YOCO: LIGHT-WEIGHT RATE CONTROL MODEL LEARNING

Yangfan Sun, Li Li, Zhu Li

University of Missouri-Kansas City
Dept. of Engineering and Computer Science
5100 Rockhill Rd., Kansas City, MO 64110
ysb5b@umsystem.edu, {lil1, lizhu}@umkc.edu

Shan Liu

Tencent America
661 Bryant St
Palo Alto, CA 94301
shanl@tencent.com

ABSTRACT

The knowledge of the bitrates and coding parameters is the prerequisite for implementing the rate control mechanism. Many previous works have been attempted to avoid the actual coding to obtain these factors by studying their regularity of correlation. However, these works only focus on the studies on the picture level or the coding unit (CU) level rather than on the sequence level. In this paper, we propose the YOCO (You Only Code Once) light-weight rate control model learning scheme, which can achieve the sequence-level rate control by managing the constant rate factor (CRF). It utilizes the unified information extracted from the compressed videos in the bitstream and pixel domains, leveraging the deep learning algorithm to learn the rate control model as these factors' algebraic relevance. For each video sequence, we can allocate the bitrates extremely approaching the target value with coding at the estimated CRF setting. We compare the application of pixel domain information on each rate control model (linear and the quadratic R-CRF models) to validate their effectiveness. The experimental results demonstrate the improvement of accuracy on the bitrate estimation in the different definitions, especially in the high ones.

***Index Terms*—** Bitstream domain, Constant rate factor (CRF), Pixel domain, Rate control, Sequence level

1. INTRODUCTION

As a necessary mechanism of an encoder (such as H.265/HEVC [1]), rate control is responsible for the best video quality subject to the bitrate constraint. An accurate estimation of the correlation between the bitrate and coding parameters [2] in the different coding levels is crucial for the rate control mechanism. The picture-level [3] [4] and the coding unit (CU)-level [5] [6] rate control dominantly attract most attentions. However, these works have an inevitable disadvantage of requiring the adjustment of macroblock (MB)-level inner algorithms to the deployed codec. In order to avoid this drawback, we aim at developing a sequence-level rate control scheme, which can be adapted into different codecs with different inner structures.

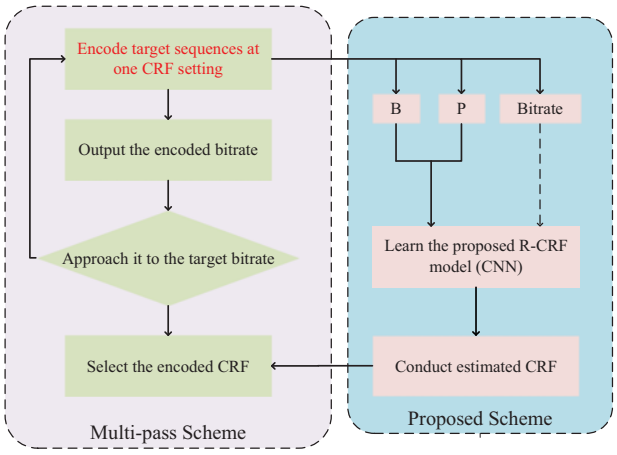

Fig. 1. Framework of the multi-pass and proposed schemes.

The sequence-level rate control can be achieved by adjusting the sequence-level coding parameters. For example, in x265, the coding parameter can be chosen from the constant quantization parameter (CQP) or constant rate factor (CRF). CRF outperforms CQP at the bitrate usage since it considers video content complexity and motions in the quantization parameter (QP) adjustment strategy for each frame [7]. However, the relationship between bitrate and CRF is harder to estimate than the relationship between bitrate and CQP. A multi-pass encoding scheme is usually needed to obtain an accurate relationship between bitrate and CRF, which prevents its usage in the low-latency video communication scenarios.

Many attempts have been done to avoid the excessive computational complexity of multi-pass processing. Covell et al. [8] proposed a one-pass neural network (NN)-based rate control scheme through a linear rate control model. Sun et al. [9] improved [8] the estimated performance by presenting a quadratic resolution-independent rate control model. However, both of them simply concentrated on the bitstream domain information, while completely ignoring the pixel domain information. Xu et al. [10] proposed a convolutional neural network (CNN) based frame-level method to train

Authorized licensed use limited to: Auckland University of Technology. Downloaded on October 29,2020 at 10:52:33 UTC from IEEE Xplore. Restrictions apply.

rate-distortion (R-D) relationship relying on images directly. Although their research was not the sequence-level rate control scheme, it shows the potential effectiveness of the pixel domain information. Motivated by them, we try to add the pixel domain information to the rate control scheme in the sequence level.

In this paper, we propose a light-weight rate control model learning scheme aiming at the estimation of the bitrate and the corresponding CRF by leveraging a novel neural network. We use the pixel domain features from the fingernail differential frames without breaking the consistency of the bitstream domain features. Fig. 1 shows the comparison between the multi-pass and the proposed schemes. For the proposed scheme, the first coding pass is used to extract the bitstream domain features. The pixel domain features are extracted from the non-coding process that enhances the characteristics of the video content.

This paper is organized as follows. In Section 2, the framework of our proposed scheme and the derivation of pixel-domain features will be introduced in detail. In Section 3, the architecture of deep learning networks and its hyper-parameters will be discussed. In Section 4, we will show the experimental results. Section 5 will conclude the paper.

2. PROPOSED RATE CONTROL SCHEME

In this section, we will explain the proposed rate control scheme in details. Our method estimates the CRF with the given encoded bitrate by taking advantage of the rate control model, along with the hybrid compressed features. We denote the proposed scheme as $f(.)$ to estimate the predicted $C\hat{R}F$ by using the bitstream domain features B, the pixel domain features P, and the target bitrate R as input,

$$C\hat{R}F = f([B, P], R). \tag{1}$$

The complete scheme can be split into two subtasks. The first one is the model parameters estimation $t(.)$ that uses the hybrid compressed features and the network trainable parameters Θ to predict the estimated model parameters $\hat{Y}$, generating the R-CRF model. The second one is the coding parameter estimation $m(.)$ that calculates the predicted rate $\hat{R}$ in accordance with the generated model. As shown,

$$\begin{cases} \hat{Y} = t([B, P], \Theta) \\ C\hat{R}F = m(\hat{Y}, R). \end{cases} \tag{2}$$

The predefined pixel-wise loss function is used to guide the convergence of Θ.

$$L(\Theta) = \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2, \tag{3}$$

where Y and $\hat{Y}$ denote the sets of the ground truth and the estimated model parameters, respectively.

2.1. Coding Parameters Estimation

First, we introduce the procedure of the coding parameters estimation. It leverages the rate control model to implement the estimated process. Two types of the rate control model formula, including the linear [8] and quadratic R-CRF model [9], are tested in order to select the optimal model,

$$C\hat{R}F = \begin{cases} \hat{Y_1} \cdot ln(R) + \hat{Y_2},\ \hat{Y} = [\hat{Y_1}, \hat{Y_2}] \\ \hat{Y_1} \cdot (ln(R))^2 + \hat{Y_2} \cdot ln(R) + \hat{Y_3},\ \hat{Y} = [\hat{Y_1}, \hat{Y_2}, \hat{Y_3}] \end{cases} \tag{4}$$

Different number of parameters need to be estimated as shown in Equation (4). Then, the estimated parameters can be used to calculate the predicted CRF with the given target bitrate R. In addition, the inclusion and exclusion situations of the pixel domain features P are investigated to validate the advantage of pixel information.

2.2. Model Parameters Estimation

Second, the rate control model parameters can be learned at this subtask with the assistance of the proposed neural network. The network input includes the bitstream domain features B and the pixel domain features P. The percentage of I/B/P frames, the number of macroblocks (MB), texture bits, motion vectors, peak signal to noise ratio (PSNR), and the real bitrate are covered in the former features set from one-pass encoding.

The pixel domain features P are used to compensate for the pixel information and have the following three characteristics. First, the pixel domain features P are able to provide rich spatial and temporal information to describe the video content. Second, they can be obtained with the low time consumption (no need to implement the encoding process) to be used in the low-latency video communication scenarios. Third, they have a similar format with the bitstream domain features B. Therefore, we generate them from thumbnail size successive differential video frames.

The initial step of P extraction is to scale the reconstructed video frames $F_k \in \mathbb{R}^{H \times W}$ to the desired thumbnail frames $S_k \in \mathbb{R}^{h \times w}$. In our experiments, h and w are set as 16 and 12, respectively. The thumbnail frames S_k are appropriate enough to represent the original frames, however, require much fewer computational complexity for the subsequence operations. Then, we obtain the differential thumbnail frames $D(S_k, S_{k+1})$ to remove the temporal redundancy between the successive frames,

$$D(S_k, S_{k+1}) = \begin{cases} 0\ , k = 1 \\ S_k - S_{k+1}\ ,\ else, \end{cases} \tag{5}$$

where k denotes the number of frames. After that, we project the differential thumbnail frames $D(S_k, S_{k+1})$ on a subspace A to preserve the maximum amount of information with a limited number of dimensions for further compression. The

Authorized licensed use limited to: Auckland University of Technology. Downloaded on October 29,2020 at 10:52:33 UTC from IEEE Xplore. Restrictions apply.

principle component analysis (PCA) is used in this paper to localize the subspace A as follows,

$$[A, \lambda] = eig(D(S_k, S_{k+1})^T D(S_k, S_{k+1})), \quad (6)$$

where $eig(.)$ denotes the generalized eigenproblem formulation. The λ and A are the eigenvalues and eigenvectors, respectively. The value of λ determines the amount of information contained in the corresponding eigenvectors. We keep the N (N=8) dimensions of eigenvectors with the top eigenvalues to carry the most useful information. Subsequently, the differential thumbnail frames $D(S_k, S_{k+1})$ are projected on the subspace A to reduce the spatial dimension. The Euclidean Norm p_k as the representative of $D(S_k, S_{k+1})$ is denoted by

$$p_k = \sum_{n=1}^{N} (D(S_k, S_{k+1})_n \times A)^2, \quad (7)$$

where n represents the number of dimensions. We assemble the p_k as the elements of vector $\vec{p} \in \mathbb{R}^{n \times 1}$. To be consistent with the bitstream domain features B, we derive the average $M(.)$ and the variance $V(.)$ from the $\vec{p}$ to express the characteristics of the video content. $M(.)$ represents the content complexity, while $V(.)$ illustrates the degree of scene changes. To describe the video content in a finer granularity, we also try to divide the complete video into several fragments (2,4 or 16 fragments). The best performance with relative less computational complexity can be seen from two fragments division. Therefore, we totally extract six pixel domain parameters P that represent the pixel information,

$$P = [M(\vec{p}), V(\vec{p}), M(\vec{p}_f), V(\vec{p}_f), M(\vec{p}_s), V(\vec{p}_s)]. \quad (8)$$

where $\vec{p}_f$ and $\vec{p}_s$ denote the parameters of the first and second half fragments. Fig. 2 shows the mean $M(\vec{p})$ and the variance $V(\vec{p})$ of 9 test sequences on the coordinate. We can observe a potential correlation between the pixel domain features and the content properties. Fig. 3 shows the details of pixel information in four special samples (Bus, Flowers, News and Container). These samples represent four different situations, which are HVHM (High Variance and High Mean), HVLM (High Variance and Low Mean), LVHM and LVLM. After the pixel domain features are obtained, we feed them into $t(.)$ operator to predict the corresponding model parameters $\hat{Y}$.

3. NEURAL NETWORK

Fig. 4 shows the architecture of the proposed neural network including three fully-connected layers of 100 neurons, 50 neurons, and the number of neurons corresponding to the number of model parameters, respectively. The input contains 13 bitstream domain features B, 6 pixel-domain features P and the encoded bitrate R. The label is the model parameters. The packages of $MinMaxScaler$ and $StandardScaler$ are utilized to preprocess the features and labels. The $ReLU$ is

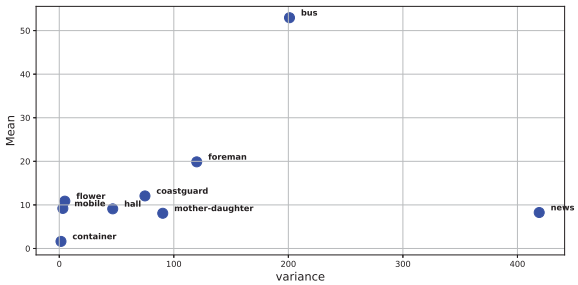

Fig. 2. The pixel domain features of typical testing sequences.

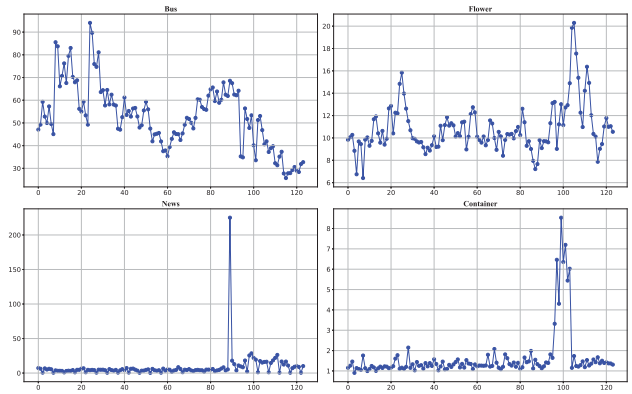

Fig. 3. Differential frames representative in pixel domain for four special test sequences.

used as the activate function of the first two fully-connected layers. We introduce $Adam$ as the optimizer. The learning rate is set as 0.005. The constant iterative cessation mechanism is employed and executed at the 30,000th iteration. We train the network on CPU only with 4 hours of training time cost.

4. EXPERIMENTS

In this section, a series of experiments is designed to evaluate the efficiency of the pixel-domain features. 28,381 video chips are split and collected from YouTube channel at various spatial resolutions (270p, 480p, 720p and 1080p). To enhance sample diversity, they are downloaded according to the scales of complexity and video materials, e.g., sport, entertainment, news, etc.. Each 5-second video clip contains 125 frames, encoded by x265. At the end of coding, we obtain the encoded bitrate R and corresponding features in accordance with CRF in the range of $[10, 42]$. One single CRF encoded features (CRF=26) including the bitstream and the pixel domain features are used as input. Then, the R-CRF model parameters Y are generated through the least square regression algorithm with the given R and the corresponding CRF. The ratio of training, validation, and test set is 80%, 10%, and 10%, re-

Authorized licensed use limited to: Auckland University of Technology. Downloaded on October 29,2020 at 10:52:33 UTC from IEEE Xplore. Restrictions apply.

Table 1. The model accuracy of various learning-based estimation methods on each spatial resolution

Estimated Model	$\delta_R \leq \pm 20\%$				$\delta_R \leq \pm 10\%$			
	270p	480p	720p	1080p	270p	480p	720p	1080p
Quadratic+P	96.13%	94.23%	92.92%	85.76%	82.80%	77.69%	73.32%	59.31%
Quadratic	95.77%	93.16%	87.90%	81.92%	79.78%	74.71%	65.66%	54.66%
Improvement	+0.36%	+1.07%	+5.02%	+3.84%	+3.02%	+2.98%	+7.66%	+4.65%
Avg.	+2.57%				+4.58%			
linear+P	95.33%	88.97%	84.66%	74.77%	74.63%	62.56%	53.88%	42.15%
linear	94.32%	88.72%	83.20%	72.15%	72.01%	60.72%	52.80%	39.90%
Improvement	+1.01%	+0.25%	+1.44%	+2.62%	+2.62%	+1.84%	+1.08%	+2.25%
Avg.	+1.33%				+1.95%			

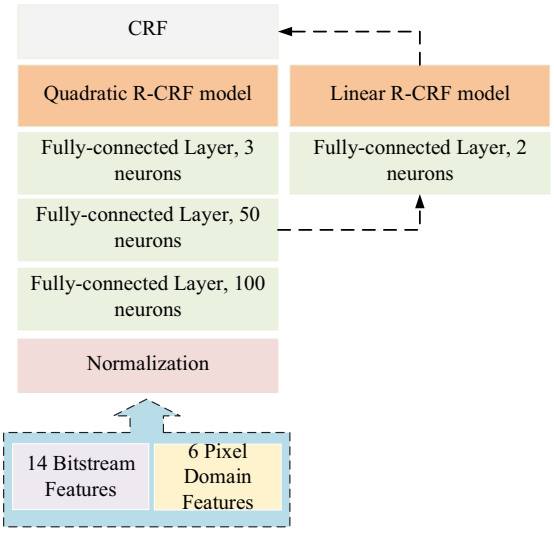

Fig. 4. The proposed neural network architecture.

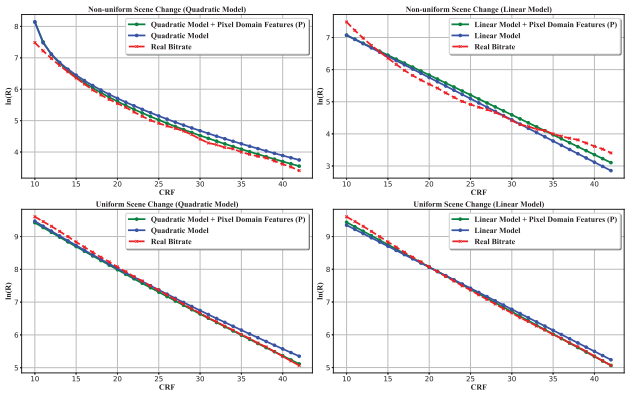

Fig. 5. The performance of R-CRF models with or w/o P in the uniform or non-uniform scene change situations, the top figures belong to the non-uniform scene change situation while the bottom figures belong to the uniform scene change situation.

spectively. We use the related bitrate error δ_R as the metric to evaluate the performance of the proposed algorithm,

$$\delta_R = \frac{R - \hat{R}}{R} \times 100\%. \tag{9}$$

Table 1 shows the performance of both quadratic and linear R-CRF models with or without the pixel domain features P. It is observed that the accuracy of CRF estimation is improved on both R-CRF models when involving the pixel domain features. On average, the accuracy of estimation with the proposed scheme increases by 4.58% and 2.57% at δ_R within 10% and 20% on the quadratic model, while it increases by 1.95% and 1.33% on the linear model. In high-resolution scenarios, the clearer improvement by using the quadratic model can be observed. The 7.66% and 5.02% estimated accuracy (720p) and the 4.65% and 3.84% estimated accuracy (1080p) increases at δ_R within 10% and 20%.

Fig. 5 visualizes the fitting performance of both models in the non-uniform and uniform scene change situations. Obviously, we observe a better fitting result in both situations by using the quadratic model along with the pixel domain features P. In the linear model scheme, the pixel domain features P affect the fitting performance in the uniform scene change situation positively, however, the performance shows no obvious improvement in the non-uniform scene change situation.

5. CONCLUSION

In this paper, we propose the YOCO light-weight rate control model learning that achieves accurate one-pass sequence-level rate control by adjusting the constant rate factor (CRF). We propose using hybrid features including the bitstream and pixel domain features as input to estimate the model parameters more accurately. We test the proposed hybrid features by using both linear and quadratic models. The experimental results show that clear performance improvement are obtained due to the hybrid features.

Authorized licensed use limited to: Auckland University of Technology. Downloaded on October 29,2020 at 10:52:33 UTC from IEEE Xplore. Restrictions apply.

6. REFERENCES

[1] Gary J Sullivan, Jens-Rainer Ohm, Woo-Jin Han, and Thomas Wiegand, "Overview of the high efficiency video coding (hevc) standard," *IEEE Transactions on circuits and systems for video technology*, vol. 22, no. 12, pp. 1649–1668, 2012.

[2] Siwei Ma, Wen Gao, and Yan Lu, "Rate-distortion analysis for h. 264/avc video coding and its application to rate control," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 12, pp. 1533–1544, 2005.

[3] Sudeng Hu, Hanli Wang, Sam Kwong, and Tiesong Zhao, "Frame level rate control for h. 264/avc with novel rate-quantization model," in *Multimedia and Expo (ICME), 2010 IEEE International Conference on*. IEEE, 2010, pp. 226–231.

[4] Lin Sun, Oscar C Au, Wei Dai, Yuanfang Guo, and Ruobing Zou, "An adaptive frame complexity based rate quantization model for intra-frame rate control of high efficiency video coding (hevc)," in *Signal & Information Processing Association Annual Summit and Conference (APSIPA ASC), 2012 Asia-Pacific*. IEEE, 2012, pp. 1–6.

[5] Junjun Si, Siwei Ma, and Wen Gao, "Efficient bit allocation and ctu level rate control for high efficiency video coding," in *Picture Coding Symposium (PCS), 2013*. IEEE, 2013, pp. 89–92.

[6] Shengxi Li, Mai Xu, Zulin Wang, and Xiaoyan Sun, "Optimal bit allocation for ctu level rate control in hevc," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 11, pp. 2409–2424, 2017.

[7] Werner Robitza, "Crf guide (constant rate factor in x264 and x265)," `https://slhck.info/video/2017/02/24/crf-guide.html`, Accessed: 2017-02-24.

[8] Michele Covell, Martín Arjovsky, Yao-chung Lin, and Anil Kokaram, "Optimizing transcoder quality targets using a neural network with an embedded bitrate model," *Electronic Imaging*, vol. 2016, no. 2, pp. 1–7, 2016.

[9] Yangfan Sun, Mouqing Jin, Li Li, and Zhu Li, "A machine learning approach to accurate sequence-level rate control scheme for video coding," in *2018 25th IEEE International Conference on Image Processing (ICIP)*. IEEE, 2018, pp. 1013–1017.

[10] Bin Xu, Xiang Pan, Yan Zhou, Yiming Li, Daiqin Yang, and Zhenzhong Chen, "Cnn-based rate-distortion modeling for h. 265/hevc," in *Visual Communications and Image Processing (VCIP), 2017 IEEE*. IEEE, 2017, pp. 1–4.

Authorized licensed use limited to: Auckland University of Technology. Downloaded on October 29,2020 at 10:52:33 UTC from IEEE Xplore. Restrictions apply.