Just noticeable difference-aware per-scene bitrate-laddering for adaptive video streaming

Vignesh V Menon, Jingwen Zhu, Prajit T Rajendran, Hadi Amirpour, Patrick Le Callet, Christian Timmerer

JUST NOTICEABLE DIFFERENCE-AWARE PER-SCENE BITRATE-LADDERING FOR ADAPTIVE VIDEO STREAMING

Vignesh V Menon[1] *Jingwen Zhu*[2] *Prajit T Rajendran*[3] *Hadi Amirpour*[1]
Patrick Le Callet[2] *Christian Timmerer*[1]

[1]Christian Doppler Laboratory ATHENA, Alpen-Adria-Universität, Klagenfurt, Austria
[2]Nantes Universite, Ecole Centrale Nantes, CAPACITES SAS, CNRS, LS2N, UMR 6004, F-44000 Nantes, France
[3]CEA, List, F-91120 Palaiseau, Université Paris-Saclay, France

ABSTRACT

In video streaming applications, a fixed set of bitrate-resolution pairs (known as a *bitrate ladder*) is typically used during the entire streaming session. However, an optimized bitrate ladder per scene may result in *(i)* decreased storage or delivery costs or/and *(ii)* increased *Quality of Experience*. This paper introduces a Just Noticeable Difference (JND)-aware per-scene bitrate ladder prediction scheme (JASLA) for adaptive video-on-demand streaming applications. JASLA predicts jointly optimized resolutions and corresponding constant rate factors (CRFs) using spatial and temporal complexity features for a given set of target bitrates for every scene, which yields an efficient constrained Variable Bitrate encoding. Moreover, bitrate-resolution pairs that yield distortion lower than one JND are eliminated. Experimental results show that, on average, JASLA yields bitrate savings of 34.42% and 42.67% to maintain the same PSNR and VMAF, respectively, compared to the reference *HTTP Live Streaming* (HLS) bitrate ladder Constant Bitrate encoding using x265 HEVC encoder, where the maximum resolution of streaming is Full HD (1080p). Moreover, a 54.34% average cumulative decrease in storage space is observed.

***Index Terms*—** Bitrate ladder, per-scene encoding, video streaming, Just Noticeable Difference.

1. INTRODUCTION

Motivation: *Video on Demand* (VoD) and live video streaming are widely embraced in video services, and their applications have attracted tremendous attention in recent years [1]. Since streaming services continuously adapt video delivery to the end user's network conditions and device capabilities, *HTTP Adaptive Streaming* (HAS) continues to grow and has become the *de-facto* standard for delivering video over the Internet [2]. In HAS, each video is encoded at a set of bitrate-resolution pairs, referred to as *bitrate ladder*. Traditionally, a fixed bitrate ladder, *e.g.*, *HTTP Live Streaming* (HLS) bitrate ladder[1], is used for all video contents.

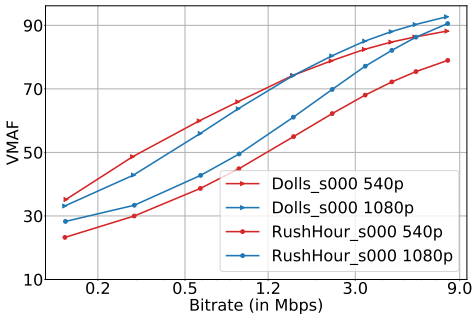


Fig. 1: RD curve of 540p and 1080p CBR encodings of *Dolls_s000* and *RushHour_s000* [4] video sequences using x265 HEVC encoder at *slower* preset.

However, due to the vast diversity in video content characteristics and network conditions, the "one-size-fits-all" can be optimized per *scene* to increase the *Quality of Experience* (QoE) or decrease the bitrate of the representations as introduced for VoD services [3]. Per-scene encoding schemes are based on the fact that one resolution performs better than others in a *scene* for a given bitrate range, and these regions depend on the *video complexity* [5]. As shown in Fig. 1, for *Dolls_s000*, the cross-over bitrate between 540p and 1080p resolutions happens at approximately 2.0 Mbps, which means at bitrates lower than 2.0 Mbps, 540p resolution outperforms 1080p in terms of VMAF[2]. In comparison, at bitrates higher than 2.0 Mbps, 1080p resolution outperforms 540p. On the other hand, for *RushHour_s000*, 1080p yields higher VMAF over the entire bitrate range, which means 1080p should be selected for the bitrate ladder for the entire bitrate range. This *content-dependency* to choose the optimal bitrate-resolution pairs is the basis for introducing a *per-scene* encoding scheme. Each scene in a video and its corresponding downscaled versions are encoded at several bitrates. The bitrate-resolution pair with the highest quality is selected for each target bitrate [3]. Considering M resolutions and N bitrates, $M \times N$ test encodings are needed to determine

[1]https://developer.apple.com/documentation/http_live_streaming/hls_authoring_specification_for_apple_devices, last access: Apr 30, 2023.

[2]https://netflixtechblog.com/vmaf-the-journey-continues-44b51ee9ed12, last access: Apr 30, 2023.

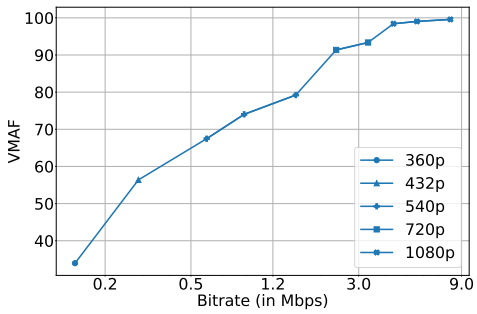


Fig. 2: RD curve of HLS[1] CBR encoding of *Characters_s000* video sequence (segment) of VCD dataset [4] using x265 HEVC encoder at *slower* preset. The points with a bitrate greater than 3.6 Mbps are in the perceptually lossless region.

the optimal per-scene bitrate ladder. To avoid a brute force encoding of all bitrate-resolution pairs, some methods pre-analyze the video contents[3]. Katsenou *et al.* [6] introduced a content-gnostic method that employs machine learning to find the bitrate range for each resolution that outperforms other resolutions. Bhat *et al.* [7] proposed a Random Forest (RF) classifier to decide the best encoding resolution over different quality ranges.

As shown in Fig. 2, the bitrate-resolution pairs of the bitrate ladder may not always be perceptually different in video quality. It is observed that there are multiple representations with similar video quality (*i.e.*, VMAF close to 100) for the *Characters_s000* sequence using the HLS bitrate ladder. Having many perceptually redundant representations for the bitrate ladder may not result in improved quality of experience, but it may lead to increased storage and bandwidth costs [8]. Hence, predicting the threshold where visible distortion occurs compared to the original scene (referred to as JND) is critical. Wang *et al.* [9] proposed a model using Support Vector Regression (SVR) to predict JND based on masking effect features [10] extracted from the source video and quality degradation features computed from various encoded videos. Zhang *et al.* [11] improved the JND prediction accuracy by considering the spatial and temporal information features via deep learning. However, the source (raw) video needs to be encoded several times (*e.g.*, QP from 1 to 51 with a step size of 1) before conducting these models, which is computationally expensive. Zhu *et al.* [12] proposed a JND prediction model that only inputs the source video. Moreover, the Video-Wise JND dataset (HD-VJND) is collected using CRF as a proxy, and a JND prediction model is proposed by extracting three types of features [13]. Even though these JND prediction models only take the source video as input, they are still computationally intensive because of the high complexity of features (*e.g.*, masking effect features [10]).

Contributions: In this paper, a **J**ust Noticeable Difference (JND)-**a**ware per-**s**cene bitrate **la**dder prediction scheme (JASLA) is proposed that improves encoding bitrate ladders for adaptive video-on-demand streaming applications. JASLA predicts optimized resolution and the corresponding constant rate factor (CRF) using spatial and temporal complexity features, for all target bitrates defined by the streaming service provider for efficient constrained Variable Bitrate (cVBR) encoding. Furthermore, a JND threshold prediction scheme is implemented to eliminate the representations which yield distortion lower than the noticeable distortion for every scene.

2. JASLA ARCHITECTURE

The JASLA architecture is shown in Fig. 3. The resolution and the corresponding CRF for each bitrate in the bitrate ladder are predicted for every scene using the scene's spatial and temporal complexity features, the set of pre-defined resolutions (R), and the set of pre-defined bitrates (B) for an efficient cVBR steaming. An optimized bitrate ladder for every scene ensures streaming quality with no bitrate fluctuations. R is input to JASLA to confirm that only the resolutions supported by the streaming service provider are selected to generate the optimized bitrate ladder. Next, the bitrate-resolution pairs whose perceptual quality is less than one JND compared to the source video are eliminated. In this way, the number of representations needed for streaming is reduced. The encoding process is carried out only for the predicted bitrate-resolution-CRF pairs for every scene.

JASLA comprises three steps: *(i)* scene complexity features extraction, *(ii)* optimized resolution and CRF prediction, and *(iii)* JND threshold prediction which are described in the following.

2.1. Scene Complexity Features Extraction

In video streaming applications, an intuitive method for feature extraction would be to utilize Convolutional Neural Networks (CNNs) [14]. However, such models have several inherent disadvantages, such as higher training time, inference time, and storage requirements, which are impractical in streaming applications. The popular state-of-the-art video complexity features are Spatial Information (SI) and Temporal Information (TI)[4]. But the correlation of SI and TI features with the encoding output features such as bitrate, encoding time *etc.* are very low, which is insufficient for encoding parameter prediction in streaming applications [15–18].

In this paper, seven DCT-energy-based features [19], the average luma texture energy E_Y, the average gradient of the luma texture energy h, the average luminescence L_Y, the average chroma texture energy E_U and E_V (for U and V planes) and the average chrominance L_U and L_V (for U and V planes), which are extracted using VCA[5] open-source video complexity analyzer [18, 20] are used as the spatial and temporal complexity measures [5, 21] of every scene.

[3]https://bitmovin.com/per-title-encoding/, last access: Apr 30, 2023.

[4]https://www.itu.int/rec/T-REC-P.910-202207-I, last access: Apr 30, 2023.

[5]https://vca.itec.aau.at, last access: Apr 30, 2023.

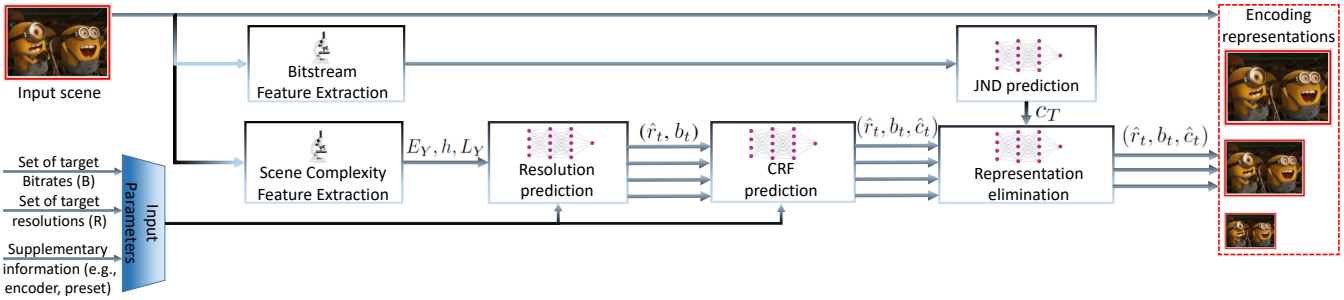


Fig. 3: JASLA architecture.

Algorithm 1: Optimized resolution and CRF prediction

Inputs:

R : set of all resolutions $\tilde{r}_m \; \forall \; m \in [1, M]$

M : number of resolutions in R

B : set of all bitrates $b_t \; \forall \; t \in [1, N]$

N : number of bitrates in B

E_Y, h, L_Y : average scene complexity

Output: $(\hat{r}, b, \hat{c})$ pairs of the bitrate ladder

for $t \in [1, N]$ **do**

  for $m \in [1, M]$ **do**

    Determine $v_{\tilde{r}_m, b_t}$ with $[E_Y, h, L_Y, log(b_t)]$, using the model trained for $\tilde{r}_m$.

  $\hat{r}_t = \arg\max_{\tilde{r}_m \in R}(v_{\tilde{r}, b_t})$

  Determine $\hat{c}_t$ with $[E_Y, h, L_Y, log(b_t)]$, using the model trained for $\hat{r}_t$.

  $(\hat{r}_t, b_t, \hat{c}_t)$ is the $(t)^{th}$ point of the bitrate ladder

2.2. Optimized Resolution and CRF Prediction

For each scene, the optimized resolution for a given target bitrate is predicted using the scene's spatial and temporal features, the set of supported resolutions (R), and the set of target bitrates (B). To determine the bitrate-resolution pairs of the bitrate ladder, VMAF is predicted for each target bitrate (b_t) in the set B for all resolutions $\tilde{r}$ in R, denoted as $v_{\tilde{r}, b_t}$. From the predicted VMAF values, the resolution which yields the maximum VMAF value is chosen as the optimized resolution for the target bitrate. Random Forest (RF) models are trained to predict VMAF for every resolution supported by the streaming service provider. This ensures *scalability* of design, where there is no requirement to retrain the entire network to add a new resolution to the framework.

Using the E_Y, h, L_Y features, optimized CRF $\hat{c}_t$ is estimated for every $(\hat{r}_t, b_t)$ representation of the bitrate ladder for cVBR encoding. Prediction models are trained for each resolution $\tilde{r}$ in R, which determines $\hat{c}_t$ based on E_Y, h, L_Y and $log(b_t)$ for every scene. The minimum and maximum CRF (c_{min} and c_{max}, respectively) are chosen based on the target codec. For example, x265[7] supports a CRF range between 0 and 51. The prediction algorithm for the bitrate ladder is shown in Algorithm 1.

2.3. JND Threshold Prediction

This paper proposes a reduced complexity JND prediction model derived from [13], which predicts the minimum CRF where perceptual distortion is introduced, as shown in Fig. 4. Three types of low-complexity features, *(i)* scene complexity features, *(ii)* bitstream features, and *(iii)* Gray-Level Co-occurrence Matrix (GLCM) features are extracted from the input video scene to predict the JND threshold CRF (c_T).

1) *Scene complexity features* X_S = { E_Y, h, L_Y, E_U, E_V, L_U, L_V } are used instead of the masking effect features in [13] due to efficiency reasons.
2) *Bitstream features* [22]: The scene is first compressed into a near-lossless version to extract bitstream features (denoted as X_B), including framerate, bitrate, framesize, motion (horizontal and vertical), *etc.* The bitstream features are extracted using *videoparse*[6] without decoding pixel information [23].
3) *Gray-Level Co-occurrence Matrix (GLCM) features* [24]: Among the nature scene statistic features used in [13], only GLCM features [24] are used in this paper owing to their importance in JND prediction. Each frame is cropped into patches of size $Q \times Q$. For each patch, the co-occurrence matrix is computed as X_G = {contrast, dissimilarity, homogeneity, angular second moment, energy, correlation}.

This paper considers five types of pooling, *i.e.*, { mean, std, max, skew, kurt}. Pooled scene complexity features is computed as $\hat{X}_S = F_t(X_S)$, where F_t is the temporal pooling among frames. Similarly, pooled bitstream features are estimated as $\hat{X}_G = F_t(X_G)$. Pooled GLCM features are computed as $\hat{X}_G = F_t(F_s(X_G))$, where F_s is the spatial pooling among patches.

All extracted features are concatenated into one feature vector, and Forward-Sequential Feature Selection (F-SFS) [25] selects 15 features. The number of features is determined based on a trade-off between complexity and accuracy. The selected features are shown in Table 1. These features are fed into a Support Vector Regression (SVR) for predicting the minimum CRF (c_T) where noticeable quality distortion (first JND) is observed.

[6] https://github.com/Telecommunication-Telemedia-Assessment/bitstream_mode3_videoparser, last access: Apr 30, 2023.

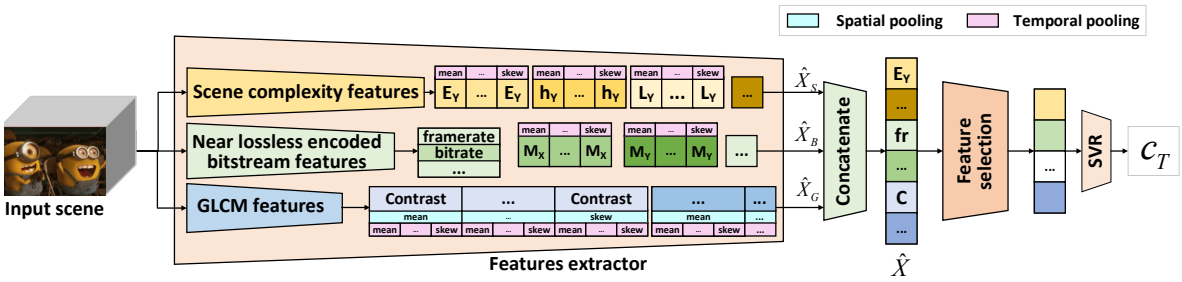


Fig. 4: JND threshold prediction model architecture.

Table 1: List of the fifteen features selected by F-SFS.

$\hat{X}_S = F_t(X_S)$	$\hat{X}_B = F_t(X_B)$	$\hat{X}_G = F_t(F_s(X_G))$
max(L_Y)	kurt(AvMotionX)	mean(mean(dissimilarity))
max(L_U)	kurt(AvMotionY)	kurt(kurt(dissimilarity))
	kurt(SpatialComplexity)	max(mean(homogeneity))
		mean(mean(homogeneity))
		skew(std(angular second moment))
		kurt(std(angular second moment))
		kurt(skew(angular second moment))
		mean(skew((energy))
		std(max((correlation))
		kurt(max((contrast))

Representation elimination: c_T is used to eliminate perceptually redundant representations from the bitrate ladder as shown in Algorithm 2. There shall be only one representation in the bitrate ladder where the selected optimized resolution is the maximum supported resolution (r_{max}), and the predicted optimized CRF is lower than c_T. Other higher bitrate representations are eliminated.

Algorithm 2: Representation elimination

Inputs:
N : number of bitrates in B
$(\hat{r}, b, \hat{c})$ pairs of the bitrate ladder
c_T : JND threshold CRF
r_{max} : maximum resolution in R
Output: $(\hat{r}, b, \hat{c})$ pairs for encoding
$t = 1, flag = 0$
while $t \leq N$ **do**
  if $\hat{r}_t == r_{max}$ *and* $\hat{c}_t < c_T$ **then**
    $flag++$
  if *flag* > 1 **then**
    Eliminate $(\hat{r}_t, b_t, \hat{c}_t)$ from the ladder.
  $t++$

3. EVALUATION

3.1. Test Methodology

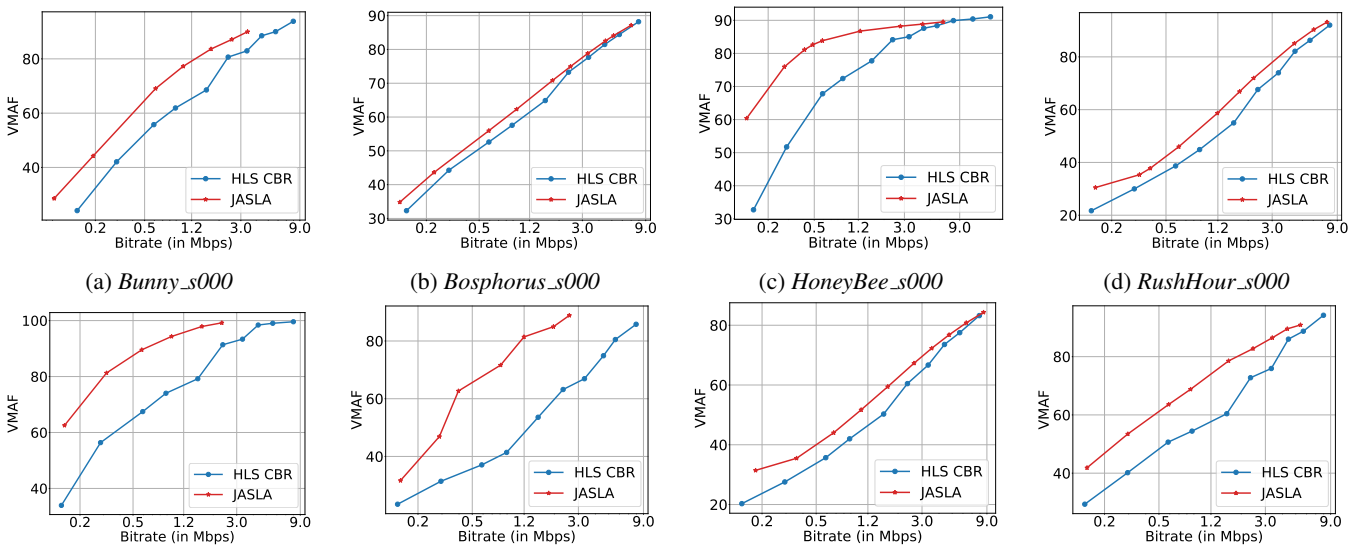
In this paper, four hundred video sequences (*i.e.*, 80% of all sequences) from the Video Complexity Dataset [4] are used as the training dataset, and the remaining (20%) is used as the test dataset. The video sequences are encoded at 30fps using x265[7] v3.5 with the *slower* preset. The bitrate-ladder specified in Apple HLS authoring specifications[1] are considered in the evaluation, *i.e.*, R= {360p, 432p, 540p, 720p, 1080p} and B = {145, 300, 600, 900, 1600, 2400, 3400, 4500, 5800, 8100}. E_Y, h and L_Y features are extracted using VCA[5] v1.5 open-source video complexity analyzer [18] run in eight CPU threads using x86 SIMD optimization [26]. Hyperparameter tuning is performed to obtain a balance between the model size and performance for VMAF and CRF prediction models, which results in the following parameters[8] for VMAF and CRF prediction models: *min_samples_leaf* = 1, *min_samples_split* = 2, *n_estimators* = 100, and *max_depth* = 14. Furthermore, the bitstream features are extracted from the CRF=5 encoded bitstream for each scene. $Q \times Q$ is set as 64×64 to determine GLCM features. The JND prediction model is trained on HD-VJND datasets [13] for the Full HD (1080p) resolution by five-fold cross-validation. The kernel of SVR is the Radial basis function[9] with the parameters $\epsilon = 0.0001$ and regularization parameter $C = 0.1$ determined by a greedy hyperparameter search.

The following metrics are considered during the evaluation: *(i)* quality in terms of PSNR and VMAF[2], *(ii)* bitrate, and *(iii)* encoding time. Since the content is assumed to be displayed at Full HD (1080p) resolution [3], the encoded content is scaled to 1080p resolution, and VMAF and PSNR are calculated. Bjøntegaard delta rates [27] BDR_P and BDR_V refer to the average increase in bitrate of the representations compared with that of the fixed bitrate ladder encoding to maintain the same PSNR and VMAF, respectively. BD-PSNR and BD-VMAF refer to the average increase in PSNR and VMAF, respectively, at the same bitrate compared with the reference bitrate ladder encoding scheme. The relative difference in the storage space required to store all representations (ΔS) is also evaluated as:

$$\Delta S = \frac{\sum b_{opt}}{\sum b_{ref}} - 1 \qquad (1)$$

where $\sum b_{ref}$ and $\sum b_{opt}$ represent the sum of bitrates of all representations in the reference bitrate ladder encoding and JASLA encoding, respectively.

[7] https://videolan.org/developers/x265.html, last access: Apr 30, 2023.
[8] https://scikit-learn.org/stable/modules/ensemble.html#forests-of-randomized-trees, Last access: Apr 30, 2023.
[9] https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVR.html, Last access: Apr 30, 2023.



(a) *Bunny_s000* (b) *Bosphorus_s000* (c) *HoneyBee_s000* (d) *RushHour_s000*

(e) *Characters_s000* (f) *Eldorado_s005* (g) *Runners_s000* (h) *Wood_s000*

Fig. 5: Comparison of RD curves of representative scenes (a) *Bunny_s000* (E_Y =22.40, h=4.70, L_Y=129.21), (b) *Bosphorus_s000* (E_Y=26.77, h=16.08, L_Y=140.54), (c) *HoneyBee_s000* (E_Y=42.93, h=7.91, L_Y=103.00), (d) *RushHour_s000* (E_Y =47.75, h=19.70, L_Y=101.66), (e) *Characters_s000* (E_Y =45.42, h=36.88, L_Y=134.56), (f) *Eldorado_s005* (E_Y =100.37, h=9.23, L_Y=109.06), (g) *Runners_s000* (E_Y =105.85, h=22.48, L_Y=126.60), (h) *Wood_s000* (E_Y=124.72, h=47.03, L_Y=119.57) using HLS CBR encoding (blue line), JASLA encoding (red line).

3.2. Experimental Results

The performance of the VMAF, CRF, and JND threshold prediction models is investigated in the first experiment. The average R^2 score of the VMAF and CRF prediction models are estimated as 0.93 and 0.97, respectively. Hence, a strong positive correlation exists between the predicted and ground truth values. The average MAE of the prediction models is estimated as 3.25 and 1.86, respectively. The MAE of the JND threshold prediction model is observed to be 0.96, which shows that JASLA works with sufficient prediction accuracy.

The second experiment analyzes the runtime complexity of JASLA. JASLA predicts resolution and CRF at a rate of 300 frames per second, *i.e.*, 0.4s per video segment. Compared to [13], the JND prediction runtime in JASLA is decreased by 97.24%.

The third experiment analyzes the bitrate saving and storage reduction results of JASLA compared to the HLS CBR encoding. Using JASLA encoding, BDR_P, BDR_V, and ΔS are observed as -34.42%, -42.67% and -54.34%, respectively, compared to the HLS CBR encoding. Moreover, JASLA encoding yields an average BD-PSNR and BD-VMAF of 2.90 dB and 9.51, respectively. Fig. 5 shows the RD curves of eight representative video sequences (scenes) with HLS CBR encoding and JASLA encoding. The representative scenes exhibit a variety of spatial and temporal complexities (in terms of E_Y, h, and L_Y). JASLA yields the highest VMAF at the same target bitrates for all scenes. Moreover, the perceptually lossless representations are eliminated from the bitrate ladder.

4. CONCLUSIONS

This paper proposes a JND-aware per-scene bitrate ladder prediction scheme (JASLA) for adaptive video-on-demand streaming applications. JASLA predicts the optimized resolution and corresponding CRF for given target bitrates for every video scene based on content-aware spatial and temporal complexity features. A JND threshold prediction scheme is proposed, eliminating representations that yield distortion lower than one JND from the bitrate ladder. The performance of JASLA is analyzed using the x265 open-source HEVC encoder against a standard HLS bitrate ladder with the maximum resolution of Full HD (1080p). It is observed that, on average, streaming using JASLA requires 34.42% and 42.67% fewer bits to maintain the same PSNR and VMAF, respectively, compared to the reference HLS bitrate ladder, along with a 54.34% cumulative decrease in the storage space needed to store representations.

JASLA shall be extended in the future by preparing a JND prediction model for Ultra HD (2160p) videos, thereby enabling the use of JASLA in UHD adaptive streaming.

5. ACKNOWLEDGMENT

The financial support of the Austrian Federal Ministry for Digital and Economic Affairs, the National Foundation for Research, Technology and Development, and the Christian Doppler Research Association is gratefully acknowledged. Christian Doppler Laboratory ATHENA: https://athena.itec.aau.at/.

6. REFERENCES

[1] Cisco, "Cisco Visual Networking Index: Forecast and Trends, 2017–2022," *White Paper*, February 2019.

[2] A. Bentaleb *et al.*, "A Survey on Bitrate Adaptation Schemes for Streaming Media Over HTTP," *IEEE Communications Surveys Tutorials*, vol. 21, no. 1, pp. 562–585, 2019.

[3] J. De Cock *et al.*, "Complexity-based consistent-quality encoding in the cloud," in *IEEE International Conference on Image Processing (ICIP)*, 2016, pp. 1484–1488.

[4] H. Amirpour *et al.*, "VCD: Video Complexity Dataset," in *Proceedings of the 13th ACM Multimedia Systems Conference*, 2022.

[5] V. V. Menon *et al.*, "JND-aware Two-pass Per-title Encoding Scheme for Adaptive Live Streaming," 2023. [Online]. Available: https://www.techrxiv.org/articles/preprint/JND-aware_Two-pass_Per-title_Encoding_Scheme_for_Adaptive_Live_Streaming/22256704

[6] A. V. Katsenou *et al.*, "Content-gnostic bitrate ladder prediction for adaptive video streaming," in *Picture Coding Symposium (PCS)*, 2019.

[7] M. Bhat *et al.*, "Combining Video Quality Metrics To Select Perceptually Accurate Resolution In A Wide Quality Range: A Case Study," in *2021 IEEE International Conference on Image Processing (ICIP)*, 2021, pp. 2164–2168.

[8] T. Huang *et al.*, "Deep Reinforced Bitrate Ladders for Adaptive Video Streaming." New York, NY, USA: Association for Computing Machinery, 2021, p. 66–73.

[9] H. Wang *et al.*, "Prediction of satisfied user ratio for compressed video," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 6747–6751.

[10] S. Hu *et al.*, "Compressed image quality metric based on perceptually weighted distortion," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5594–5608, 2015.

[11] Y. Zhang *et al.*, "Deep learning based just noticeable difference and perceptual quality prediction models for compressed video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 3, pp. 1197–1212, 2022.

[12] J. Zhu *et al.*, "On The Benefit of Parameter-Driven Approaches for the Modeling and the Prediction of Satisfied User Ratio for Compressed Video," in *IEEE International Conference on Image Processing (ICIP)*, 2022, pp. 4213–4217.

[13] J. Zhu *et al.*, "Subjective test methodology optimization and prediction framework for Just Noticeable Difference and Satisfied User Ratio for compressed HD video," in *2022 Picture Coding Symposium*, 2022.

[14] J. You and J. Korhonen, "Deep Neural Networks for No-Reference Video Quality Assessment," in *2019 IEEE International Conference on Image Processing (ICIP)*, 2019, pp. 2349–2353.

[15] V. V. Menon *et al.*, "INCEPT: Intra CU Depth Prediction for HEVC," in *2021 IEEE 23rd International Workshop on Multimedia Signal Processing (MMSP)*, 2021, pp. 1–6.

[16] V. V. Menon *et al.*, "Perceptually-aware per-title encoding for adaptive video streaming," in *2022 IEEE International Conference on Multimedia and Expo (ICME)*, 2022, pp. 1–6.

[17] V. V. Menon *et al.*, "Transcoding Quality Prediction for Adaptive Video Streaming," 2023. [Online]. Available: https://arxiv.org/abs/2304.10234

[18] V. V. Menon *et al.*, "Green Video Complexity Analysis for Efficient Encoding in Adaptive Video Streaming," in *First International ACM Green Multimedia Systems Workshop (GMSys '23)*, 2023.

[19] N. B. Harikrishnan *et al.*, "Comparative evaluation of image compression techniques," in *2017 International Conference on Algorithms, Methodology, Models and Applications in Emerging Technologies (ICAMMAET)*, 2017, pp. 1–4.

[20] V. V. Menon, "Video Coding Enhancements for HTTP Adaptive Streaming," in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022, p. 6905–6909.

[21] V. V. Menon *et al.*, "Video Quality Assessment with Texture Information Fusion for Streaming Applications," 2023. [Online]. Available: https://arxiv.org/abs/2302.14465

[22] A. Raake *et al.*, "A bitstream-based, scalable video-quality model for HTTP adaptive streaming: ITU-T P.1203.1," in *2017 Ninth International Conference on Quality of Multimedia Experience (QoMEX)*, 2017, pp. 1–6.

[23] R. R. R. Rao *et al.*, "Bitstream-Based Model Standard for 4K/UHD: ITU-T P.1204.3 — Model Details, Evaluation, Analysis and Open Source Implementation," in *2020 Twelfth International Conference on Quality of Multimedia Experience (QoMEX)*, 2020, pp. 1–6.

[24] R. M. Haralick *et al.*, "Textural features for image classification," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-3, no. 6, pp. 610–621, 1973.

[25] F. J. Ferri, P. Pudil, M. Hatef, and J. Kittler, "Comparative study of techniques for large-scale feature selection," in *Machine Intelligence and Pattern Recognition*. Elsevier, 1994, vol. 16, pp. 403–413.

[26] P. K. Tiwari *et al.*, "Accelerating x265 with Intel® Advanced Vector Extensions 512," *White Paper on the Intel Developers Page*, 2018. [Online]. Available: https://www.intel.com/content/dam/develop/external/us/en/documents/mcw-intel-x265-avx512.pdf

[27] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," *VCEG-M33*, 2001.