Effective Context-Aware Bitrate Ladder Construction for Adaptive Video Streaming

Anant Shukla
Concordia University
Montreal, Canada
me@anantshukla.com

Avni Gupta
Concordia University
Montreal, Canada
avni.gupta12@gmail.com

ABSTRACT

The increasing prevalence of digital content consumption has established video streaming as a predominant force, constituting a significant portion of global internet traffic. This upsurge in demand necessitates adaptive bitrate streaming to ensure seamless user experiences. Traditional fixed bitrate ladders often prove inadequate in adapting to diverse network conditions, leading to the exploration of intelligent and context-aware solutions. Our research introduces a Deep Reinforcement Learning (DRL) approach for context-aware bitrate ladder construction, considering both objective and subjective Quality of Experience (QoE) metrics.

The DRL algorithm undergoes training on video features, network bandwidth, and storage costs, providing an adaptive solution to varying content, network, and storage demands. We compile a dataset of diverse high-resolution videos, evaluate different video feature extraction techniques, and propose a fused QoE metric. Our approach, which incorporates both content and network awareness, aims to address the limitations of fixed bitrate ladders, offering a more responsive and optimized streaming experience.

The study outlines the system architecture and implementation details, while discussing its limitations, paving the way for future work in variable bitrate encoding and alternative DRL architectures. In summary, our research contributes to the advancement of understanding and implementation of context-aware bitrate ladder construction in video streaming.

Author Keywords

Context Aware; Bitrate Ladder; Adaptive Video Streaming; QoE Metric Fusion

CCS Concepts

•**Information systems** → **Multimedia streaming;**

1. INTRODUCTION

In the ever-expanding landscape of digital content consumption, video streaming has emerged as a dominant force, commanding a significant share of global internet traffic. With the proliferation of high-speed internet and the ubiquity of connected devices, users increasingly turn to adaptive bitrate streaming to enjoy seamless video experiences. This surge in demand has necessitated a paradigm shift in video delivery mechanisms, requiring intelligent and context-aware solutions to enhance user satisfaction and optimize network resources.

Category	*Downstream Traffic Share*
Video Streaming	48.9%
Social Networking	19.3%
Web	13.1%
Messaging	6.7%
Gaming	4.3%
Marketplace	4.1%
Others	3.5%

Table 1. Mobile downstream traffic Q1 2021 [24]

The exponential growth of video traffic on the internet is undeniable, with video streaming constituting a substantial portion of the data transmitted across networks. According to Sandvine's 2023 Global Internet Phenomena Report, video usage grew by 24% in 2022, now representing 65% of all internet traffic [2]. Whether for entertainment, education, or communication, users are drawn to the dynamic and engaging nature of video content. This surge poses challenges to content delivery networks (CDNs) and necessitates sophisticated strategies to efficiently handle the diverse demands of a global audience.

According to Android Central's report of September 2021, streaming video consumes approximately 0.7GB per hour for a 480p video, 1.5 GB per hour for 1080p, and 7.2 GB per hour for a 4K stream [24]. At the heart of contemporary video streaming is Adaptive Bitrate Streaming (ABR), a technology designed to dynamically adjust the quality of video streams based on the viewer's network conditions. ABR algorithms assess factors such as available bandwidth, device capabilities, and network congestion to seamlessly switch between different bitrate renditions, ensuring a smooth viewing experience. This dynamic adjustment, however, heavily relies on the construction of a bitrate ladder.

The bitrate ladder is a crucial component of ABR, representing a set of pre-encoded video versions at different quality levels. Traditionally, these ladders are statically defined, offering a fixed range of bitrates to accommodate various network conditions. However, with the diverse array of devices, network

CHI'20, April 25–30, 2020, Honolulu, HI, USA

ISBN 978-1-4503-6708-0/20/04…$15.00

DOI: https://doi.org/10.1145/3313831.XXXXXXX

types, and user preferences, fixed bitrate ladders often fall short of delivering optimal performance.

Fixed bitrate ladders encounter challenges in addressing the intricacies of real-world scenarios. They struggle to adapt to the unpredictable nature of network conditions, leading to sub-optimal quality and buffering issues. Moreover, the static nature of these ladders overlooks the content characteristics, often resulting in inefficient utilization of available resources.

To overcome the limitations of fixed bitrate ladders, this research proposes an approach leveraging DRL. By employing DRL, we aim to construct bitrate ladders that are context-aware, taking into account various factors such as video features, quality metrics, video sizes, network conditions, and the importance of storage costs. Our key contributions are as follows:

- We curated a dataset comprising 80 high-resolution videos spanning diverse genres, including Gaming, Music, Action, News, Mixed Content, and Animation.
- We conducted a comprehensive analysis of various video feature extraction techniques, assessing their impact on the bitrate ladder constructed by the DRL algorithm.
- We trained a machine learning model to create a fused QoE metric by combining Mean Opinion Score (MOS) with objective QoE metrics (VMAF, SSIM, and PSNR) to predict MOS for new datasets that have not been through subjective tests.
- We systematically evaluated and compared the fused QoE metric with VMAF to identify their impact on bitrate ladder construction.
- We performed the training of a DRL algorithm to calculate an optimal bitrate ladder, taking as input both video and network characteristics, enabling the algorithm to learn intricate patterns and correlations.

2. RELATED WORK

The majority of video streaming platforms adopt Adaptive Bitrate technologies such as Dynamic Adaptive Streaming over HTTP (DASH) [15] and HTTP Live Streaming (HLS) [10] to ensure Quality of Experience for viewers. In these technologies, content is divided into short segments (usually 4-10 seconds) and encoded into different bitrates. A manifest file is created with video metadata, and each video is encoded into different resolution-bitrate settings. Clients adaptively select the optimal bitrate based on network conditions, ensuring smooth playback with transitions between bitrates for a seamless streaming experience.

The traditional approach to bitrate ladder construction involves the use of fixed bitrate-resolution pairs [14], which are not optimized for specific content. This "one-size-fits-all" approach has been commonly used to reduce streaming costs and improve the quality of experience for end-users.

Numerous studies have focused on more efficient and content-optimized methods for bitrate ladder construction. For example, a method has been proposed that extracts spatio-temporal features from uncompressed content and trains machine-learning models to predict the Pareto front, resulting in a significant reduction in computation required [12]. Angeliki et al. proposed a machine learning-based scheme for predicting the bitrate ladder based on the content of the video using two constituent models [13]. Another study conducted a benchmark of several handcrafted and deep learning-based approaches for predicting content-optimized bitrate ladders, highlighting the growing interest in machine learning techniques for this task [23]. In a blog post, the industrial leader Netflix discussed the introduction of "shot-based encodes" for 4K content, which involves optimizing the bitrate ladder based on the specific characteristics of individual shots within a video [4]. While all these solutions are content-aware, they lack network awareness and are inefficient in adapting to rapid network fluctuations, leading to sub-optimal streaming quality during varying network speeds.

In contrast to content-aware bitrate ladders, **context-aware bitrate ladders** take into account broader contextual factors, including network conditions, device capabilities, and user preferences. By incorporating information about the network's bandwidth, latency, and device capabilities, context-aware bitrate ladders strive to deliver an optimized streaming experience tailored not only to the content being streamed but also responsive to the dynamic conditions of the network and user requirements. Hadi Amirpour et al. proposed optimizing the bitrate ladder dynamically based on the context in which the video is being viewed, providing the best possible quality of experience for the viewer [9]. Another scheme for constructing context-aware bitrate ladders, DeepLadder, presents a novel approach by integrating deep learning techniques into bitrate ladder construction. Leveraging advanced neural networks, DeepLadder seeks to dynamically optimize the bitrate ladder based not only on content characteristics but also on contextual factors, ushering in a more adaptive and intelligent streaming solution [11].

Our solution is both **content-aware and network-aware; therefore, it is inherently context-aware**.

3. PROBLEM DESCRIPTION AND MOTIVATION

The focal point of this research is the generation of a context-aware bitrate ladder, a pivotal element in Adaptive Bitrate Streaming for video delivery. Unlike traditional fixed bitrate ladders or content-aware approaches, the context-aware bitrate ladder aims to dynamically optimize video streaming by considering a multitude of factors beyond content characteristics.

We analyzed two videos featuring disparate genres—an action-packed fight scene and a music clip; evaluating their VMAF at both 240p (Ref. Figure 1) and 1080p (Ref. Figure 2) resolutions across various bitrates. Our findings revealed significant differences in VMAF values for both videos at the same bitrate and resolution, emphasizing that diverse content necessitates different encoding bitrates.

As observed in Figure 3, we segmented a single video into 10-second segments and encoded all the segments with 720p resolution and 100Kbps, 750Kbps, 285Kbps, and 7000Kbps. There is a drastic variation in VMAF scores among the video

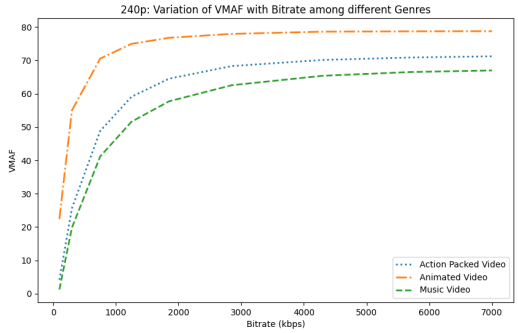

Figure 1. VMAF scores across bitrates for different video genres at 240p

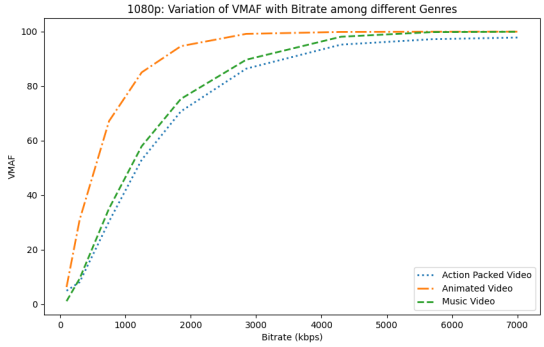

Figure 2. VMAF scores across bitrates for different video genres at 1080p

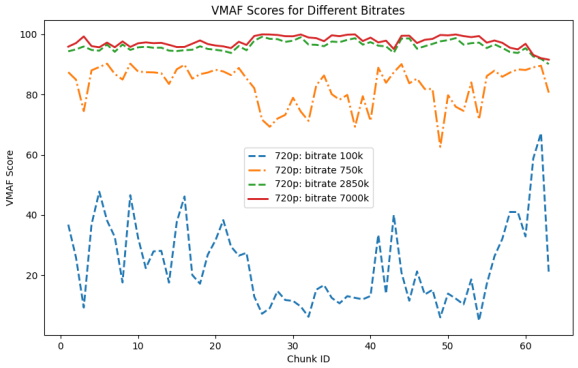

Figure 3. Temporal Evolution of VMAF Scores Across Data Chunks

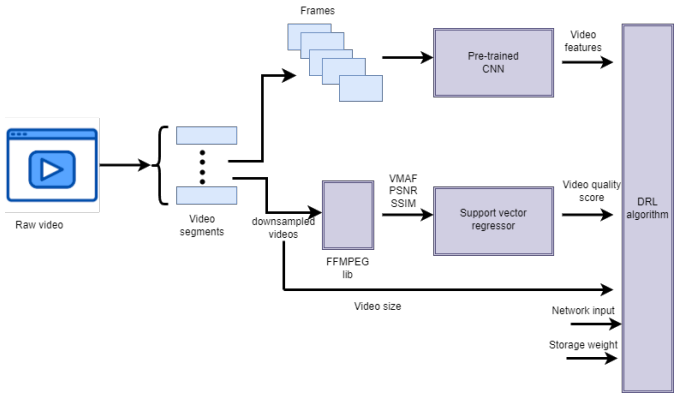

Figure 4. Broad System Architecture

segments, indicating that different portions of the video should be encoded with different bitrates to achieve an overall optimal QoE.

Intuitively, fluctuations in network conditions can impact streaming performance; high bitrates may result in stalls during poor network conditions, while low bitrates on a stable network may compromise QoE. Recognizing this dependency on network conditions, we incorporated diverse network traces into the bitrate ladder construction. This ensures that the bitrate ladder is adaptive to varying network scenarios, contributing to a more robust and seamless streaming experience.

While existing context-aware approaches predominantly focus on objective video metrics [11] in conjunction with network conditions to construct a bitrate ladder, our innovative methodology takes a more comprehensive stance. We incorporate both objective and subjective QoE metrics, alongside detailed video frame features, to formulate a bitrate ladder. This holistic approach aims to provide a more nuanced and adaptable solution for optimizing streaming quality.

In essence, our findings underscore the importance of video content features, prevailing network traffic capacities, and the overall storage cost on demand when constructing an optimal bitrate ladder. Traditional heuristics often struggle to adeptly integrate such metrics from diverse perspectives. In contrast, we approach the bitrate ladder challenge as a sequential decision-making process, aligning with the principles of reinforcement learning. Moreover, we leverage the power of DRL because it excels in generalizing from raw, unprocessed data without requiring manual engineering interventions.

4. SYSTEM ARCHITECTURE

Motivated by the analysis conducted, we propose a Deep Reinforcement Learning algorithm for the construction of a bitrate ladder. In this section, we present the overall system architecture (Ref. Figure 4).

The bitrate ladder generation begins by segmenting the input video, followed by the extraction of frames from each segment. These frames undergo feature extraction using pre-trained Convolutional Neural Network (CNN) models, and

Segment size	*Encoding time*
3 sec	557 sec
7 sec	484 sec

Table 2. Encoding time for different segment duration

the resulting feature maps are stored in a Hierarchical Data Format (HDF) for subsequent use in DRL.

Simultaneously, the video segments are downsampled into various bitrate-resolution pairs, comprising six resolutions (144p, 240p, 360p, 480p, 720p, 1080p) and nine bitrates (100, 300, 750, 1250, 1850, 2850, 4300, 5700, 7000) kbps. This process yields a total of 54 video segments for each original video segment. To assess QoE, metrics such as PSNR, SSIM, and VMAF are extracted using the FFmpeg library. These metrics are then combined with MOS to obtain a fused QoE metric.

The Proximal Policy Optimization (PPO) [22] DRL algorithm forms the core of our proposed system for bitrate ladder construction. The DRL agent is designed to make sequential decisions based on a set of inputs that collectively contribute to the generation of an adaptive bitrate ladder. These inputs include video features and video QoE metrics, providing insights into the perceptual quality of the encoded videos. Additionally, network statistics are considered, reflecting the dynamic conditions of the network, while storage weights account for the overall storage cost implications. The DRL agent also takes into consideration past actions, ensuring a sequential and coherent decision-making process. By integrating these diverse inputs, the DRL agent engages in a continuous learning process, dynamically adapting the bitrate ladder to varying content, network, and storage demands. This adaptability and responsiveness make the DRL-based system well-suited for optimizing the streaming experience across a multitude of scenarios and conditions. The following subsections delve into the specifics of how each input is processed and how the DRL agent refines its decision-making over time.

4.1 Video input

To facilitate effective deep-learning training, a diverse and comprehensive video dataset is essential for generalization. Initially, we utilized two publicly available datasets [19] and [25], but observed a lack of diversity in video content. Subsequently, we curated a set of 80 videos from various sources, encompassing different genres such as Sports, Gaming, Music, Action, News, Mixed Content, and Animation, all in high resolution and bitrate.

To prepare the videos for experimentation, we employed the FFmpeg library to convert them into DASH format. To ensure a broad representation, each video was segmented into 10-second chunks. We chose this segment size based on the evaluation of FFmpeg's encoding time (Ref. Table 2), finding that smaller segment sizes significantly increased encoding time. Refer Table 2, the encoding time for segments of 3 sec was 557 sec while for 7 sec segments was 484 sec which is considerably less. Moreover, we realised that objective QoE scores for smaller segments were lesser when compared to bigger segment sizes. To validate our choice of a larger

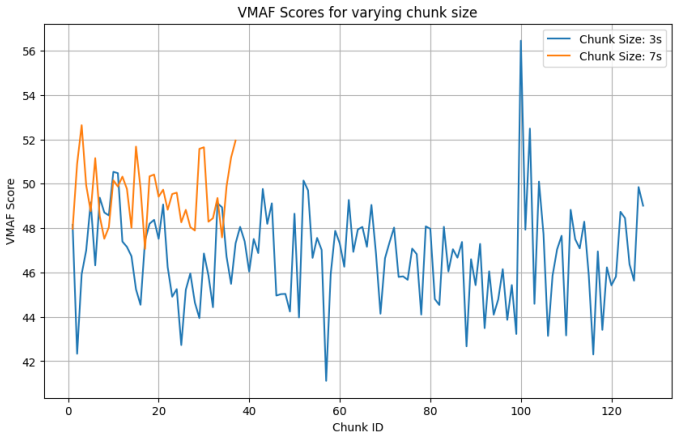

Figure 5. Variation in VMAF across different video segment durations

segment size, we compared the VMAF values between 3-second and 7-second segments. The evaluation revealed that 7-second segments provided superior VMAF values (Refer Figure 5).

4.1.1 Video features

In the feature extraction phase, we employed state-of-the-art CNNs to capture meaningful representations of video content. The process involved extracting frames from video segments using the FFmpeg library, which was then fed into pre-trained image classifiers to generate feature maps. These feature maps were subsequently saved in HDF for later integration into the DRL model.

To ensure a comprehensive assessment, we experimented with four distinct CNN models: ResNet50 V1, ResNet50 V2, XceptionNet, and VGG. These models were selected for their proven efficacy in handling image-related tasks. We changed the last layer of VGG16 and ResNet50 V2 in order to get the uniform set of features from all the CNN models. Later in the study, we compared the performance of these models by evaluating which CNN-generated features contributed to the most effective construction of bitrate ladders for our videos.

4.1.2 Video QoE metrics

Video QoE metrics quantify the perceived quality of video content and assess the fidelity of visual information. These metrics provide a numerical representation of video quality, aiding in the evaluation and comparison of different video encoding and streaming methods.

To create a comprehensive dataset for analysis, each 10-second video segment was downsampled into combination of resolution (144p, 240p, 360p, 480p, 720p, 1080p) and bitrate (100, 300, 750, 1250, 1850, 2850, 4300, 5700, 7000) kbps pair. This resulted in 54 resolution-bitrate pairs for each video segment and in total 181,440 video segments for DRL training. Subsequently, these segments were utilized for QoE metrics calculations.

Subjective Video QoE Metrics: Subjective video QoE metrics involve human observers who assess video quality based

on their perception. Ratings are typically gathered through subjective studies where viewers express their opinions on the visual experience. These metrics are crucial for understanding the end-user experience, capturing nuances that automated algorithms might miss, and guiding the optimization of video delivery systems. The major challenge in the collection of subjective metrics is that they are resource-intensive and time-consuming.

Objective Video QoE Metrics: Objective video QoE metrics are computational algorithms designed to automatically evaluate video QoE without human intervention. These metrics can be obtained without human input and therefore are a valuable tool for research. However, Objective metrics may not align perfectly with human subjective assessments. Viewer preferences, emotional engagement, and other subjective aspects are challenging to quantify objectively. We generated three metrics for the input videos -

- **Structural Similarity Index** [17]: SSIM measures the perceived structural similarity between an original video frame and a compressed frame. It evaluates the luminance, contrast, and structure of the frame, aiming to mimic human perception. The SSIM index ranges from -1 to 1, with 1 indicating perfect similarity.
- **Peak Signal-to-noise Ratio** [8]: PSNR quantifies the ratio between the maximum possible signal strength and the noise or distortion introduced during compression of video. It is calculated using the mean squared error (MSE) between the original and compressed signals. Higher PSNR values indicate better quality, with a maximum value of infinity.
- **Video Multi-Method Assessment Fusion** [20]: VMAF was developed by Netflix and takes into account various visual features and characteristics of the video to provide a score that correlates well with human perception of video quality. It uses a machine learning model that combines multiple quality metrics.

In our study, we aimed to integrate both subjective and objective QoE for robust results. To achieve this, we fused distortion-oriented and perception-oriented QoE metrics to align with MOS for a video [6]. Facing the challenge of lacking subjective scores for our test set, we addressed it by leveraging a comprehensive video quality metric dataset from Netflix [5]. This dataset comprises of 420 videos evaluated by 65 subjects, yielding 9750 continuous-time and 9750 retrospective subjective opinion scores along with associated VMAF, SSIM, and PSNR scores. Content genres cover action, documentary, sports, animation and video games and content characteristics span diverse categories, including natural and animated videos,

For metric fusion, we employed a **ν-support vector regression (ν-SVR)** model, trained to estimate QoE scores by utilizing multiple metrics, namely PSNR, SSIM, and VMAF. We utilized [5] for training the model. The ν-SVR model involved three hyperparameters: ν, representing the proportion of support vectors to total samples; C, the regularization parameter on the loss function; and γ, the radius parameter

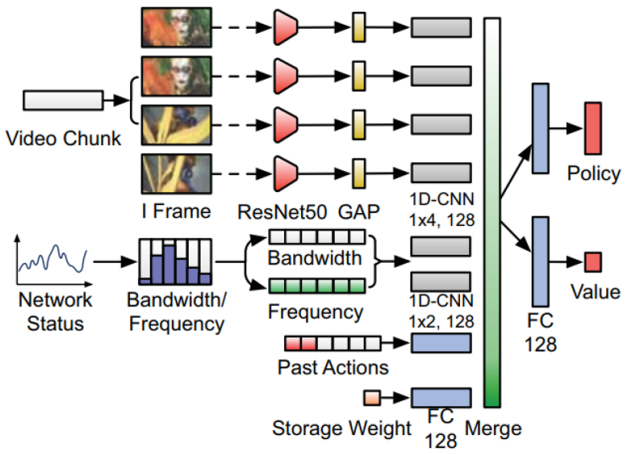

Figure 6. NN Architecture Overview

of the polynomial kernel. A grid search was used to find the best values for hyperparameters with $\nu \in [0.01, 0.1, 0.5, 0.9]$ $C \in [2^{-5}, 2^{15}]$ and $\gamma \in [2^{-15}, 2^{3}]$. Grid search gave the best parameters as: ν: 0.9, C: 32768 and γ: 0.0625 The model was then trained with these hyperparameters and the trained model was then applied to predict the quality scores for our test dataset. We fed these scores as video QoE metric input to the DRL.

Later in the study, we trained the DRL using VMAF as the video quality metric and compared the output bitrate ladder with the bitrate ladder generated using fused QoE scores [6].

4.2 Network statistics

We used network datasets to mimic real-life situations better. This data includes over 3,000 network traces, lasting about 50 hours. We got these traces from different public datasets like HSDPA [21], and FCC [1]. We split the data randomly into two parts: 80% for teaching the computer system and 20% for testing.

4.3 Storage weight

Storage weight depends on the preference of the content provider. Some may give more weightage to the content quality and others to storage. We fixed the storage weight to 0.5 in our experiments.

4.4 Neural Network Architecture

The neural network (NN) architecture is presented in Figure 6.

State For each video chunk t, we have the state space - S_t = { F_t, N_t, P_t, w } where, F_t represents the video features, N_t represents the network bandwidth, P_t represents the past actions and w represents the storage cost weight.

Past Actions: Agent takes past action sequence P_t = { a_0, ..., a_{t-1} } as input where a_i is the action for video resolution i.

Reward r_t: Given network condition C, we want to maximize video QoE and bandwidth utilization and minimize the storage cost.

$$r_t = \underbrace{\sum_t (a_t|C)U(a_t,C)}_{\text{Bandwidth Utilization}} + \underbrace{\sum_t (a_t|C)Q(a_t)}_{\text{Video Quality}} - w\underbrace{\sum_t \S_z(a_t)/t}_{\text{Storage Cost}}$$

$$U(a_t,C) = \begin{cases} B_r(a_t)/C_{a=a_t} & S_z(a_t) \leq C_{a=a_t} \\ 1 - B_r(a_t)/C_{a=a_t} & S_z(a_t) > C_{a=a_t} \end{cases}$$

U: Actual network utilization of the selected chunk size in the current network state

$B_{\mathrm{r}}(a_{\mathrm{t}})$: Bitrate for the picked chunks

$C_{a=a_t}$: Network bandwidth under all actions a_t

$p(a_t|C)$: Probability that the chunk a_t being selected over the given network condition C

Q: Expected video quality $Q(a_t)$ selected by action sequence $\{ a_0, \ldots, a_{\mathrm{t\text{-}1}} \}$ for chunk $\{0,\ldots,t\}$

S_z: Average chunk size for the action sequence $\{ a_0, \ldots, a_{\mathrm{t\text{-}1}} \}$

5. IMPLEMENTATION

We utilized scikit-learn [18] to implement the ν-SVR model [16], TensorFlow version 2.15 [3], and Keras [7] were used to implement the DRL model. Additionally, pre-trained Neural Network models from Keras, initially trained on ImageNet, were used to extract features from the videos. FFmpeg was used for video processing tasks like converting the videos to DASH, downsampling, and extracting various QoE metrics like VMAF, SSIM and PSNR. Model training was performed on a high-speed cluster, and the computational resources included 4 CPUs, each equipped with 8 cores, 32 GB RAM, and an Nvidia Tesla P6 GPU.

Training duration was approximately 16 hours for 80,000 epochs for each DRL model, with this cost incurred solely in the offline stage. Inference, carried out using the online model takes 2-5 seconds.

6. RESULTS

We conducted training for the DRL model using all possible combinations of input features extracted from four distinct CNNs: ResNet50 V1, ResNet50 V2, VGG16, and XceptionNet. Additionally, two QoE metrics, VMAF and fused QoE, were deployed. This process resulted in the creation of eight distinct trained models. We then evaluated the bitrate ladders generated by the DRL models for all network traces used during testing.

To contrast the bitrate ladders produced, we focused on ResNet50 V1 and VMAF as the QoE metric for two distinct networks. One network exhibited an average bandwidth of 2.7 Mbps, while the other had an average bandwidth of 4.6 Mbps (refer to Table 3 and Table 4). As anticipated, the bitrate ladder generated in the low-bandwidth network featured lower bitrates for resolutions compared to the corresponding resolutions in the bitrate ladder for the high-bandwidth network, where higher bitrates were observed.

Representation	*Resolution*	*Encoding Bitrate (Kbps)*
Rep #1	144	215.3
Rep #2	240	537.4
Rep #3	360	750.9
Rep #4	480	1177.7
Rep #5	720	964.3
Rep #6	1080	2031.4
Avg. Bandwidth		2.7 Mbps
Min Bandwidth		0.23 Mbps
Max Bandwidth		4.7 Mbps

Table 3. Computed Bitrate Ladder for a Low Bandwidth Network using XceptionNet and VMAF score

Representation	*Resolution*	*Encoding Bitrate (Kbps)*
Rep #1	144	2864.3
Rep #2	240	2734.9
Rep #3	360	3122.9
Rep #4	480	2993.6
Rep #5	720	1335.1
Rep #6	1080	3252.3
Avg. Bandwidth		4.6 Mbps
Min Bandwidth		2.7 Mbps
Max Bandwidth		5.4 Mbps

Table 4. Computed Bitrate Ladder for a High Bandwidth Network using XceptionNet and VMAF score

Another comparison was conducted, this time focusing on the variation in bitrate ladders generated by the two QoE metrics: VMAF and the fused QoE. For this comparison, we maintained the network configuration as network_norway_bus16 (high bandwidth) and ResNet50 V1 was used to extract the video features. The observation revealed that the bitrate ladder constructed using the fused QoE metric exhibited a more conservative approach in assigning bitrates to resolutions in contrast to VMAF (refer to Table 5 and Table 6).

Furthermore, we conducted a comparison among the bitrate ladders generated using the four CNNs: ResNet50 V1 (Refer table 5), ResNet50 V2 (Refer table 7), VGG16 (Refer table 8), and XceptionNet (Refer table 4). This comparison was performed while maintaining the same QoE metric and network configuration. Unfortunately, no discernible conclusions could be drawn from the analysis of the four bitrate ladders. Further research will be conducted to delve deeper into this analysis, and the results of the extended investigation will also be reported.

7. LIMITATIONS

Due to the unavailability of subjective QoE metrics in our DRL model's training dataset, the ν-SVR was trained on a different dataset. Achieving more robust results might have been possible if the same dataset were used for training both the ν-SVR and DRL. Moreover, our video training dataset for the DRL model comprised 84 videos, which may be considered limited. A larger dataset could potentially enhance the training accuracy and overall results.

Representation	*Resolution*	*Encoding Bitrate (Kbps)*	*VMAF*
Rep #1	144	4286.9	24.27
Rep #2	240	4028.3	61.38
Rep #3	360	3123.0	77.52
Rep #4	480	2735.0	84.22
Rep #5	720	3640.3	88.42
Rep #6	1080	2864.3	89.51

Table 5. Encoding Bitrate and VMAF for Different Resolutions using ResNet50V1 and VMAF score

Representation	*Resolution*	*Encoding Bitrate (Kbps)*	*VMAF*
Rep #1	144	3252.312	23.944
Rep #2	240	3122.982	59.8
Rep #3	360	2864.323	77.528
Rep #4	480	1335.164	72.754
Rep #5	720	2734.993	88.4175
Rep #6	1080	2993.652	89.511

Table 6. Encoding Bitrate and VMAF for Different Resolutions using ResNet50 v1 and Fused QoE score

Representation	*Resolution*	*Encoding Bitrate (Kbps)*	*VMAF*
Rep #1	144	1335.164	21.9235
Rep #2	240	3252.312	59.8
Rep #3	360	2734.993	77.52
Rep #4	480	3122.982	84.22
Rep #5	720	2993.652	88.41
Rep #6	1080	2864.323	89.51

Table 7. Encoding Bitrate and VMAF for Different Resolutions using ResNet50 V2 and Fused QoE score

Representation	*Resolution*	*Encoding Bitrate (Kbps)*	*VMAF*
Rep #1	144	3122.982	23.94
Rep #2	240	2993.652	59.8
Rep #3	360	1335.164	68.05
Rep #4	480	2734.993	84.22
Rep #5	720	2864.323	88.41
Rep #6	1080	3381.641	89.51

Table 8. Encoding Bitrate and VMAF for Different Resolutions using VGG16 and Fused QoE score

Furthermore, owing to time constraints, the DRL model underwent training for only 80,000 epochs. Extending the training duration, preferably to around 1,000,000 epochs, is expected to yield improved performance. Finally, the network datasets used did not include traces for high bandwidth conditions, which resulted in the creation of bitrate ladders that lack high bitrates, such as 7000 Kbps. A more diverse dataset would be used to address this limitation in future work, which would enhance the model's ability to generalize and perform effectively under a broader range of network conditions.

8. FUTURE WORK

The current research primarily focuses on constructing an optimal bitrate ladder using Constant Bitrate (CBR). In future work, we aim to broaden this focus by incorporating variable bitrate (VBR) encoding. Additionally, our research will extend to evaluate the performance of alternative Deep Reinforcement Learning architectures for an optimal bitrate ladder construction.

9. CONCLUSION

In conclusion, our research project has successfully implemented a Context-Aware Bitrate Ladder Construction approach using Deep Reinforcement Learning. Going beyond the consideration of objective Quality of Experience, our DRL algorithm incorporates subjective QoE metrics in the construction of an optimal bitrate ladder considering the video features and the available bandwidth, while minimizing the storage cost.

10. REFERENCES

[1] 2016. Raw Data Measuring Broadband America. In *Fixed Broadband Report*. https://www.fcc.gov/reports-research/reports/measuring-broadband-america/raw-data-measuring-broadband-america-2016

[2] 2023. Sandvine's 2023 Global Internet Phenomena Report. (2023). https://www.prnewswire.com/news-releases/

[3] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2015. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. (2015). https://www.tensorflow.org/ Software available from tensorflow.org.

[4] Megha Manohara Jan De Cock Anne Aaron, Zhi Li and David Ronca. 2015. Per-Title Encode Optimization. (2015). https://netflixtechblog.com/per-title-encode-optimization-7e99442b62a2

[5] Christos Bampis, Zhi Li, Ioannis Katsavounidis, Te-Yuan Huang, Chaitanya Ekanadham, and Alan Bovik. 2021. Towards Perceptually Optimized Adaptive Video Streaming -A Realistic Quality of Experience Database. *IEEE Transactions on Image Processing* PP (04 2021), 1–1. DOI:http://dx.doi.org/10.1109/TIP.2021.3073294

[6] Aaron Chadha, Ioannis Katsavounidis, Ayan Kumar Bhunia, Cosmin Stejerean, Mohammad Umar Karim Khan, and Yiannis Andreopoulos. 2022. Domain-Specific Fusion Of Objective Video Quality Metrics *(MM '22)*. Association for Computing Machinery, New York, NY, USA, 1387–1395. DOI: http://dx.doi.org/10.1145/3503161.3548375

[7] François Chollet. 2015. keras. (2015). https://github.com/fchollet/keras

Representation	*Resolution*	*Encoding Bitrate (Kbps)*	*VMAF*
Rep #1	144	2864.322636	23.94
Rep #2	240	2734.992994	59.8
Rep #3	360	3122.981922	77.52
Rep #4	480	2993.652279	84.22
Rep #5	720	1335.164086	73.39
Rep #6	1080	3252.311565	89.51

Table 9. Encoding Bitrate and VMAF for Different Resolutions using XceptionNet and Fused QoE score

[8] Fernando Fardo, Victor Conforto, Francisco Oliveira, and Paulo Rodrigues. 2016. A Formal Evaluation of PSNR as Quality Measurement Parameter for Image Segmentation Algorithms. (05 2016).

[9] Hadi Amirpour Sergey Gorinsky Junchen Jiang Hermann Hellwagner Farzad Tashtarian, Abdelhak Bentaleb and Christian Timmerer. 2024. ARTEMIS: Adaptive Bitrate Ladder Optimization for Live Video Streaming. (2024).

[10] Kalle Honkanen, Opinnäytetyö Toukokuu, Tietotekniikka Tietoliikennetekniikka, and Tampereen Ammattikorkeakoulu. 2011. HTTP live streaming. (01 2011).

[11] Tianchi Huang and Lifeng Sun. 2021. Optimized Bitrate Ladders for Adaptive Video Streaming with Deep Reinforcement Learning. In *Proceedings of the SIGCOMM '20 Poster and Demo Sessions (SIGCOMM '20)*. Association for Computing Machinery, New York, NY, USA, 46–48. DOI: http://dx.doi.org/10.1145/3405837.3411387

[12] Angeliki V. Katsenou, Joel Sole, and David R. Bull. 2021a. Efficient Bitrate Ladder Construction for Content-Optimized Adaptive Video Streaming. *IEEE Open Journal of Signal Processing* 2 (2021), 496–511. DOI:http://dx.doi.org/10.1109/OJSP.2021.3086691

[13] Angeliki V. Katsenou, Joel Solé, and David R. Bull. 2021b. Efficient Bitrate Ladder Construction for Content-Optimized Adaptive Video Streaming. *IEEE Open Journal of Signal Processing* 2 (2021), 496–511. https://api.semanticscholar.org/CorpusID:231855356

[14] Vignesh V Menon, Hadi Amirpour, Mohammed Ghanbari, and Christian Timmerer. 2022. Efficient bitrate ladder construction for live video streaming. 99–100. DOI: http://dx.doi.org/10.1145/3510450.3517300

[15] Michail Michalos, Stelios Kessanidis, and S.L. Nalmpantis. 2012. Dynamic adaptive streaming over HTTP. *Journal of Engineering Science and Technology Review* 5 (06 2012), 30–34. DOI: http://dx.doi.org/10.25103/jestr.052.06

[16] Suganyadevi M.V and Babulal C.K. 2014. Support Vector Regression Model for the prediction of Loadability Margin of a Power System. *Applied Soft Computing* 24 (11 2014), 304–315. DOI: http://dx.doi.org/10.1016/j.asoc.2014.07.015

[17] Jim Nilsson and Tomas Akenine-Möller. 2020. Understanding SSIM. (06 2020).

[18] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, and others. 2011. Scikit-learn: Machine learning in Python. *Journal of machine learning research* 12, Oct (2011), 2825–2830.

[19] Jason J. Quinlan and Cormac J. Sreenan. 2018. Multi-Profile Ultra High Definition (UHD) AVC and HEVC 4K DASH Datasets *(MMSys '18)*. Association for Computing Machinery, New York, NY, USA, 375–380. DOI: http://dx.doi.org/10.1145/3204949.3208130

[20] Reza Rassool. 2017. VMAF reproducibility: Validating a perceptual practical video quality metric. In *2017 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*. 1–2. DOI:http://dx.doi.org/10.1109/BMSB.2017.7986143

[21] Haakon Riiser, Paul Vigmostad, Carsten Griwodz, and Pål Halvorsen. 2013. Commute Path Bandwidth Traces from 3G Networks: Analysis and Applications. In *Proceedings of the 4th ACM Multimedia Systems Conference (MMSys '13)*. Association for Computing Machinery, New York, NY, USA, 114–118. DOI: http://dx.doi.org/10.1145/2483977.2483991

[22] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal Policy Optimization Algorithms. *CoRR* abs/1707.06347 (2017). http://arxiv.org/abs/1707.06347

[23] Ahmed Telili, Wassim Hamidouche, Sid Ahmed Fezza, and Luce Morin. 2022. Benchmarking Learning-based Bitrate Ladder Prediction Methods for Adaptive Video Streaming. In *2022 Picture Coding Symposium (PCS)*. 325–329. DOI: http://dx.doi.org/10.1109/PCS56426.2022.10018038

[24] Omri Wallach. 2021. The World's Most Used Apps, by Downstream Traffic. (2021). https://www.visualcapitalist.com/the-worlds-most-used-apps-by-downstream-traffic/#google_vignette

[25] Anatoliy Zabrovskiy, Christian Feldmann, and Christian Timmerer. 2018. Multi-Codec DASH Dataset. In *Proceedings of the 9th ACM Multimedia Systems Conference (MMSys '18)*. Association for Computing Machinery, New York, NY, USA, 438–443. DOI: http://dx.doi.org/10.1145/3204949.3208140