SMPTE

MI

MOTION IMAGING JOURNAL

TECHNICAL PAPER

KEYWORDS BITRATE LADDER // LIVE STREAMING // RATE-QUALITY CURVES // VIDEO CODING // VIDEO COMPRESSION

E se fosse possível saber a qualidade ideal de um vídeo antes mesmo de codificá-lo? Isso é exatamente o que pesquisadores da Dolby Laboratories acabaram de demonstrar. O estudo desta edição, propõe um modelo de *deep learning* que prediz a curva completa de qualidade (VMAF) de vídeos eliminando a necessidade de dezenas de *encodings* de teste para encontrar os parâmetros ideais.

Os resultados impressionam: Inferência em apenas 0,27 seg por clipe de 2 segundos; Redução significativa de recursos de computação em nuvem; Erro médio de apenas ±7 pontos no score VMAF; Arquitetura baseada na rede SlowFast, capturando informações espaciais e temporais do vídeo.

Para quem trabalha com streaming, compressão de vídeo ou IA aplicada a mídia, este paper é leitura obrigatória. O futuro da entrega de vídeo é *content-driven* — e está acontecendo agora!

Tom Jones Moreira
(tom.jones@youcast.tv.br)

Efficient Content-Driven Encoding Towards a Target Video Quality

By Trisha Mittal, Subhadra Gopalakrishnan, Jaclyn Pytlarz, Robin Atkins, Benjamin Rolling, and Gabe Russell

We propose a novel content-driven workflow that predicts optimal encoding parameters to achieve a target perceptual video quality. We do so by designing a deep learning model that, based on the video input, predicts the entire VMAF rate-distortion curve. Such an approach efficiently reduces the number of encoding attempts, minimizes necessary cloud computing resources, and maximizes perceptual video quality.

Abstract

Video streaming workflows aim to maximize video quality while still maintaining smooth video streaming performance. A traditional fixed bitrate ladder consists of predetermined bitrate-resolution pairs optimized across a wide variety of content. Consequently, these pairs are rarely optimized for a given piece of content. Some encoding tools address this by encoding each piece of video content with many codec parameters and then evaluating the results using a video quality metric. However, this process requires significant computation, which increases cost and encoding time. In this paper we propose a novel content-driven workflow that predicts optimal encoding parameters to achieve a target perceptual video quality. We do so by designing a deep learning model that, based on the video input, predicts a Video Multimethod Assessment Fusion (VMAF) rate-distortion curve. Our results indicate that such a content-driven approach efficiently reduces the number of encoding attempts, minimizes necessary cloud computing resources, encodes most efficiently, and maximizes perceptual video quality.

In recent years, video streaming has emerged as a widely adopted approach for consuming multimedia content. Videos stored on streaming platform servers are compressed and delivered to users based on their display settings, network conditions, and device capabilities while aiming to maintain quality of experience (QoE). However, compressing video for optimized bitrate transmission while preserving high visual quality is challenging, as it can directly impact the user's viewing experience.

HTTP Adaptive Streaming (HAS) has been the widely adopted protocol for video delivery.[1] Traditionally, HAS used a "one-size-fits-all" encoding approach,[2] where a fixed bitrate ladder is employed. Each video is encoded at a set of predetermined bitrate-resolution pairs, independent of the video's specific characteristics or the users' varying network conditions, leading to a suboptimal QoE.[3] An improvement over the fixed bitrate ladder solution in video streaming is to introduce encoding parameters based on the content genre.[4] For instance, sports content, characterized by high motion and fast scene changes, can benefit from higher bitrates. More recently, per-title encoding schemes have been adopted widely across the industry.[5-7] They optimize video quality by selecting the best bitrate-resolution pairs based on the content, ensuring that each resolution performs optimally within a specific bitrate range. This involves encoding video segments at multiple bitrates to exhaustively identify the optimal encoding for the given resolution and then evaluating the result with a video quality prediction tool. While this enhances video delivery, performing this large amount of test encodings and subsequent evaluations is computationally expensive.

This paper proposes a novel content-driven method that predicts the encoding parameters required to achieve a target perceptual video quality. We designed a deep learning model that, based on the video input, predicts a video quality rate distortion. Our results indicate that such a content-driven approach efficiently reduces the number of encoding attempts, minimizes necessary cloud computing resources, encodes most efficiently, and maximizes perceptual video quality.

Background

As mentioned in the previous section, a static bitrate ladder optimized for a wide range of content is the traditional approach that provides the recommended codec parameters for the available bitrate or requested quality. This method does not consider the per-video content/complexity, which is why more advanced techniques have recently been proposed.

Estimating Encoding Parameters for Target Video Quality

Per-title encoding optimization,[8] the technique proposed by Netflix, encodes each source title at different codec parameters (typically quantization and resolution) and measures the resulting bitrate and video quality (as measured by VMAF)[9] to construct the Pareto Front (PF)[10] across all rate-distortion curves. VMAF is a sophisticated video quality metric that combines human vision modeling with machine learning to predict subjective video quality scores. It considers various aspects of video quality, such as sharpness, detail preservation, and artifact visibility, providing a more accurate assessment of perceived video quality compared to traditional metrics like Peak Signal-to-Noise Ratio (PSNR) or Structural Similarity Index Measure (SSIM).[11]

The PF points of a video determine its convex hull, which defines the bitrate ladder. This content-aware, per-title bitrate ladder significantly outperforms the traditional static approach. However, the process involves a large encoding

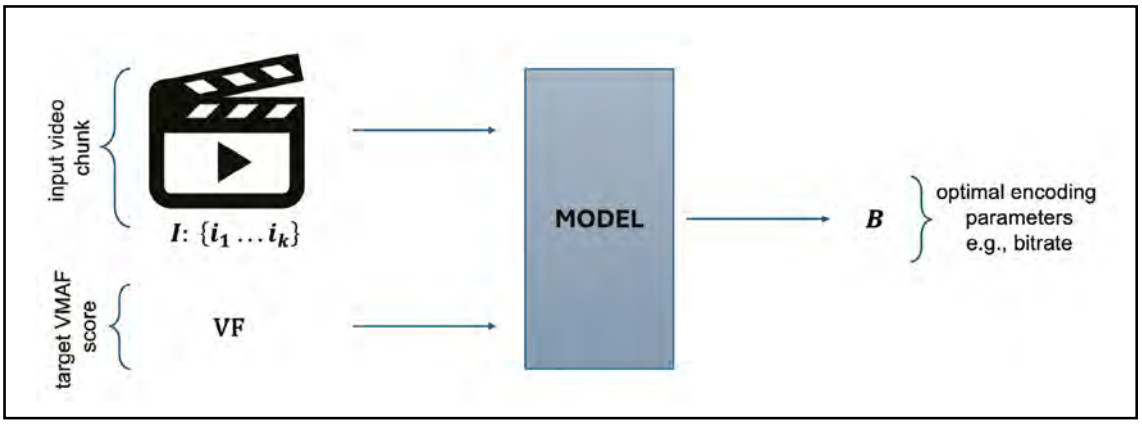

FIGURE 1. Overview of the problem statement.

parameter space, including resolution, quantization parameter, codec type, and preset, resulting in substantial computational complexity to build the convex hull. Early implementations of this approach calculated the bitrate ladder for the entire video, even though a video may contain multiple scenes with varying visual complexities. To take this into account, Netflix proposed an optimized version of their method.[12] In this version, a video is first divided into shots, which are composed of similar adjacent frames that behave similarly when encoding parameters are changed. Then, each shot is processed separately to construct a per-shot bitrate ladder.

Predicting Content-Driven Encoding Parameters

Recently, machine learning has been utilized to reduce the computational cost of content-aware encoding methods.[13-17] These approaches extract hand-crafted frame-level spatio-temporal features from source videos and employ machine-learned models to predict encoding parameters, which are then used to construct bitrate ladders or optimize compression.[18]

Cock et al.[19] introduced a cloud-based encoding approach that selects optimized per-title bitrate ladders by identifying suitable bitrate-resolution pairs for video chunks based on visual complexity using multi-pass video coding via a Constant Rate Factor (CRF). A method based on perceptual video quality using Just Noticeable Difference (JND) and Support Vector Regression (SVR) was proposed in Ref 20. Silhavy et al.[21] applied multiple machine learning algorithms to generate bitrate, resolution, and quality tuples using VMAF.

Problem Statement

As shown in **Fig. 1,** given an input video chunk, $\boldsymbol{I}$ (~2 sec) with $\boldsymbol{k}$ frames, $\{i_1 \dots i_k\}$ and a target VMAF score VF , the goal is to estimate the optimal encoding parameters, such as bitrate $\boldsymbol{B}$. This method avoids generating numerous encodings at various parameters and can, therefore, be an efficient approach to achieving optimal parameters.

Methodology

To build a model that can estimate these encoding parameters, we start by fitting a prior for the VMAF rate-distortion curve. A prior in this context refers to our initial assumption about the shape and behavior of the VMAF rate-distortion curve based on prior knowledge and observations. This acts as our starting point for the estimation process, which can be refined and learned with the availability of more data. Once we have a decent estimate of a prior, we get the curve parameters for the VMAF rate-distortion curve and build a deep learning model to estimate the rate-distortion curve parameters.

Once our model can estimate these curve parameters for a given video chunk, we can perform an easy lookup at a target VMAF to get to the optimal encoding parameters.

Step 1: Defining a prior for VMAF Rate-Distortion curve

Rate-distortion (RD) curves have specific properties that facilitate the selection of an appropriate prior. For instance, RD curves are monotonically increasing in the range of valid bitrate values:

$$\forall x_1, x_2 \in \left[0,\ B_{\max}\right],\quad x_1 < x_2 \Rightarrow f(x_1) \leq f(x_2)$$

where $\boldsymbol{f}$ and $\boldsymbol{B}_{\max}$ correspond to the RD curve and maximum bitrate, respectively.

Moreover, the curve is concave, where the curve's slope is non-increasing in the valid range. At lower bitrate values, the same increase in bitrate corresponds to a larger change in corresponding VMAF than it does for higher bitrate values. The curve also requires the ability to reach saturation for an arbitrary bitrate and above since that can be achievable for certain videos.

The above criteria leads us to the selection of the tanh family of functions. The tanh (hyperbolic tangent) function maps input values to the range (−1,1) with an S-shaped curve, being zero-centered. It is linear near zero but saturates to −1 or 1 for large negative or positive inputs, respectively. We parametrize our curve as follows:

$$\widehat{VF} = \tanh\left(\left(\frac{B}{1000}\right)^{c_2}\right) * c_1$$

where $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$ are the curve parameters. We depict some examples in **Fig. 2** for this curve. It should be noted that for

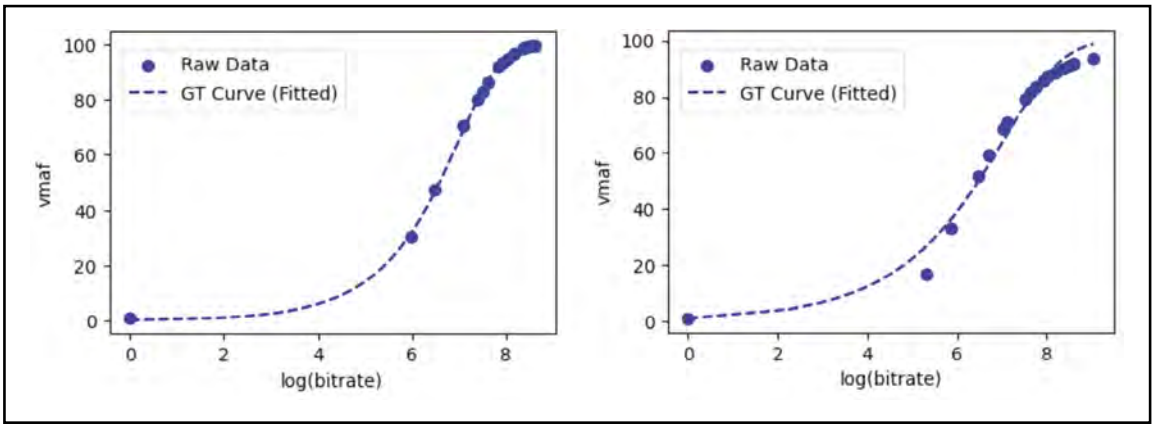

FIGURE 2. Defining a curve prior to the VMAF rate-distortion curve.

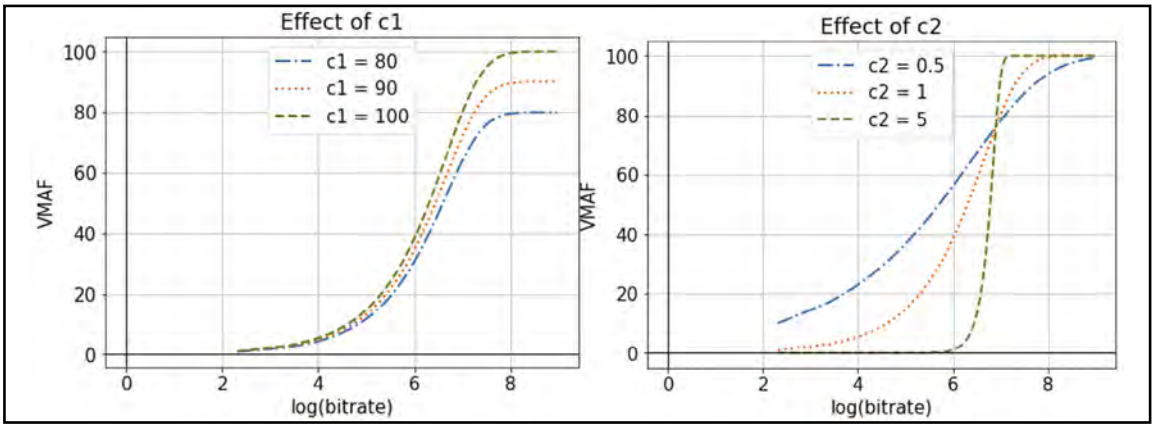

FIGURE 3. Effect of $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$ on the prescribed curve.

learning our curves, we add the VMAF value 0 for a bitrate value of 0. However, we have valid data for bitrates starting at 200 Kbits/s (~ $\log_e 5$.)

Analyzing the curve equation better, we see that $\boldsymbol{c}_1$ controls the point of saturation and $\boldsymbol{c}_2$ controls the knee point, as seen in **Fig. 3**.

It is important to emphasize that VMAF as a metric can be very noisy and does not always show a clean, monotonically increasing behavior as one would expect. So, this parameterization of the data helps stabilize the fitting process.

Step 2: Estimating curve parameters

Dataset: Our dataset comprises 5,336 two-second video chunks from 16 video titles encoded in AVC format using the x264 codec with a resolution of 1080p. Additionally, for each video chunk, we have multiple pairs of VMAF score (VF) and bitrate ($\boldsymbol{B}$). We use these pairs of ($\boldsymbol{B}$, VF) to fit the prior as defined above and get the ground truth curve parameters, $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$. To fit curves and find $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$, we use a non-linear least squares optimization and minimize the sum of the squares of the residuals, which are the differences between the observed data points and the values predicted by the model. In this process, we use the defined prior (a non-linear model) function and provide data points along with initial parameter estimates. The curve-fitting function iteratively adjusts the model's parameters to reduce the sum of squared residuals, leveraging the Gauss-Newton algorithm.[22] The optimization continues until it converges to a set of parameters where the sum of squared residuals is minimized.

Hence, for every video chunk, $\boldsymbol{I}$ with $\boldsymbol{k}$ frames, $\{\boldsymbol{i}_1 \ldots \boldsymbol{i}_k\}$ we have associated curve parameters $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$ obtained from the multiple pairs of target VMAF scores and bitrate values, $\{(\boldsymbol{B}_1,\boldsymbol{VF}_1), \ldots ,(\boldsymbol{B}_m,\boldsymbol{VF}_m)\}$.

Out of the 16 video titles, we used 14 video titles for training and 2 video titles for testing**.**

Training Regime: The model takes as input the $\boldsymbol{k}$ video chunk frames, $\{\boldsymbol{i}_1 \ldots \boldsymbol{i}_k\}$, and predicts the two parameters, $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$ as shown in **Fig. 4**.

We can compare the predicted parameters, $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$ against the actual curve parameters $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$ using the L_2 loss function. This error is used for backpropagating through the model and training the model.

$$error = L_2(\bar{c}_1, c_1) + L_2(\bar{c}_2, c_2)$$

Model Architecture: Our model design is based on the SlowFast network[23] architecture (shown in **Fig. 5**). It was originally designed for video recognition by leveraging two parallel pathways: the slow pathway and the fast pathway. The slow pathway processes video frames at a low frame rate,

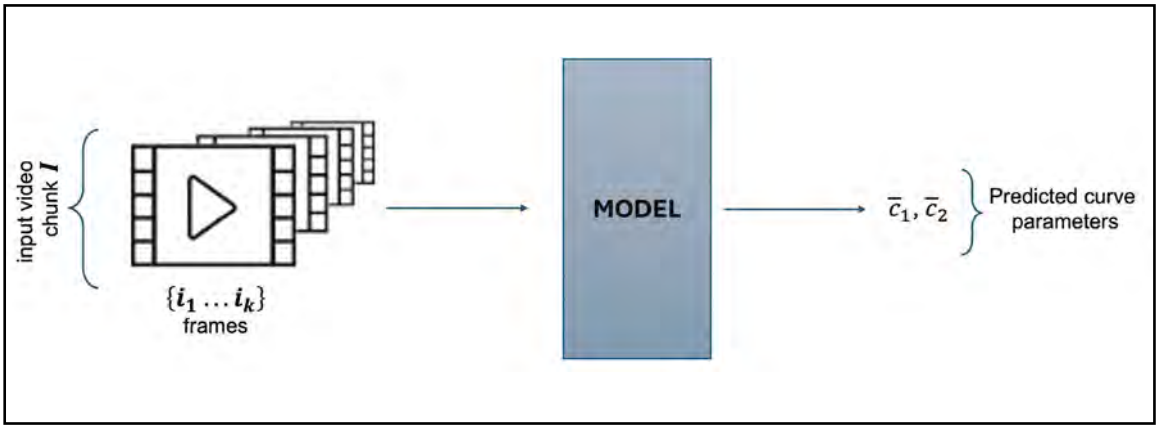

FIGURE 4. Our training regime for predicting the encoding parameters.

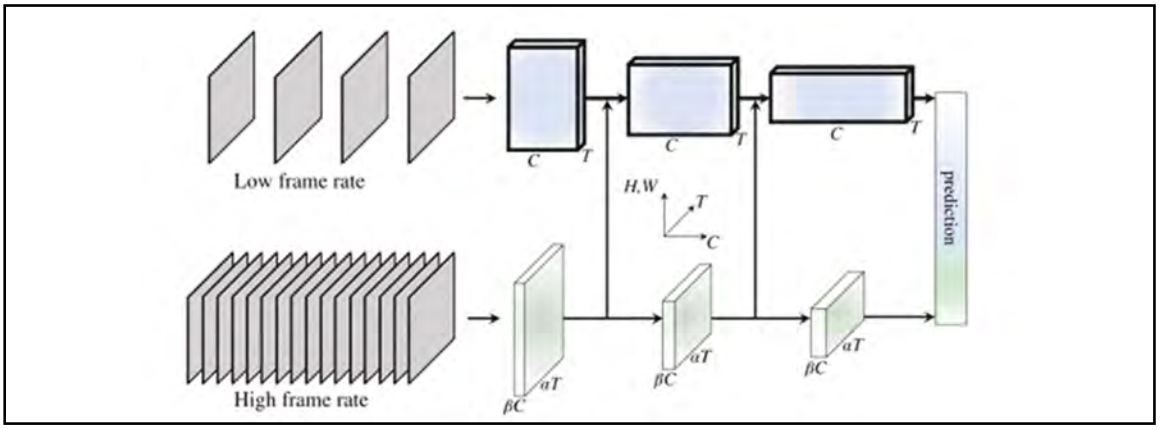

FIGURE 5. Our model backbone is the SlowFast network architecture (borrowed image from Ref. 22.

capturing spatial semantics with feature maps of dimensions $\boldsymbol{A}=\boldsymbol{C}\times\boldsymbol{T}\times\boldsymbol{H}\times\boldsymbol{W}$, where $\boldsymbol{C}$ is the number of channels, $\boldsymbol{T}$ is the temporal dimension, and $\boldsymbol{H}$, $\boldsymbol{W}$ are the spatial dimensions. In contrast, the fast pathway operates at a high frame rate, capturing rapid temporal information with reduced complexity (b$\boldsymbol{C}$ channels) and increased temporal resolution (a$\boldsymbol{T}$ frames), where b is a reduction factor for the number of channels and a is a factor for the increase in the number of frames. The outputs from these pathways are fused at multiple stages to combine spatial and temporal information effectively, resulting in a robust video recognition performance. This architecture allows the model to capture detailed motion patterns while maintaining computational efficiency.

Because SlowFast networks capture both slow and fast temporal dynamics in videos through their dual-pathway design, integrating long-term and short-term temporal information with high-resolution spatial details, this comprehensive processing capability aligns with both the encoder's temporal efficiency and VMAF's need to evaluate spatial features. This makes SlowFast a highly effective model for predicting perceptual video quality as measured by VMAF.

Experiments and Results

Implementation Details: In the SlowFast network architecture, the Slow Pathway, which focuses on learning the spatial information, takes as input only three frames (first, middle, last), which are resized 448 × 448. For the Fast Pathway, where the focus is to learn the temporal information from the video, we send every alternate frame, resized to 56 × 56. The model was trained for 50 epochs using exponential degradation using Adam optimizer.[24]

Computational Efficiency: When benchmarked, this model on CPU at inference time performs at 0.27 sec per 'two-second' video segment.

Experimental Results: We used 2-sec clips from 14 titles (~4232 clips) for training and the remaining (~1004 clips) for testing purposes. On average, we witnessed an error of ± 7 in the VMAF score. In **Fig. 6**, we show some results of our model's prediction. The red points and the red curve denote the predicted curve from the model. Cyan blue denotes the ground truth VMAF values for the corresponding bitrate values. The darker blue points and the darker blue curve denote the curve, $\left(\left(\frac{R}{1000}\right)^{c_2}\right) * c_1$.

The curve on the left corresponds to a photo-realistic outdoor nature scene clip, and the curve on the right corresponds to a scene clip from the opening credit, which is mostly sparse, with little text on the screen.

In **Fig. 7**, we analyzed the model predictions corresponding to the two clips in **Fig. 6** (left vs. left, right vs. right). On the y-axis, we plotted the error between the ground truth and

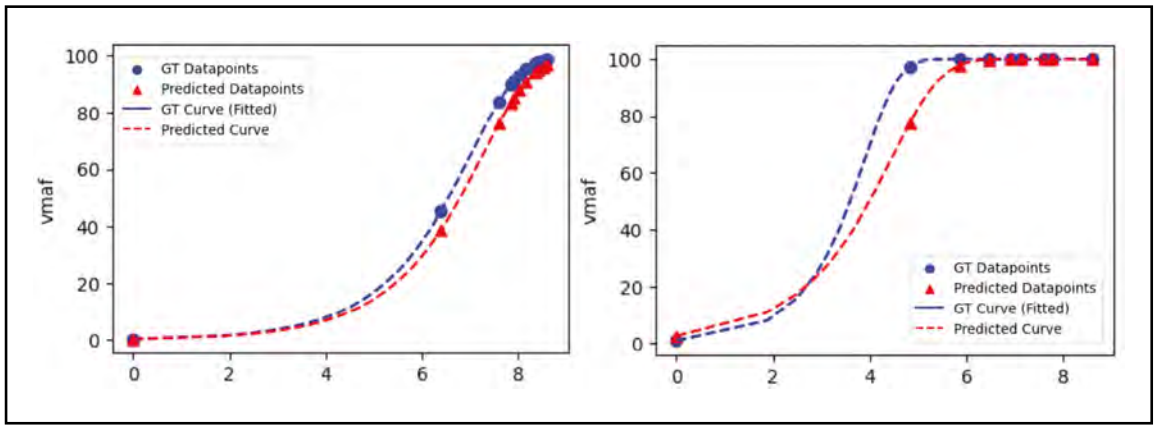

FIGURE 6. Sample ground truth and predicted curves from our models for two clips from our test set.

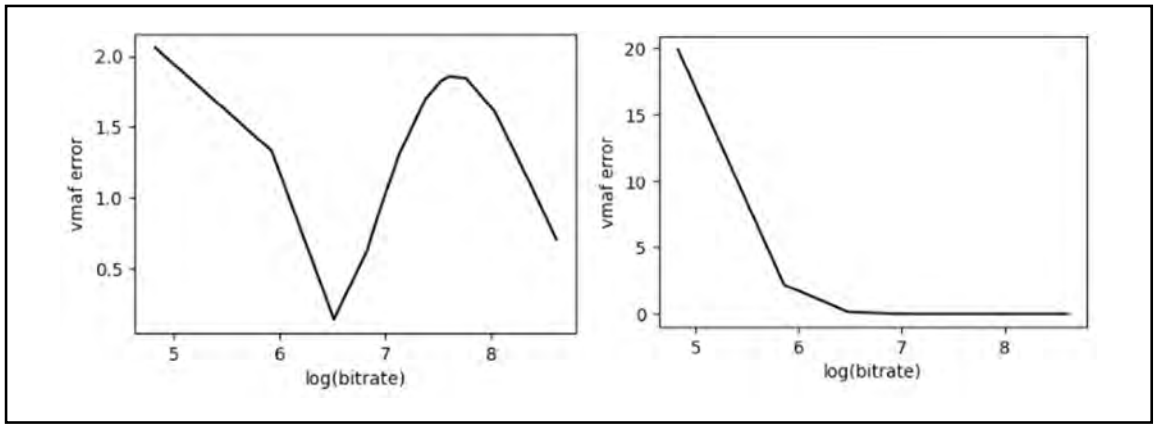

FIGURE 7. VMAF error for the clips from Fig. 6 corresponding to the bitrate range.

predicted VMAF, and the x-axis is the bitrate range. From the plots, we observed higher VMAF errors at lower bitrates.

While our model can predict the curve parameters reasonably well, on further analysis, we observe that the model predictions can be significantly off when certain high-motion aspects like fire and smoke are present in the video. We depict one such example in **Fig. 8.**

Another point to highlight that can also be seen in the plot in **Fig. 8** is the noisy nature of the VMAF score. It can be seen from the green points that around bitrate 2000 Kbits/s, we notice a higher VMAF score, which dips with increasing bitrates. It is important to emphasize that the model needs to be robust to such noise.

Ablation Experiments: We also performed two ablation experiments (removal of components in the model to understand their contribution to the overall system). One experiment was to see the impact of using variable resolution frames for the Slow Pathway. We experimented with the first, middle, and last frames at 56 × 56 pixels each, 112 × 112 pixels, 224 × 224 pixels, 448 × 448 pixels, and 896 × 896 pixels each. We observed that the model performance increases until 448 × 448 and saturates after that.

The other experiment was the choice of the three frames for the Slow Pathway and the choice of 24 frames for the Fast Pathway. For choosing the three frames, we ran an experiment by selecting the three most distinct frames out of the 48 frames in the two-second clip for the Slow pathway and choosing 24 frames out of the 48 frames that have the most distinction consecutively. We did not observe a significant performance increase with such experiments.

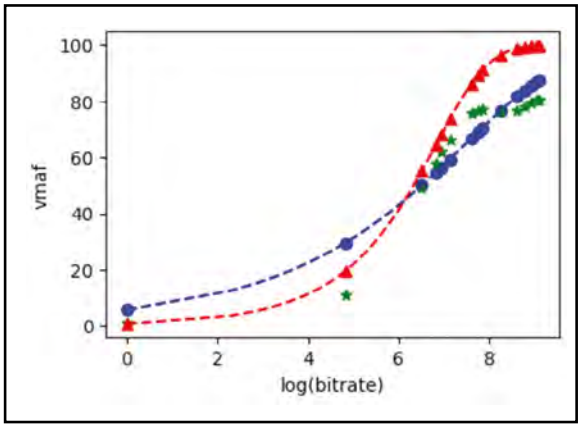

FIGURE 8. An example of a failure case of our model.

Conclusion

In this paper, we propose a novel content-driven method that predicts the VMAF rate-distortion curves. We designed a

deep learning model that, based on the video input, predicts a video quality rate-distortion curve. Our results indicate that such a content-driven approach is an efficient way to reduce the number of encoding attempts, minimize necessary computing resources, and maximize perceptual video quality.

For now, the model only accounts for the quantization parameter as the codec parameter that can be adjusted. With further work, we would like to explore the scalability of our approach to variable-resolution input videos. As next steps, it would be important to consider target factors beyond objective video quality metrics such as viewing distance and screen size. The current model only predicts the bitrate for a chunk size of 2 sec. A scene detection model can be used to ensure the bitrates remain consistent across various chunks of the same scene to ensure a smooth viewing experience. Furthermore, there is potential in designing a better curve equation. The parameter c_1 currently converges to 100, and in practice, this might be worth revisiting (See **Fig. 3**). Another follow-up step is to evaluate our proposed model via subjective tests to assess video quality.

References

1. Rafael Huysegems et al. "HTTP/2-Based Methods to Improve the Live Experience of Adaptive Streaming," *Proc. of the 23rd ACM International Conference on Multimedia, pp. 541-550. Oct.* 2015.
2. "How Netflix Pioneered Per-Title Video Encoding Optimization," *Streaming Media*, Feb. 3, 2016. [Online]. Available: https://www.streamingmedia.com/Articles/Editorial/Featured-Articles/How-Netflix-Pioneered-Per-Title-Video-Encoding-Optimization-108547.aspx. [Accessed: Dec. 10, 2024].
3. Ahmed Telili, et al. "Bitrate Ladder Prediction Methods for Adaptive Video Streaming: A Review and Benchmark." *arXiv preprint arXiv:2310.15163* (2023).
4. Stefan Lederer, Christopher Müller, and Christian Timmerer. "Dynamic adaptive streaming over HTTP dataset," *Proc. of the 3rd Multimedia Systems Conference*, 2012, pp. 89–94, Feb. 2012.
5. J. De Cock, Li Z, M. Manohara, A. Aaron, "Complexity-based Consistent-quality Encoding in the Cloud," *Proc. 2016 IEEE International Conference on Image Processing (ICIP) 2016*, pp. 1484-1488, Sep. 2016.
6. C. Timmerer, H. Hellwagner, "Mipso: Multi-period per-scene optimization for HTTP adaptive streaming," *Proc. 2020 IEEE International Conference on Multimedia and Expo (ICME) 2020,* pp. 1-6 Jul. 2020.
7. H. Amirpour, C. Timmerer, M. Ghanbari, "PSTR: Per-Title Encoding Using Spatio-Temporal Resolutions," *Proc. 2021 IEEE International Conference on Multimedia and Expo (ICME) 2021,* pp. 1-6, Jul. 2021.
8. "Per-Title Encode Optimization," *Netflix Tech Blog*, Dec. 14, 2015. [Online]. Available: https://netflixtechblog.com/per-title-encode-optimization-7e99442b62a2. [Accessed: Dec. 10, 2024].
9. "VMAF: The Journey Continues," *Netflix Tech Blog*, June 5, 2018. [Online]. Available: https://netflixtechblog.com/vmaf-the-journey-continues-44b51ee9ed12. [Accessed: Dec. 10, 2024].
10. Vilfredo Pareto, *Manuel d'économie politique*. Vol. 38. Giard & Brière, 1909.
11. Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600-612, Apr. 2004.
12. "Dynamic Optimizer: A Perceptual Video Encoding Optimization Framework," *Netflix Tech Blog*, Apr. 9, 2019. [Online]. Available: https://netflixtechblog.com/dynamic-optimizer-a-perceptual-video-encoding-optimization-framework-e19f1e3a277f. [Accessed: Dec. 10, Sep. 2024].
13. J. Šuljug, S. Rimac-Drlje, "Fast Content Adaptive Representation Selection in HEVC-Based Video Coding for Streaming Applications," *Proc. Elmar - International Symposium Electronics in Marine*. pp. 169-74, Sep. 2022.
14. M. Bhat, J. M. Thiesse, P. Le Callet, "A case study of machine learning classifiers for real-time adaptive resolution prediction in video coding," *Proc. IEEE International Conference on Multimedia and Expo.* pp. 1-6, Jul. 2020.
15. V. V. Menon, H. Amirpour, M. Ghanbari, C. Timmerer, "Perceptually-Aware Per-Title Encoding for Adaptive Video Streaming," *Proc. IEEE International Conference on Multimedia and Expo*, pp. 1-6, 2022.
16. F. Nasiri, W. Hamidouche, L. Morin, N. Dholland, J.Y Aubie, „Ensemble Learning for Efficient VVC Bitrate Ladder Prediction," Proc. the European Workshop on Visual Information Processing (EUVIP). 2022. [Online]. Available: http://arxiv.org/abs/2207.10317
17. A. V. Katsenou, J. Sole, D. Bull. "Efficient Bitrate Ladder Construction for Content-Optimized Adaptive Video Streaming," *IEEE Open J Signal Process*, 2:496-511, Dec. 2021.
18. A. Elahi, A. Falahati, F. Pakdaman, M. Modarressi, M. Gabbouj, "NCOD: Near-Optimum Video Compression for Object Detection," *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1-5, May 2023.
19. J. De Cock, Z. Li, M. Manohara, and A. Aaron, "Complexity-based consistent-quality encoding in the cloud," *Proc. 2016 IEEE International Conference on Image Processing (ICIP),* pp. 1484–1488, Sep. 2016.
20. M. Takeuchi, S. Saika, Y. Sakamoto, T. Nagashima, Z. Cheng, K. Kanai, J. Katto, K. Wei, J. Zengwei, and X. Wei, "Perceptual Quality Driven Adaptive Video Coding Using JND Estimation," *Proc. 2018 Picture Coding Symposium (PCS)*, pp. 179-183, May 2018.
21. D. Silhavy, C. Krauss, A. Chen, A.-T. Nguyen, C. Müller, S. Arbanowski, S. Steglich, and L. Bassbouss. "Machine Learning for Per-Title Encoding." *SMPTE Mot. Imag. J.*, 131(3): 42–50, May 2022.
22. James V. Burke and Michael C. Ferris, "A Gauss-Newton Method for Convex Composite Optimization," *Mathematical Programming*, 71(2): 179–194, Nov. 1995.
23. C. Feichtenhofer, H. Fan, J. Malik, K. He, "SlowFast Networks for Video Recognition," *Proc. of the IEEE/CVF International Conference on Computer Vision (ICCV),* pp. 6202-6211, Oct. 2019.
24. D. P. Kingma, J. Ba. "Adam: A Method for Stochastic Optimization." [Online]. Available: *arXiv preprint arXiv*:1412.6980, Dec. 2014.

About the Authors

Trisha Mittal is a senior researcher at Dolby Laboratories, CA. Her research spans developing algorithms using artificial intelligence, machine learning, deep learning and computer vision for various algorithms including affective computing.

Subhadra Gopalakrishnan is a computer vision engineer at Apple. Previously, she was part of Dolby's Vision Science Research Group as an applied scientist working on deep learning algorithms focused on image and video enhancement.

Jaclyn Pytlarz is a senior staff researcher at Dolby Laboratories. She leads Dolby's Vision Science research team where she applies human perception research to the media entertainment industry.

Robin Atkins is a principal researcher at Dolby Laboratories, where leads a team of researchers responsible for innovating new visual experiences and inventing video processing algorithms.

Ben Rolling has built a career in online media technologies and their applications, focusing on implementing emerging innovations. He began by managing live-streaming events, including award shows and concerts, and advanced to enhancing these experiences by directly integrating time-aligned event data into the presentations.

Gabe Russell is a senior product manager at Dolby Laboratories, focused on cloud-based media processing workflows for Dolby Hybrik. With a background in content production and video engineering, he began working on players for large-scale interactive live events.