Benchmarking Learning-based Bitrate Ladder Prediction Methods for Adaptive Video Streaming

Ahmed Telili, Wassim Hamidouche, Sid Ahmed Fezza, Luce Morin

Benchmarking Learning-based Bitrate Ladder Prediction Methods for Adaptive Video Streaming

Ahmed Telili[1], Wassim Hamidouche[1], Sid Ahmed Fezza[2] and Luce Morin[1]
[1]Univ. Rennes, INSA Rennes, CNRS, IETR - UMR 6164, Rennes, France
[2]National Higher School of Telecommunications and ICT, Oran, Algeria

***Abstract*—HTTP adaptive streaming (HAS) is increasingly adopted by over-the-top (OTT)-based video streaming services, it allows clients to dynamically switch among various stream representations. Each of these representations is encoded to target a specific bitrate providing a wide range of operating bitrates known as the bitrate ladder. Several approaches with different levels of complexity are currently used to build such a bitrate ladder. The most straightforward method is to use a fixed bitrate ladder for all videos, which is a set of bitrate-resolution pairs, called "one-size-fits-all", and the most complex is based on the intensive encoding of all resolutions over a wide bitrate range to construct the convex-hull. This latter is then used to obtain a per-title bitrate ladder. Recently, various methods relying on machine learning (ML) techniques have been proposed to predict content-based ladder without performing exhaustive search encoding. In this paper, we conduct a benchmark study of several handcrafted- and deep learning (DL)-based approaches for predicting content-optimized bitrate ladder, which we believe provides baseline methods and will be useful for future research in this field. The obtained results, based on 200 video sequences compressed with the high-efficiency video coding (HEVC) encoder, reveal that the most efficient method predicts the bitrate ladder without performing any encoding process at the cost of a slight Bjøntegaard delta bitrate (BD-BR) loss of 1.43% compared to the exhaustive approach. The dataset and the source code of the considered methods are made publicly available at: https://github.com/atelili/Bitrate-Ladder-Benchmark.**

***Index Terms*—Bitrate ladder, video compression, HEVC, rate-quality curves, adaptive video streaming.**

I. Introduction

Today, It is obvious that video dominates the Internet. According to Cisco's report [1], video content represents today around 82% of the global Internet traffic. Moreover, the global video on demand (VOD) market is expected to grow from USD 473.39 billion in 2022 to USD 1,690.35 billion by 2029 [2]. Thus, video providers invest a significant amount of resources to optimize the video encoding process before transmission targeting higher quality of experience (QoE) and sustainable video streaming.

The quality of content provided may differ from one client to another and may be influenced by various factors such as network bandwidth, display resolution, and viewing conditions. To deliver videos with the highest possible visual quality at the lowest possible bitrate, streaming service providers rely on various state-of-the-art video streaming technologies and standards, such as HTTP adaptive streaming (HAS) [3]. In HAS, video content is divided into short segments of 2s to 10s. Each segment is pre-encoded at different resolutions and quality levels on the server-side, before being transmitted over HTTP to a client device with specific bandwidth, display resolution and computing resources. HTTP live streaming (HLS) introduced by Apple [4] and dynamic adaptive streaming over HTTP (DASH) developed by MPEG [5] are the two main HAS specifications. In both, the methods used to calculate an adequate bitrate ladder play a critical role in HAS.

The most traditional and straightforward method for selecting bitrate-resolution pairs relies on a static bitrate ladder for all videos, often referred to as *one-size-fits-all*. In such a method, predefined pairs of bitrates and resolutions for all videos are specified, regardless of their content or visual complexity[1]. One of the commonly adopted static bitrate ladder is that proposed by Apple in Tech Note TN2224 [7]. However, such a method does not allow adaptation to video content, which can vary considerably in motion, texture and scene complexity. Therefore, a fixed bitrate ladder is not suitable for all content types and cannot deliver the best video quality for a given title at the lowest possible bitrate. Hence, as a consequence of this, any content-gnostic method is suboptimal. A more advanced technique proposed by Netflix, called per-title encoding optimization [8], encodes each source title at different bitrates and resolutions in order to build the Pareto front (PF) across all rate-distortion curves using video multi-method assessment fusion (VMAF) [9] as a perceptual quality metric. The PF points of a video determine its convex hull, which describes the bitrate ladder, as illustrated in Fig. 1. This content-aware per-title bitrate ladder outperforms the static approach. However, given the large encoding parameter space (resolution, quantization parameter, codec type and preset) a huge computational complexity is required to build the convex hull. Typically, such an encoding process is done in the cloud. In addition, this method calculates the bitrate ladder for the entire video, while a given video may contains several scenes of different visual complexity. It is therefore more appropriate to split the video into scenes based on its content properties and then process them separately. ,To address these issues, several machine learning (ML) based approaches have been developed recently by actors in academia [10]–[12] and industry [13] communities to predict a per-scene or per-shot bitrate ladder without the need for an exhaustive encoding. Here we seek to address and investigate the content-gnostic

This work has been supported by Région Bretagne under the DEEPTEC project.

[1]Visual complexity is conventionally defined as the level of detail or intricacy contained within an image [6].

bitrate ladder prediction challenge by conducting a benchmark study of learning-based prediction methods. We summarize our contributions as follows:

- Building of a ground truth bitrate ladder with 200 video sequences encoded with high-efficiency video coding (HEVC) encoder at 4 resolutions and 31 quantization parameter (QP) values.
- Exploring several hand-crafted features with various ML models for bitrate ladder prediction.
- Designing of a bitrate ladder prediction method with pretrained deep learning (DL) models as feature extraction backbones followed by two long short term memory (LSTM) for both spatial and temporal pooling.
- Conducting a comprehensive benchmark study to assess the performance of ML-based methods for predicting bitrate ladder.

The remainder of this paper is organized as follows. Section II reviews related work, then Section III presents the development of benchmarking evaluation. Section IV provides and analyses the experimental results. Finally, Section V concludes this paper.

II. Related work

As explained in the previous section, the static bitrate ladder is the conventional approach that provides the recommended spatial resolution for the available bitrate or requested quality. This method does not take video content/complexity into account, which is why more advanced techniques have recently been proposed.

A per-title method was first proposed by Netflix [8], where the main task is to encode the whole video at different bitrates and resolutions, in order to build the convex hull of the rate-distortion curves based on VMAF. Since this method is suboptimal, given that a video can contain different scenes with different visual complexity, Netflix research team has proposed an optimised version of their method [14]. In this latter, a video is first divided into shots, which are composed of similar adjacent frames that behave similarly when encoding parameters are changed, and then each shot is processed separately to construct a per-shot bitrate ladder.

Cock *et al.* [15] introduced a cloud-based encoding approach that selects optimized per-title bitrate ladders by identifying the most suitable bitrate-resolution pairs for all video chunks based on their visual complexity. This was done using a multi-pass video coding via a constant rate factor (CRF). In [16], the authors proposed a video encoding method based on perceptual quality of the video using just noticeable difference (JND). This solution uses support vector regression (SVR) model as a JND scale estimator and a pre-encoder to generate an encoding recipe with a constant JND interval.

Angeliki *et al.* [10] proposed a content-gnostic approach that can predict the bitrate ladder using Gaussian processes regression (GPR) with rational quadratic kernels and handcrafted spatio-temporal features extracted from the uncompressed video in their native resolutions. Similar to [10], Silhavy *et al.* [11] applied several ML algorithms such as SVR, multilayer perceptron and random forest regression (RFR) to generate a tuple of bitrate, resolution, and quality using VMAF to determine the convex hull.

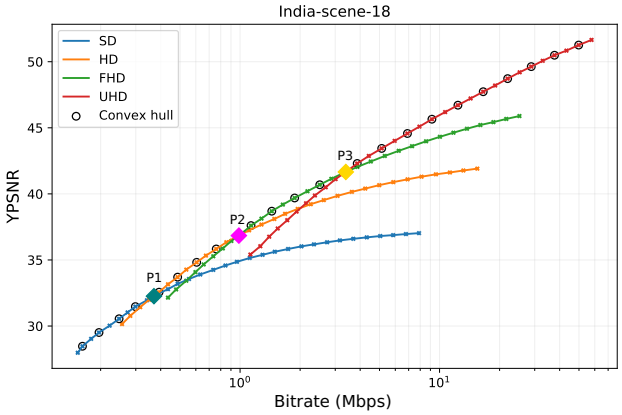


Fig. 1: Example of rate-distortion curves and convex hull construction for the sequence of India-scene-18. P1, P2 and P3 denote the intersection points between SD-HD, HD-FHD and FHD-UHD resolutions, respectively.

Bitmovin proposed a technique [17] that includes a complexity analysis step of the input video, then the results of this analysis are fed into a ML model to adapt the encoding profile to match the content complexity characteristics. Another technique proposed by CAMBRIA is based on encoding complexity estimation by running a fast CRF encoding [18]. MUX [13] proposed a DL-based approach that predicts the bitrate ladder using a multi-layer neural network. Although all the industrial techniques listed below are of interest, it is not possible to provide more details as they are not publicly available.

III. Benchmark description for bitrate ladder construction

This work is driven by the remarkable success of learning-based techniques for image and video coding optimization [19]. As discussed in Section II, several approaches based on handcrafted features were exploited, however, a few DL-based methods have been proposed mainly due to the lack of large-scale labeled datasets for this topic. Consequently, in this paper, we first build a database of rate-distortion convex hulls composed of 200 video shots encoded by HEVC, then we conduct a comprehensive benchmarking study of ML-based methods to assess their performance and reliability. Figure 2 shows a high-level flowchart of the proposed benchmark for both handcrafted feature-based and DL-based models.

A. Database construction

Database: It is essential for the current study to have a large video database covering a variety of scenes. For this purpose, we used the dataset proposed by Angeliki *et al.* [10]. It contains 100 ultra high definition (UHD) video shots collected from different publicly available sources. Each sequence is composed of 64 frames at 60 fps. While this may be sufficient for training models based on handcrafted features,

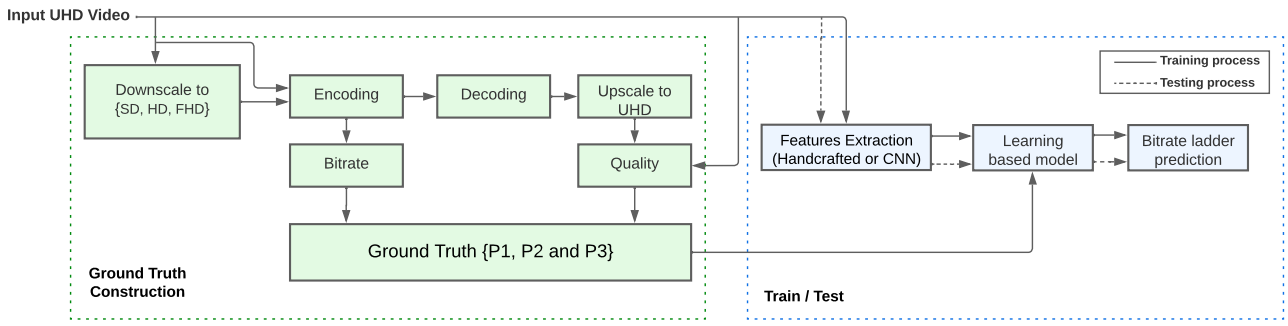


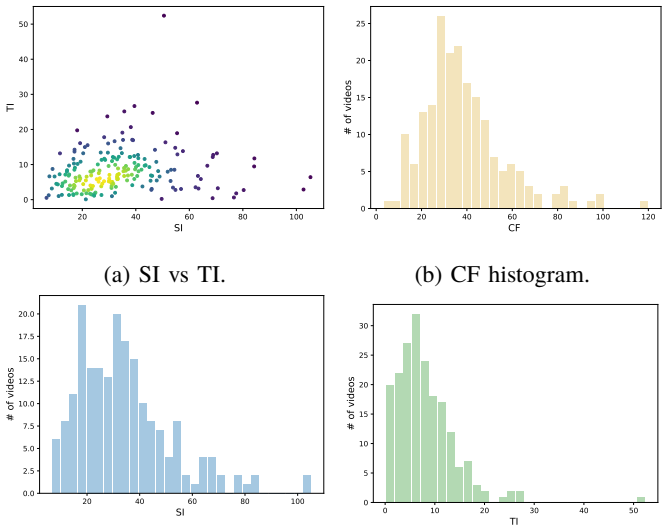
Fig. 2: High-level flowchart of the proposed benchmark, including the ground truth construction, train and test process.

it is not suitable for training a deep neural network. Thus, we collected additional 100 UHD video sequences: 20 pristine uncompressed videos from Waterloo IVC 4K Video Quality Database [20] and 80 high quality videos from YouTube-UGC database [21]. All sequences were split into scenes to ensure that each sequence contains a single scene using PySceneDetect tool [22]. Finally, the sequences were temporally cropped to 64 frames at 60fps to match the first dataset. Figure 3 reflects the diversity of videos content in terms of three basic descriptors: spatial information (SI), temporal information (TI) and colorfulness (CF) of the considered dataset. This figure shows a wide spatial, color and temporal coverage of the dataset.

Convex hull construction: To construct the ground truth and identify the crossed bitrates for the different resolutions, we spatially downscaled all the sequences using Lanczos-3 filter implemented by FFmpeg [23] at three resolutions: FHD, HD and SD. Next, we encoded the four resolutions of each video sequence with the x265 software HEVC encoder in Random Access configuration using QP values in the range: $QP = \{15, \ldots, 45\}$. In total, we generated 24800 different HEVC encoded video shots. Subsequently, all the encoded bitstreams were decoded and upsampled to the native resolution ($2160p$) using the same filter. Next, we compute the objective video quality of the decoded sequence using two full reference quality metrics including YPSNR and VMAF. Finally, the intersection points between rate-distortion curves of the same video across resolutions were defined to construct the convex hull. We define three intersection points SD-HD, HD-FHD and FHD-UHD as P1, P2 and P3, respectively, as illustrated in Fig. 1.

B. Methods based on handcrafted features

The main step in building a model based on handcrafted features is to select the most relevant features. In our study, we started by building an initial set of features that have been successfully used in compression-related research [24]. Specifically, as spatial features, we used the gray level co-occurrence matrix (GLCM) and extracted its basic features: contrast, correlation, homogeneity, energy and entropy. Moreover, CF, SI and estimated noise were extracted for the same purpose. On the other hand, we consider temporal coherence (TC) with its interframe statistics (mean, standard deviation,



(a) SI vs TI. (b) CF histogram.

(c) SI histogram. (d) TI histogram

Fig. 3: Distributions of SI, TI, and CF descriptors of the considered dataset.

kurtosis, skewness and entropy), TI and normalized cross correlation (NCC) as temporal features. In addition, given the strong correlation between the intersection points: P3, P2 and P1, we therefore used the predicted cross-over bitrates as features.

To reduce the number of features and keep only the most significant ones, we used two types of feature selection algorithms. The first type is model-based feature selectors, from which we considered RFR and SVR to fit a regression model in order to eliminate the less relevant descriptors. The other technique is to apply recursive feature elimination (RFE). In this study, we used the ExtraTrees as a target regressor. We applied these feature selection methods on 10 test-training iterations using stratified sampling. Figure 4 illustrates the median Pearson linear correlation coefficient (PLCC) performance for predicting P3 in terms of YPSNR. The selected features of predicted cross-over bitrates are given in Table II for both YPSNR and VMAF RD curves.

C. Methods based on deep neural network

Recently, deep convolutional neural networks (CNNs) have shown outstanding performance in a wide range of computer

TABLE I: List of features and their statistics.

Features	Statistics
Grey-Level Co-occurrence Matrix (GLCM)	F1.meanGLCM$_{con}$, F2.stdGLCM$_{con}$, F3.meanGLCM$_{cor}$, F4stdGLCM$_{cor}$, F5.meanGLCM$_{hom}$, F6.stdGLCM$_{hom}$, F7.meanGLCM$_{enr}$, F8.stdGLCM$_{enr}$, F9.meanGLCM$_{ent}$, F10.stdGLCM$_{ent}$
Temporal Coherence (TC)	F11.meanTC$_{mean}$, F12.meanTC$_{std}$, F13.stdTC$_{mean}$, F14.stdTC$_{std}$, F15.meanTC$_{skw}$, F16.stdTC$_{skw}$, F17.meanTC$_{kur}$, F18.stdTC$_{kur}$, F19.meanTC$_{entr}$, F20.stdTC$_{entr}$
Spatial Information (SI)	F21.meanSI, F22.stdSI
Temporal Information (TI)	F23.meanSI, F24.stdTI
Colorfulness (CF)	F25.meanCF, F26.stdCF
Noise	F27.meanNoise, F28.std Noise
Normalized Cross Correlation (NCC)	F29.meanNCC, F30.stdNCC
Predicted cross-over bitrates	F31.P3, F32.P2

TABLE II: Selected features of predicted cross-over bitrates for RD curves based on YPSNR and VMAF.

Intersection points	YPSNR	VMAF
P3	F5, F11, F12, F13, F21, F22, F27, F29	F5, F7, F11, F12, F15, F16, F22, F25, F27
P2	F10, F11, F13, F31	F10, F11, F13, F26, F31
P1	F1, F3, F4, F5, F7, F21, F22, F31, F32	F5, F11, F21, F22, F24, F26, F27, F31, F32

vision tasks thanks to their powerful feature extraction ability. However, due to their inherent complexity, massive amounts of data are required to train CNN from scratch. For this reason, we used pretrained models on ImageNet [25] as backbone models to serve as deep feature descriptors.

From each sequence in the train set, 16 frames are extracted before applying a sliding window on each frame to extract 156 patches of size 224×224, since downscaling the input frame will alter its quality. Each patch is used as input to the CNN to extract the discriminating features. After the feature extraction process, the features are fed into two LSTM models to capture both long-range dependencies among image patches and long-range dependencies among frames. The training is performed for 200 epochs, with an initial learning rate of $1e-4$ and the mean squared error (MSE) as the loss function.

IV. Experimental results

A. Experimental setup

As there are no publicly available implementations of most of the approaches listed in Section II, we evaluated the proposed models against three alternative methods outlined below:

- Ground truth (GT) ladder: this approach is based on exhaustive encoding as described in Section III. This solution generates the optimal bitrate ladder, however, it significantly increases the processing time and cost.

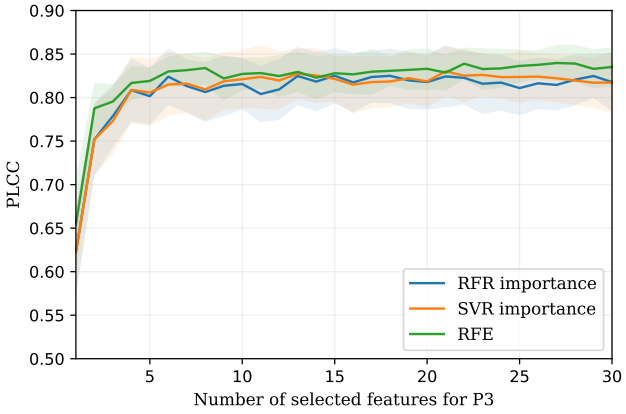


Fig. 4: Feature selection performance (PLCC) of the three selected algorithms for P3 cross-point prediction in terms of YPSNR. The shaded error bar represents the standard deviation of PLCC over 10 iterations.

- Apple ladder (AL): static bitrate ladder proposed by Apple in [4].
- Reference ladder (RL): static bitrate resolution pairs are obtained by averaging the ground truth bitrate ladder of the training dataset.

To assess the performance of each model, we used three correlation metrics: R-squared (R2), Spearman rank order correlation coefficient (SROCC) and Pearson linear correlation coefficient (PLCC) to measure the relationship between the predicted values and the ground truth. We also used the accuracy to evaluate the performance of each approach in predicting the optimal resolution over all tested bitrates. Finally, we computed the Bjøntegaard delta bitrate (BD-BR) score, where for two bitrate ladders, one ladder is chosen as the "anchor" while the other as the "test".

B. Results and analyses

In this study, we trained and tested four classic ML regression models: Gaussian Process, XGBoost, Random Forest and ExtraTrees Regressor, as well as well-known deep CNN models such as ResNet-50, VGG16, Densenet169 and EfficientNet B7. Table III presents the median performance of these different models over 10 iterations. The first observation is that the methods based on handcrafted features perform better than deep neural network methods, which may be explained by the amount of data needed to train them. In fact, a dataset with 200 video sequences is still insufficient to train DL networks to outperform models based on handcrafted features. However, it is clear that they exceed the performance of static approaches enabling an average BD-BR gain of 15.68% and an average accuracy in predicting cross-over bitrates of 82% versus 53% for AL approach in terms of YPSNR. It should be noted that negative values of the BD-BR metric indicate a bitrate saving for the same quality level. Also, we can notice that ExtraTrees Regressor achieves the best results on the test dataset, enabling gains of 18.42%/18.82% and 9.02%/8.79% versus AL and RL approaches, respectively, in terms of YPSNR/VMAF. The histogram of BD-BR per sequence compared to AL and GT is

TABLE III: Performance comparison of evaluated models on the proposed dataset. The top result is highlighted in **boldfaced**.

Quality metric	YPSNR / VMAF						
Model \Metric	R2 ↑	SROCC ↑	PLCC ↑	Accuracy ↑	BD-BR vs GT ↓	BD-BR vs AL ↓	BD-BR vs RL ↓
ExtraTrees Regressor♦	**0.7635 / 0.6420**	**0.8174 / 0.6635**	**0.9000 / 0.8277**	**0.8779 / 0.8400**	**1.433% / 2.704%**	**-18.427%** / -18.827%	**-9.025% / -8.798%**
XGBoost♦	0.6165 / 0.5533	0.7560 / 0.6470	0.8278 / 0.7997	0.8578 / 0.8347	2.320% / 3.444%	-18.099% / -18.650%	-8.706% / -8.608%
Gaussian Process♦	0.6390 / 0.4292	0.7620 / 0.4918	0.8473 / 0.6983	0.8566 / 0.8012	1.740% / 5.254%	-18.244% / -18.328%	-6.286% / -7.688%
Random Forest Regressor♦	0.6758 / 0.5899	0.7993 / 0.6564	0.8440 / 0.8059	0.8671 / 0.8300	1.535% / 3.052%	-18.324% / **-18.887%**	-8.879% / -8.616%
Densenet 169★	0.4725 / 0.4216	0.6423 / 0.6167	0.7756 / 0.6433	0.8166 / 0.7901	3.380% / 3.820%	-15.669% / -15.892%	-8.169% / -7.851%
VGG16★	0.5172 / 0.4992	0.5236 / 0.5112	0.7652 / 0.7601	0.8223 / 0.8052	3.083% / 4.125%	-15.536% / -15.812%	-8.088% / -7.593%
ResNet-50★	0.4564 / 0.4045	0.5680 / 0.5367	0.7457 / 0.6962	0.8483 / 0.8278	2.424% / 2.969%	-15.806% / -15.941%	-8.300% / -7.810%
EfficientNet B7★	0.4237 / 0.3920	0.5649 / 0.5612	0.7159 / 0.6905	0.8004 / 0.7781	3.396% / 4.742%	-15.506% / -15.771%	-8.012% / -7.607%

★ Deep neural network-based models. ♦ Handcrafted feature-based models.

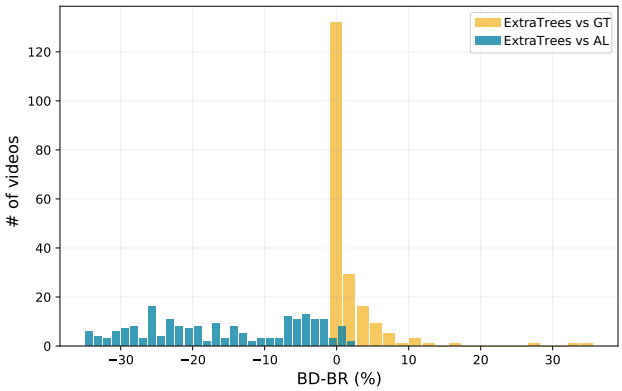


Fig. 5: Histogram of BD-BR for ExtraTrees model compared to AL and GT approaches in terms of YPSNR.

provided in Fig. 5. All the learning-based models considered in this study predict the bitrate ladder without performing any encoding process and hence can be used in live streaming applications.

V. Conclusion

In this paper, we conducted a comprehensive analysis and empirical study of learning-based methods for bitrate ladder prediction in adaptive video streaming. We built a new dataset and tested several well-known ML and DL models for cross-over bitrates prediction. Experimental results demonstrated that ExtraTrees Regressor outperforms other learning-based methods and, compared to the static ladder, it is able to achieve 18% bitrate gain. Moreover, this method significantly reduces the complexity of the exhaustive search method at the cost of 1.4% BD-BR loss. On the other hand, the performance of the pretrained CNN models shows the great potential of using transfer learning techniques for the bitrate ladder prediction task. As future works, we plan to expand the proposed dataset by including more codec types and presets, which can increase the performance of DL models.

References

[1] U. Cisco, "Cisco annual internet report (2018–2023) white paper," *Cisco: San Jose, CA, USA*, 2020.

[2] (2021). [Online]. Available: https://www.fortunebusinessinsights.com/press-release/video-streaming-market-10038

[3] A. Bentaleb, B. Taani, A. C. Begen, C. Timmerer, and R. Zimmermann, "A survey on bitrate adaptation schemes for streaming media over http," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 562–585, 2019.

[4] Http live streaming. [Online]. Available: https://developer.apple.com/streaming/

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

[6] A. Forsythe, "Visual complexity: Is that all there is?" in *Engineering Psychology and Cognitive Ergonomics*, D. Harris, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 158–166.

[7] (2021) Best practices for creating and deploying http live streaming media for the iphone and ipad. [Online]. Available: https://developer.apple.com/documentation/http_live_streaming/http_live_streaming_hls_authoring_specification_for_apple_devices

[8] (2015) Per-title encode optimization. [Online]. Available: https://netflixtechblog.com/per-title-encode-optimization-7e99442b62a2

[9] Z. Li, A. Aaron, I. Katsavounidis, A. Moorthy, and M. Manohara, "Toward a practical perceptual video quality metric," https://netflixtechblog.com/toward-a-practical-perceptual-video-quality-metric-653f208b9652, 2016.

[10] A. V. Katsenou, J. Sole, and D. R. Bull, "Content-gnostic bitrate ladder prediction for adaptive video streaming," in *2019 Picture Coding Symposium (PCS)*. IEEE, 2019, pp. 1–5.

[11] 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 Motion Imaging Journal*, vol. 131, no. 3, pp. 42–50, 2022.

[12] H. Xing, Z. Zhou, J. Wang, H. Shen, D. He, and F. Li, "Predicting rate control target through a learning based content adaptive model," in *2019 Picture Coding Symposium (PCS)*, 2019, pp. 1–5.

[13] (2018) Instant per-title encoding. [Online]. Available: https://www.mux.com/blog/instant-per-title-encoding

[14] I. Katsavounidis. (2018) Dynamic optimizer — a perceptual video encoding optimization framework. [Online]. Available: https://netflixtechblog.com/dynamic-optimizer-a-perceptual-video-encoding-optimization-framework-e19f1e3a277f

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

[16] 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," in *2018 Picture Coding Symposium (PCS)*, 2018, pp. 179–183.

[17] (2020) Per-title encoding. [Online]. Available: https://bitmovin.com/per-title-encoding

[18] (2018) Save bandwidth and improve viewer quality of experience with source adaptive bitrate ladders. [Online]. Available: https://capellasystems.net/wp-content/uploads/2021/01/CambriaFTC_SABL.pdf

[19] Y. Zhang, S. Kwong, and S. Wang, "Machine learning based video coding optimizations: A survey," *Information Sciences*, vol. 506, pp. 395–423, 2020.

[20] Z. Li, Z. Duanmu, W. Liu, and Z. Wang, "Avc, hevc, vp9, avs2 or av1?—a comparative study of state-of-the-art video encoders on 4k videos," in *International Conference on Image Analysis and Recognition*. Springer, 2019, pp. 162–173.

[21] Y. Wang, S. Inguva, and B. Adsumilli, "Youtube ugc dataset for video compression research," in *2019 IEEE 21st International Workshop on Multimedia Signal Processing (MMSP)*. IEEE, 2019, pp. 1–5.

[22] Intelligent scene cut detection and video splitting tool. [Online]. Available: https://pyscenedetect.readthedocs.io/en/latest/

[23] Ffmpeg. [Online]. Available: https://www.ffmpeg.org/

[24] M. Afonso, F. Zhang, and D. R. Bull, "Spatial resolution adaptation framework for video compression," in *Applications of Digital Image Processing XLI*, vol. 10752. SPIE, 2018, pp. 209–218.

[25] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.