12%.

## I. Introduction

The importance of visual communications in our daily activities and interactions has increased dramatically in recent years, not least due to restrictions imposed by the global COVID-19 pandemic. We are all creating and consuming increased volumes of video data with video streaming companies reporting major increases in video downloads shortly after WHO declared COVID-19 as a pandemic [1].

HTTP Adaptive Streaming (HAS) is a process employed by most video services to address dynamically changing network conditions. In Dynamic Adaptive Streaming over HTTP (DASH) [2], video content is encoded by varying spatial resolution and quantization level in order to adapt to the changing state of a heterogeneous network and to differing display device specifications. For example, if a streaming client monitors a change in the rate of an incoming video chunk that cannot support a smooth (without re-buffering) play-out, it will signal the need to switch to a stream at a lower bitrate. To this end, the creation of a set of video encodings at different bitrates is required at the server. This set of encodings are normally represented using a bitrate ladder. The traditional HAS solution uses a fixed bitrate ladder (a set of fixed bitrate-resolution pairs) but this approach cannot ensure a high quality of experience for all types of video content.

An improvement over this fixed solution is to introduce differentiation based on content genre, e.g. [3]. For example, higher bitrates can be used for sports content with rapid motion and fast scene changes. Previous solutions, however, were not tailored to video content characteristics, resulting in noticeable visual artifacts.

Recently, content-customised solutions have been reported and adopted by industry, such as those used by Netflix [4]–[7]. The key task is to invest in pre-processing where each video title is split into shorter clips or chunks, usually associated with shots. Each short video chunk is encoded using optimized parameters, i.e. resolution, quantization level, intra-period, etc., with the aim of building the Pareto Front (PF) across all Rate-Quality curves. Then a set of target bitrates is used to find the best encoded bitstreams. The quality metric used for this in the Netflix case is Video Multi-method Assessment Fusion (VMAF) [8]. Given the extensive parameter space (compression levels, spatial and temporal resolution, codec type etc.) and taking into account the fact that this process must be repeated for each video chunk, the amount of computation needed is massive. As a consequence, the industry heavily relies on cloud computing services, and this comes at a high cost in financial, time and compute terms.

Many other approaches that provide content-driven customisation have been proposed recently. Most of these methods first conduct a complexity analysis. An approach reported by Bitmovin [9], [10], performs a complexity analysis on each incoming video and inputs that into a machine-learning model to adjust the encoding profile to match the content. CAMBRIA [11], estimates the encoding complexity by running a fast constant rate encoding [11]. In [12], trial encodes are used to collect coding statistics at low resolutions and these are utilized within a probabilistic framework to improve encoding decisions at higher resolutions. MUX [13] introduced a deep-learning based approach that takes, as input, the vectorized video frames and predicts the bitrate ladder. Another interesting approach that takes into account both quality constraints and bitrate network statistics was proposed by Brightcove [14], [15]. The quality metric used in this case was the Structural Similarity Index Measure (SSIM) and bitrate constraints were based on probabilistic models. Finally, recently iSize [16] proposed the use of pre-encodes within a deep learning framework to decide on the optimal set of encoding parameters and resolution at a block level.

While all of the above solutions are significant and have contributed in the enhancement of video services, it is not possible to make direct detailed comparisons as they are proprietary. In our previous work [17], we predicted the intersection points of the PSNR-Rate curves. Then, in [18],
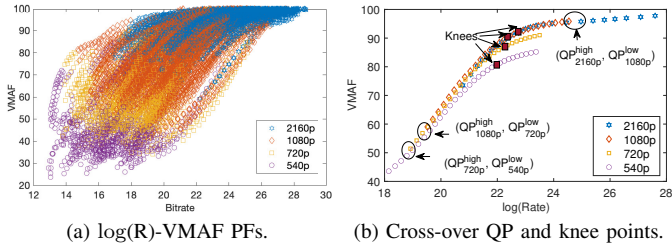
(a) log(R)-VMAF PFs.  (b) Cross-over QP and knee points.

Fig. 1: Rate-VMAF curves and Pareto fronts of the considered dataset across four spatial resolutions: 2160p, 1080p, 720p, and 540p.



(a) Distribution of knee QPs.  (b) Ladder points for all dataset.

Fig. 2: Knee points and reference VMAF bitrate ladder for all dataset.

we extended the method to the estimation of the bitrate ladder by using encodings at the intersection points to estimate the Pareto Front (PF) parameters, resolution and quantization parameters, at the target bit rates.

In this paper, we propose a new content-driven method that offers an improved bitrate ladder estimation based on VMAF. VMAF has been shown to exhibit a better correlation with perceptual quality than PSNR; hence the resulting bitrate ladders should deliver perceptually improved video streams. The method makes a feature-based prediction of the highest curvature points of the Rate-VMAF curves to guide a small set of initial encodings close to the area of interest. The results show significant improvement in terms of the number of required computations for only a small mean Bjøntegaard Delta Rate (BD-Rate) [19] cost.

The remainder of this paper is structured as follows. Section II describes the dataset and the Rate-VMAF curve characteristics. In Section III, the the definition of the reference bitrate ladder is provided. The proposed framework and the evaluation results are elaborated in Sections IV-V. Finally, conclusions are summarised in Section VI.

## II. RATE-VMAF CURVES AND CHARACTERISTICS

### A. Description of the Dataset

We employed the same dataset of 100 publicly available UHD video sequences as in our previous work [17], [20], [21]. The sequences have a native resolution of 3840×2160, the chroma format is 4:2:0, the bit depth is 10, and the frame rate 60 fps. Each sequence contains a single scene (no scene cuts) including a variety of different objects/scenes/regions of interest, camera motions, colours, and spatial activity. In this paper, we consider {2160p, 1080p, 720p, 540p} as the set of test resolutions used to develop and validate our methods. We use the Lanczos-3 filter [22] for spatial down/up-sampling throughout.

### B. Rate-VMAF Curves across Resolutions and the Pareto Front

Rate-VMAF curves exhibit characteristics that are different to those produced by other quality metrics. For consistency with previous work and ease of visualization, we convert these first to the log(Rate) domain. In Fig. 1, we illustrate the resulting Pareto surfaces for our dataset across the four spatial resolutions. It is clear that working in the log(Rate) domain is beneficial as the curves become smoother. Besides this, the saturation of VMAF at high bitrates and high resolutions is
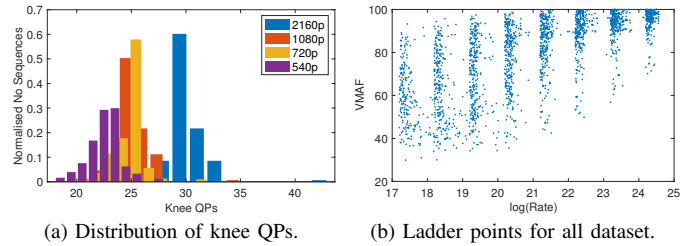
evident. This characteristic is content dependent and can be exploited when building a bitrate ladder.

### C. Intersection Points of Rate-VMAF Curves

An important characteristic of the Pareto Front (PF), used for constructing the bitrate ladder, is the set of points where resolution switches, i.e. the intersection points of the Rate-VMAF curves. It is more practical to define these intersection points as pairs of QP values, called cross-over QPs [18], $\left(QP_s^{high}, QP_{s-1}^{low}\right)$, with $s \in \mathcal{S}$ resolutions of the intersecting curves of the same video sequence. $level \in \{high, low\}$ defines the range of QPs. The resolution and level cannot be the same for both QPs in a pair. For example, in Fig. 1, the pair $(QP_{1080p}^{high}, QP_{720}^{low})$ represents the 1080p intersection with the 720p curve.

### D. Knee Points of Rate-VMAF Curves

An important characteristic of a Rate-VMAF curve is the point of highest curvature or "knee" point, $K$. This gives an indication of when the rate of improvement of the video quality will start decreasing, as shown in Fig. 1(b). We use the Kneedle algorithm, as described in [23], to compute the knee points of the curves across the resolutions. This algorithm is based on the notion that the points of maximum curvature in a dataset are approximately the set of points in a curve that are local maxima, if the curve is rotated clockwise by an angle defined by the line that connects the lowest and highest values in the dataset.

As shown in Fig. 2, the distributions of the knee QPs for the different sequences are quite tight around their mean values: 30.00±1.60 for 2160p, 24.99±1.72 for 1080p, 24.87±1.50 for 720p., and 23.08±1.52 for 540p. It is also important to note that although the knee points of the higher resolutions are usually part of the PF, the knee points of the lower resolutions are typically not part of it. This can be observed in the example of Fig. 1 (b).

## III. THE VMAF BITRATE LADDER

We first perform exhaustive encodings across resolutions for a wide range of QP values to construct the optimal log(Rate)-VMAF Pareto Front (which will serve as our reference) and then determine the intersection points of the log(Rate)-VMAF curves between different spatial resolutions. These intersection

points mark the limits of the bitrate range for which encoding at the given resolution yields the best quality [1].

## A. Definition

The initial step in constructing the bitrate ladder selects the target bitrates that will represent the rungs, i.e. $\mathcal{R}_{\mathcal{L}} = \{R_{L,1}, R_{L,2}, \ldots, R_{L,|\mathcal{L}|}\}$, where $|\mathcal{L}|$ is the cardinality of $\mathcal{R}_{\mathcal{L}}$ and $R_{L,1} < R_{L,2} < \ldots < R_{L,|\mathcal{L}|}$. The VMAF bitrate ladder is fully defined as a set of tuples $\mathcal{L}$ that comprise bitrate values $\mathcal{R}_{\mathcal{L}}$, the associated set of VMAF values $\mathcal{V}_{\mathcal{L}}$, a set of QP values $\mathcal{QP}_{\mathcal{L}}$, and a set of resolutions $\mathcal{S}_{\mathcal{L}}$, i.e.

$$\mathcal{L} := \{\langle R_{L,i}, V_{L,i}, QP_{L,i}, S_{L,i}\rangle\}_i^{|\mathcal{L}|}, \tag{1}$$

with $R_{L,1} < R_{L,2} < \ldots < R_{L,|\mathcal{L}|}$, $V_{L,1} \leq V_{L,2} \leq \ldots \leq V_{L,|\mathcal{L}|}$, $S_{L,1} \leq S_{L,2} \leq \ldots \leq S_{L,|\mathcal{L}|}$. The above constraints ensure the monotonicity of the set of parameters $\mathcal{R}_{\mathcal{L}}$, $\mathcal{V}_{\mathcal{L}}$, and $\mathcal{S}_{\mathcal{L}}$. For the set $\mathcal{QP}_{\mathcal{L}}$, monotonicity is assumed for $QP_{L,i}$ values of the same resolution.

## B. Building the Reference Ladder

In order to construct the bitrate ladder, we sample the Pareto front using the set of target bitrates $\mathcal{R}_{\mathcal{L}}$. From the resulting points, we check whether it is meaningful to retain all ladder rungs if we cannot significantly improve quality. Therefore, we monitor the slope of the sampled Rate-VMAF points so that: $\frac{dV_{L,i}}{dR_L} > \epsilon$ when $V_{L,i} > V_{high}$, where $\epsilon \in \mathbb{R}, \epsilon \rightarrow 0$ and $V_{high} = 97$. As a consequence of the above constraint, the length of the ladder might vary. The use of variable ladder lengths was suggested before in [14] and is dependent on content features and their relation to perceptual quality.

We considered the [150kbps,25Mbps] bitrate range for the ladder and that each new bitrate rung is twice that of the previous one, i.e. $R_{L,i} = 2R_{L,i-1}$. As can be seen in Fig. 2, the eight rungs on the ladder are clearly visible and are shifted to a greater or lesser extent according to the sequence. As VMAF is bounded by a maximum value of 100, the points become increasingly dense at higher VMAF values and higher bitrates. For about half (49%) of the tested sequences, there are fewer than eight rungs on the ladder - typically associated with sequences that can reach to a visual quality equivalent to the original (according to VMAF). These could be static sequences, with low amounts of structural or textural information that require lower bitrates for high quality reconstruction.

## IV. PROPOSED METHOD

Previous work [17], [18] showed that the best performing method in terms of BD-Rate cost was the interpolation-base method. The method proposed in this paper is based on encoding using only a subset of QP values per resolution. The selection of the subset is content-driven and is related to the knee point of the curve. After encoding, piece-wise cubic Hermite interpolation [24] is applied to for the interim QPs. Based
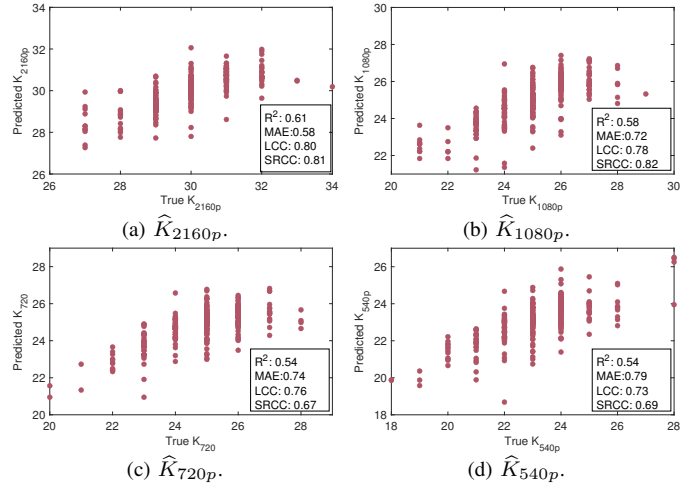
Fig. 3: Predicted Knee Points against the ground truth. Selected features used for prediction: F1-2, F4-5, F7, F9-10, F12, F15-17.

on these values, the PF is extracted. This method produces a suboptimal solution, whose accuracy depends on the number of encodes performed per resolution. The added benefit of this method is that it significantly reduces the number of encodings required compared to exhaustively encoding at all QPs.

In this work, we propose a Content-driven Interpolation-based Ladder (CIL) estimation method. This method uses content features to estimate the knee of the curve at each resolution for each sequence. Spatio-temporal features are extracted first to predict the knee QPs. We followed a sequential prediction of the knee QPs starting from the highest resolution down to the lowest. At each step, we applied feature selection, and particularly recursive feature [25]. Next we trained and tested several machine-learning regression methods, including Support Vector Machines with different kernels and Random Forests, finding that Gaussian Processes (GP), with a 5/2 Matérn covariance kernel [26], performed best for this work. To avoid overfitting, we deployed a ten-fold random cross-validation process. The results from the ten-fold cross-validation are shown in Fig. 3. Despite the fact that $R^2$, LCC, and SRCC values are not as very high, the MAE is small (<0.79), which is adequate to yield good results for the bitrate ladder estimation, as shown in Section V.

Next, the knees are used to subsample the QP range that falls within the PF. As observed for the higher resolutions, the knee points have a high probability of belonging to the PF, while this is less probable for the lower resolutions, we determine the initial encodes by evenly spacing a number $n \in \mathbb{N}$ of QPs within the following ranges:

$$QP_{j,s} \in [K_{j,s} + t_s, QP_{\max}], \tag{2}$$

where $j$ denotes the sequence, $t_s \in \mathbb{Z}$ is an offset, and $s \in \{2160p, 1080p, 720p, 540p\}$.

After encoding on the QPs above, piece-wise cubic Hermite interpolation is applied to find the Rate-VMAF values for the interim QPs. The bitrate ladder is constructed based on the encodings using the estimated QPs at the target bitrates.

## V. Predicting the Bitrate Ladder

The HEVC reference software (version HM 16.20) was employed in this study using its Random Access mode, a 64-frame Intra Period and a Group of Pictures (GoP) length of 16 frames [27], [28]. After encoding, decoding, and upscaling the spatial resolution to 2160p, we computed VMAF and bitrate at a GoP level that enabled a larger coverage of the Rate-VMAF space.

### A. Content Features

From the vast variety of low-level content features, we employ those spatio-temporal features that have been successfully used in our previous related work [17], [18]:

- Gray Level Co-occurrence Matrix (GLCM) [29] mean descriptors across frames: contrast F1.meanGLCM$_{con}$; correlation F2.meanGLCM$_{cor}$; homogeneity F3.meanGLCM$_{hom}$; energy F4.meanGLCM$_{enr}$; entropy F5.meanGLCM$_{ent}$.
- Temporal Coherence (TC) [30] with its interframe statistics: mean F6.meanTC$_{mean}$; standard deviation F7.meanTC$_{std}$; skewness F8.meanTC$_{skw}$; kurtosis F9.meanTC$_{kur}$; and entropy F10.meanTC$_{entr}$, all expressed as a mean across frames.
- Interframe Normalised Cross-Correlation (NCC) [31] statistics: mean F11.meanNCC$_{mean}$; standard deviation F12.meanNCC$_{std}$; skewness F13.meanNCC$_{skw}$; kurtosis F14.meanNCC$_{kur}$; and entropy F15.meanNCTC$_{entr}$, all expressed as a mean across frames.
- Mean Squared Error of the spatial Rescaling (RsMSE) of the first frame [20]: F16.RsMSE$_{1080p}$ (from 2160p to 1080p), F17.RsMSE$_{720p}$ (from 2160p to 720p).

### B. Compared Methods

Because there are no publicly available implementations of the proprietary solutions described in Section I, we have considered and tested the methods described below.

*1) Reference Ladder (RL):* This exhaustive search approach was used to construct our reference bitrate ladder as explained earlier. All sequences were encoded at QP values within $\{15, 16, \ldots, 45\}$ range. The reference bitrate ladder was constructed as explained in Section III-B.

*2) Naive Interpolation-based Ladder (NIL):* This method is based on encoding using only seven QP values per resolution, as in [18]. After encoding, a piece-wise cubic Hermite interpolation [24] is used to estimate the Rate-VMAF values for the interim QPs. Based on these estimated points, the ladder is constructed by encoding at the closest QP to the target bitrates.

*3) Content-driven Interpolation-based Ladder (CIL):* This is the proposed method as described above. To determine the offset $t_s$ values, the ranges of the distributions of cross-over QPs were combined with the distributions of the knee points. The presented results are for $t_{2160p} = t_{1080p} = -4$, $t_{720p} = 6$, and $t_{540p} = 10$. Furthermore, we explored the impact of the different number of initial encodes required per resolution $n \in \{4, 5, 6, 7\}$ and named accordingly the versions, CIL-$n$. We only considered up to 7 encodes in order to directly compare to NIL that uses seven initial encodes per resolution.

*4) Feature-based Predicted Ladder (FL):* The same spatio-temporal features as in CIL case are used. Similarly, feature selection, training and sequential prediction of the cross-over points per resolution takes place. GPs are also employed here with a ten-fold cross-validation. Then encodings at the cross-over QPs and on additional points are used to define linear models that help estimating the QP at the target bitrate. Further to the encodes at the six cross-over QPs, we require two additional in order to determine the unknown parameters of the linear model at each resolution. Particularly, we require one more encode for the 2160p and one for the 540p. The QP value selection for the extra encodes for each sequence $j$ is decided as below:

$$QP_s = \begin{cases} \widehat{QP}_{j,s}^{level} - \delta & \text{if } \widehat{QP}_{j,s}^{level} \geq QP_m \\ \widehat{QP}_{j,s}^{level} + \delta, & \text{otherwise} \end{cases} \quad (3)$$

where $\delta$, $QP_m \in \mathbb{N}$, $s \in \{2160p, 540p\}$, and $level \in \{low, high\}$. The $\delta$s have been selected based on the distributions of ground truth cross-over QPs. In the presented results, $QP_m = 30$, $\delta = 5$ for 2160p and $QP_m = 38$, $\delta = 2$ for 540p.

### C. Evaluation of the Predicted Ladders

We tested on a bitrate range typical for video streaming for the considered resolutions, from 150kbps to 25Mbps. We evaluated the proposed methods by computing the BD metrics (mean and mean absolute deviation (mad)[2] of BD-Rate) against RL, the maximum number of encodes required per method, and the percentage of the estimated ladder points that are identical to RL points, RL-hits.

Fig. 4 illustrates the distributions of BD-Rates for the three tested methods in (a)-(c) and the complexity - accuracy tradeoff in (d). As can be seen in (d), for the same number of initial encodes, CIL-7, slightly improves the mean BD-Rate (0.13%). As the differences between the CIL versions 5-7 are trivial, it is clear that CIL-5 results in the best accuracy-complexity tradeoff, as it reduces by 22% the required encodes compared to NIL. Moreover, FL can further decrease the complexity down to only eight initial encodes at the cost of 0.46% BD-Rate loss. However, by comparing the histograms, the FL distribution tail is heavier than the other two methods. Compared to RL, NIL reduces the maximum number of required encodes by 71% with 75.1% RL-hits, CIL-5 by 77.4% with 74.3% RL-hits, and FL by 87.1% with 36% RL-hits. Taking into account these statistics, we conclude that CIL-5 is the recommended method for an accurate and cost-effective ladder estimation.

In Fig. 5, we show a few examples of the resulting bitrate ladders. As can be seen, in most cases the performances of CIL and NIL are very similar. Although FL builds in most cases a bitrate ladder with PF points, these points are shifted, as also indicated by the RL-hits figure. On average, NIL and CIL are more successful in building ladders with points identical to RL points.

[2]We selected the mean absolute deviation (mad) instead of standard deviation because the BD-Rate distributions are not normal.
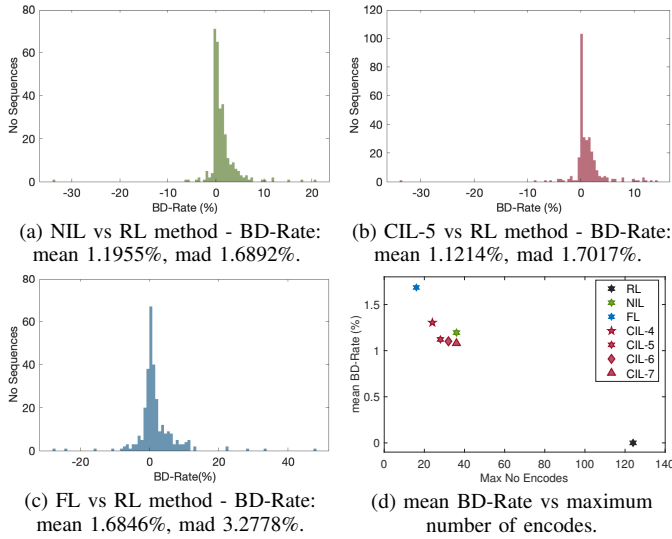
(a) NIL vs RL method - BD-Rate: mean 1.1955%, mad 1.6892%.



(b) CIL-5 vs RL method - BD-Rate: mean 1.1214%, mad 1.7017%.



(c) FL vs RL method - BD-Rate: mean 1.6846%, mad 3.2778%.



(d) mean BD-Rate vs maximum number of encodes.

Fig. 4: The BD-Rate histograms for the compared methods and complexity.



(a) Air-acrobatics.



(b) ConstructionField.
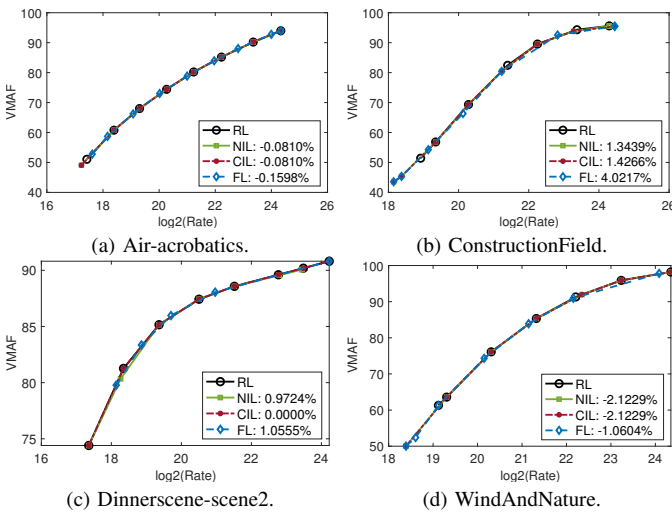


(c) Dinnerscene-scene2.



(d) WindAndNature.

Fig. 5: Predicted ladders for a selection of sequences that represent the distribution of BD-Rates in Fig. 4.

## VI. CONCLUSION

In this paper we proposed a content-driven method that can predict the bitrate ladder for adaptive streaming with significantly reduced complexity. CIL exploits spatio-temporal features extracted from uncompressed video to predict the Rate-VMAF curvature, in order to guide an interpolation based method towards the range of QP values that reside on the PF. The results showed a significant reduction of complexity, 77.4% at a small BD-Rate cost, 1.12%, when compared to the optimal reference ladder, while achieving to build a ladder with 74.3% RL-hits. Concluding, CIL with five initial encodes per resolution offers the best complexity-accuracy tradeoff and is therefore the recommended method for large-scale systems.

## REFERENCES

[1] Bitmovin, "Video Developer Report: COVID-19 and its Impact on OTT Video," https://variety.com/2020/digital/news/coronavirus-covid-ott-streaming-video-impact-webinar-1234584777, 2020.

[2] I. Sodagar, "The mpeg-dash standard for multimedia streaming over the internet," *IEEE MultiMedia*, vol. 18, no. 4, pp. 62–67, 2011.

[3] S. Lederer, C. Müller, and C. Timmerer, "Dynamic Adaptive Streaming over HTTP Dataset," 2012, ACM MMSys, pp. 89–94.

[4] A. Aaron, Z. Li, M. Manohara, J. De Cock, and D. Ronca, "Per-Title Encode Optimization," https://netflixtechblog.com/per-title-encode-optimization-7e99442b62a2.

[5] J. De Cock, Z. Li, M. Manohara, and A. Aaron, "Complexity-based consistent-quality encoding in the cloud," in *IEEE International Conference on Image Processing (ICIP)*, Sept 2016, pp. 1484–1488.

[6] I. Katsavounidis, "Dynamic optimizer - a perceptual video encoding optimization framework," https://medium.com/netflix-techblog/dynamic-optimizer-a-perceptual-video-encoding-optimization-framework-e19f1e3a277f [Online; accessed 2018-03-05].

[7] M. Afonso, A. Moorthy, L. Guo, L. Zhu, and A. Aaron, "Improving our video encodes for legacy devices," https://netflixtechblog.com/improving-our-video-encodes-for-legacy-devices-2b6b56eec5c9.

[8] Z. Li, A. Aaron, I. Katsavounidis, A. Moorthy, and M. Manohara, "The NETFLIX tech blog: Toward a practical perceptual video quality metric," http://techblog.netflix.com/2016/06/toward-practical-perceptual-video.html, [Online; accessed 2018-08-04].

[9] Bitmovin, "White Paper: Per Title Encoding," https://bitmovin.com/whitepapers/Bitmovin-Per-Title.pdf, 2018.

[10] V. P. K. Malladi, C. Timmerer, and H. Hellwagner, "MIPSO: Multi-Period Per-Scene Optimization For HTTP Adaptive Streaming," in *IEEE ICME*, 2020.

[11] Cambria, "Feature: Source Adaptive Bitrate Ladder (SABL)," https://www.capellasystems.net/capella_wp/wp-content/uploads/2018/01/CambriaFTC_SABL.pdf.

[12] K. Goswami, B. Hariharan, P. Ramachandran, A. Giladi, D. Grois, K. Sampath, A. Matheswaran, A. K. Mishra, and K. Pikus, "Adaptive Multi-Resolution Encoding for ABR Streaming," in *IEEE ICIP*, 2018.

[13] MUX, "Instant Per-Title Encoding," https://mux.com/blog/instant-per-title-encoding/.

[14] Y. A. Reznik, K. O. Lillevold, A. Jagannath, J. Greer, and J. Corley, "Optimal design of encoding profiles for abr streaming," in *ACM PV*, 2018.

[15] Y. A. Reznik, X. Li, K. O. Lillevold, A. Jagannath, and J. Greer, "Optimal multi-codec adaptive bitrate streaming," in *IEEE ICME*, 2019.

[16] E. Bourtsoulatze, A. Chadha, I. Fadeev, V. Giotsas, and Y. Andreopoulos, "Deep video precoding," *IEEE TCSVT*, 2019.

[17] A. V. Katsenou, J. Sole, and D. R. Bull, "Content-gnostic Bitrate Ladder Prediction for Adaptive Video Streaming," in *PCS*, 2019.

[18] A. V. Katsenou, J. Sole, and D. R. Bull, "Efficient bitrate ladder construction for content-optimized adaptive video streaming," 2021.

[19] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," Tech. Rep., 13th VCEGM33 Meeting, Austin. TX, 2001.

[20] M. Afonso, F. Zhang, and D. R. Bull, "Spatial resolution adaptation framework for video compression," in *SPIE Optical Engineering + Applications*, 2018, vol. Proceedings Volume 10752, Applications of Digital Image Processing XLI.

[21] M. Afonso, A. Katsenou, F. Zhang, and D. R. Bull, "Collection of 100 4K Sequences," http://angkats.github.io/Dataset-1004K, 2018.

[22] C. E. Duchon, "Lanczos filtering in one and two dimensions," *Journal of Applied Meteorology*, vol. 18, no. 8, pp. 1016–1022, 1979.

[23] V. Satopaa, J. Albrecht, D. Irwin, and B. Raghavan, "Finding a "Kneedle" in a Haystack: Detecting Knee Points in System Behavior," in *ICDCSW*, 2011.

[24] F. N. Fritsch and R. E. Carlson, "Monotone Piecewise Cubic Interpolation," *SIAM Journal on Numerical Analysis*, vol. 17, no. 2, 1980.

[25] M. Kuhn and K. Johnson, *Applied Predictive Modeling, First Edition*, Springer, New York, USA, 2013.

[26] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, MIT Press, 2005.

[27] G. J. Sullivan, J. R. Ohm, W. J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 22, no. 12, 2012.

[28] K. Sharman and K. Sühring, "Common Test Conditions for HM video coding experiments," Tech. Rep., JCTVC-AC1100 of JCT-VC, 2017.

[29] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *IEEE TSMC*, vol. SMC-3, no. 6, 1973.

[30] A. Katsenou, M. Afonso, D. Agrafiotis, and D. R. Bull, "Predicting Video Rate-Distortion Curves using Textural Features," in *PCS*, 2016.

[31] A. V. Katsenou, T. Ntasios, M. Afonso, D. Agrafiotis, and D. R. Bull, "Understanding Video Texture - a Basis for Video Compression," in *MMSP*, 2017.