LiveVBR: Energy-efficient Live Per-title Encoding for Adaptive Video Streaming

Vignesh V Menon\textsuperscript{1}, Prajit T Rajendran\textsuperscript{2}
\textsuperscript{1}Christian Doppler Laboratory ATHENA, Alpen-Adria-Universität, Klagenfurt, Austria
\textsuperscript{2}Université Paris-Saclay, CEA, List, F-91120, Palaiseau, France

30 June 2023
Outline

1. Research Goal
2. Live-VBR
3. Results
4. Summary
Research Goal

Joint optimization:
- Perceptual difference of pre-defined $v_J$ between representations.
- Minimize bitrate difference between representations.
- Maximize compression efficiency of representations.

Figure: The ideal perceptually-aware bitrate ladder envisioned in this work. Here, $v_J(v_0) = v_J(v_1) = v_J(v_{M-1}) = \Delta VMAF$
Workflow of Live-VBR

Figure: Live HTTP adaptive streaming featuring Live-VBR.
Video Complexity Feature Extraction

- Accomplished using VCA.¹
  - $E$: the average texture energy
  - $h$: the average gradient of the texture energy
  - $L$: the average luminescence

Figure: Example heatmap of Luminescence ($L$), spatial texture ($E$) and temporal activity ($h$) features of the 2nd frame of CoverSong_1080P_0a86 video of Youtube UGC dataset extracted using VCA.

Live-VBR

First point of the bitrate ladder

\[ \hat{b}_1 = b_{\text{min}} \]

Determine \( \hat{v}_r, \hat{b}_1 \) \( \forall r \in R \)

\[ \hat{v}_1 = \max(\hat{v}_r, \hat{b}_1) \]

\[ \hat{r}_1 = \arg \max_{r \in R}(\hat{v}_r, \hat{b}_1) \]

\((\hat{r}_1, \hat{b}_1)\) is the first point of the bitrate ladder.

---

Phase 2: Bitrate ladder Estimation

**Remaining points of the bitrate ladder**

![Diagram](image)

**Figure:** Estimation of the $t^{th}$ point of the bitrate ladder.

$t = 2$

while $\hat{b}_{t-1} < b_{max}$ and $\hat{v}_{t-1} < v_{max}$ do

\[
\hat{v}_t = \hat{v}_{t-1} + v_J(\hat{v}_{t-1})
\]

Determine $\hat{b}_r, \hat{v}_t \ \forall r \in R$

\[
\hat{b}_t = \min(\hat{b}_r, \hat{v}_t)
\]

\[
\hat{r}_t = \arg \min_{r \in R}(\hat{b}_r, \hat{v}_t)
\]

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

$t = t + 1$

---

Live-VBR

cVBR encoding of the bitrate ladder\textsuperscript{4}

\textbf{Figure}: Estimation of the optimized CRF to achieve the target bitrate $\hat{b}_t$ using a prediction model trained for resolution $\hat{r}_t$.

- Optimized CRF is determined for the selected $(\hat{r}_t, \hat{b}_t)$ pairs.
- cVBR encoding for the $(\hat{r}_t, \hat{b}_t, \hat{c}_t)$ pairs is performed.

Prediction models

Design

- **VMAF Prediction models**
  - Input: Video complexity features \((E, h, L)\), target bitrate \(\hat{b}_t\)
  - Output: VMAF score

- **Bitrate Prediction models**
  - Input: Video complexity features \((E, h, L)\), predicted VMAF \(\hat{v}_t\)
  - Output: Predicted bitrate \(\hat{b}_t\)

- **CRF Prediction models**
  - Input: Video complexity features \((E, h, L)\), target bitrate \(\hat{b}_t\)
  - Output: CRF value
Prediction models
Architecture and training

- Implementation using Python
- Dataset used for training: VCD dataset\textsuperscript{5} @ 30 fps
- Train-validation-test split: 70-10-20, clustered based on feature space
- Prediction model architecture: Random forest (RF) models
- Advantages of RF models: Good prediction performance, lower variance, robust to outlier
- Modular approach to ensure scalability

Results

Prediction accuracy of the models

Table: $R^2$ score and MAE of the prediction models for various resolutions.

<table>
<thead>
<tr>
<th></th>
<th>360p</th>
<th>432p</th>
<th>540p</th>
<th>720p</th>
<th>1080p</th>
<th>2160p</th>
<th>360p</th>
<th>432p</th>
<th>540p</th>
<th>720p</th>
<th>1080p</th>
<th>1440p</th>
<th>2160p</th>
</tr>
</thead>
<tbody>
<tr>
<td>VMAF</td>
<td>0.821</td>
<td>0.852</td>
<td>0.882</td>
<td>0.906</td>
<td>0.910</td>
<td>0.906</td>
<td>0.930</td>
<td>5.091</td>
<td>5.071</td>
<td>4.966</td>
<td>4.971</td>
<td>4.806</td>
<td>4.490</td>
</tr>
<tr>
<td>log(b)</td>
<td>0.867</td>
<td>0.884</td>
<td>0.901</td>
<td>0.910</td>
<td>0.932</td>
<td>0.937</td>
<td>0.943</td>
<td>0.527</td>
<td>0.505</td>
<td>0.472</td>
<td>0.456</td>
<td>0.460</td>
<td>0.472</td>
</tr>
<tr>
<td>CRF</td>
<td>0.969</td>
<td>0.969</td>
<td>0.970</td>
<td>0.969</td>
<td>0.968</td>
<td>0.967</td>
<td>0.968</td>
<td>1.823</td>
<td>1.820</td>
<td>1.820</td>
<td>1.859</td>
<td>1.860</td>
<td>1.885</td>
</tr>
</tbody>
</table>

Figure: The relative importance of (a) VMAF prediction, (b) bitrate prediction, and (c) CRF prediction for 2160p resolution determined by SHAP values.
Results

Figure: Pre-processing time ($\tau_p$) of Live-VBR for various input video resolutions.
Results

RD plots of Live-VBR using x265

Figure: RD curves of representative video sequences (segments) (a) Bunny_s000 ($E = 22.40$, $h=4.70$, $L=129.21$), (b) Characters_s000 ($E = 45.42$, $h=36.88$, $L=134.56$), (c) Eldorado_s000 ($E = 15.28$, $h=49.76$, $L=140.54$) using the HLS CBR encoding (green line), and Live-VBR encoding (red line). JND is considered as 6 VMAF in these plots.
### Results

**RD plots of Live-VBR using x265**

Figure: RD curves of representative video sequences (segments) (a) *Eldorado_s005* \( (E = 100.37, \ h = 9.23, \ L = 109.06) \), (b) *HoneyBee_s000* \( (E = 42.93, \ h = 7.91, \ L = 103.00) \), (c) *Wood_s000* \( (E = 124.72, \ h = 47.03, \ L = 119.57) \) using the HLS CBR encoding (green line), and Live-VBR encoding (red line). JND is considered as 6 VMAF in these plots.
Table: Average results of the encoding schemes compared to the HLS CBR encoding.

<table>
<thead>
<tr>
<th>Method</th>
<th>$\text{BDR}_p$</th>
<th>$\text{BDR}_v$</th>
<th>$\text{BD-PSNR}$</th>
<th>$\text{BD-VMAF}$</th>
<th>$\Delta S$</th>
<th>$\Delta T$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bruteforce (2 VMAF JND)$^6$</td>
<td>-23.09%</td>
<td>-43.23%</td>
<td>1.34 dB</td>
<td>10.61</td>
<td>-25.99%</td>
<td>4732.33%</td>
</tr>
<tr>
<td>Bruteforce (4 VMAF JND)</td>
<td>-28.15%</td>
<td>-42.75%</td>
<td>1.70 dB</td>
<td>10.08</td>
<td>-59.07%</td>
<td>4732.33%</td>
</tr>
<tr>
<td>Bruteforce (6 VMAF JND)</td>
<td>-25.36%</td>
<td>-40.73%</td>
<td>1.67 dB</td>
<td>9.19</td>
<td>-70.50%</td>
<td>4732.33%</td>
</tr>
<tr>
<td>Live-VBR (2 VMAF JND)</td>
<td>-14.25%</td>
<td>-29.14%</td>
<td>1.36 dB</td>
<td>7.82</td>
<td>23.57%</td>
<td>184.62%</td>
</tr>
<tr>
<td>Live-VBR (4 VMAF JND)</td>
<td>-18.41%</td>
<td>-32.48%</td>
<td>1.41 dB</td>
<td>8.31</td>
<td>-56.38%</td>
<td>26.14%</td>
</tr>
<tr>
<td>Live-VBR (6 VMAF JND)</td>
<td>-18.80%</td>
<td>-32.59%</td>
<td>1.34 dB</td>
<td>8.34</td>
<td>-68.96%</td>
<td>-18.58%</td>
</tr>
</tbody>
</table>

Relative storage difference

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

Relative difference in the encoding time

$$\Delta T = \frac{\tau_{p} + \sum t_{opt}}{\sum t_{ref}} - 1$$

Summary and Future Directions

- Introduced an optimized encoding bitrate ladder prediction scheme, which uses RF-based models to estimate bitrate-resolution-CRF triples for a given video segment based on its spatial and temporal characteristics.
- The bitrate ladder is predicted such that there is a perceptual difference of at least one JND between the representations.
- The prototype implementation code is released at: https://github.com/cd-athena/LiveVBR.
- The online documentation can be found at: https://cd-athena.github.io/LiveVBR/.

Further details:
Thank you for your attention!

Vignesh V Menon (vignesh.menon@aau.at)
Prajit T Rajendran (prajit.thazhurazhikath@cea.fr)