LETTER

Temporal Correlation-Based End-to-End Rate Control in DCVC

Zhenglong YANG[†a)], ***Nonmember***, **Weihao DENG**[†], ***Student Member***, **Guozhong WANG**[††], **Tao FAN**[††], ***and* Yixi LUO**[†], ***Nonmembers***

SUMMARY Recent deep-learning-based video compression models have demonstrated superior performance over traditional codecs. However, few studies have focused on deep learning rate control. In this paper, end-to-end rate control is proposed for deep contextual video compression (DCVC). With the designed two-branch residual-based network, the optimal bit rate ratio is predicted according to the feature correlation of the adjacent frames. Then, the bit rate can be reasonably allocated for every frame by satisfying the temporal feature. To minimize the rate distortion (RD) cost, the optimal λ of the current frame can be obtained from a two-branch regression-based network using the temporal encoded information. The experimental results show that the achievable BD-rate (PSNR) and BD-rate (SSIM) of the proposed algorithm are −0.84% and −0.35%, respectively, with 2.25% rate control accuracy.

key words: *end-to-end rate control, DCVC, convolutional neural network, temporal correlation*

1. Introduction

Rate control is a critical part of video compression, particularly in bandwidth-limited tasks such as live and broadcast. In recent years, end-to-end image compression [1] has shown that coding outperforms the traditional image coding. Guo *et al.* [2] proposed the first end-to-end framework for video compression, where the key components of traditional video compression are replaced by end-to-end neural networks. To improve the end-to-end video compression, Li *et al.* [3] proposed a deep contextual video compression (DCVC) model, which leverages the high-dimensional context to carry rich information for high-frequency content and achieves higher video coding quality. Since bit allocation can directly affect the rate distortion (RD) performance, Erenetin *et al.* [4] exploited frame-level bit allocation for intra- and bi-directionally frames. However, bit allocation for every frame cannot find a suitable λ to decrease the RD cost, which makes the rate control scheme in deep learning video compression remain unfeasible. Li *et al.* [5] presented an R-D-λ rate control model for the learned video compression. However, the rate control parameters are still obtained via traditional methods.

In this paper, we focused on achieving end-to-end rate control by using a convolutional neural network (CNN) to obtain the optimal bit allocation and λ. The major contributions of this paper are as follows:

(1) A two-branch residual-based network is designed to predict the bit rate ratio where the temporal encoded parameters are treated as the coding feature vector. Then, the bit rate can be reasonably allocated to every frame according to the low- and high-level coding features extracted by the designed network.

(2) A two-branch regression-based network is designed to obtain the optimal λ. To effectively decrease the RD cost, the temporal encoded information and residual feature frame are used as the input vector for the network. In addition, a regression block is added to enhance the learning and expression ability of the network.

2. End-to-End Rate Control

2.1 Framework

For end-to-end rate control, the original frame is input into the two-branch residual-based network to optimize the bit rate ratio. Then, the bit rate can be reasonably allocated to every frame by considering the bit buffer. With the allocated bit of the frame, the optimal λ can be predicted by the two-branch regression-based network for the DCVC encoder. Figure 1 shows the framework of the end-to-end rate control.

2.2 Frame Bit Allocation

To fully utilize the temporal correlation, a two-branch struc-

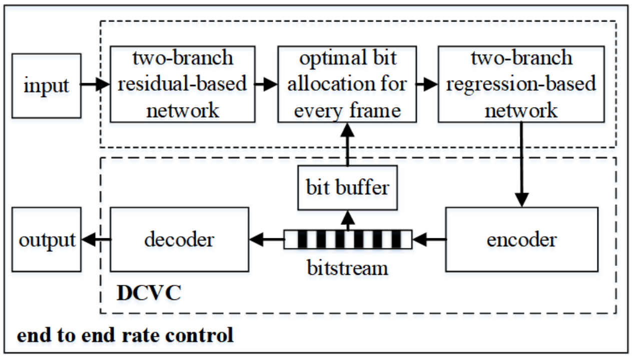
input
two-branch residual-based network
optimal bit allocation for every frame
two-branch regression-based network
bit buffer
output
decoder
bitstream
encoder
DCVC
end to end rate control

Fig. 1 End-to-end rate control framework.

Manuscript received May 3, 2024.
Manuscript revised July 4, 2024.
Manuscript publicized July 29, 2024.

[†] School of Urban Rail Transportation, Shanghai University of Engineering Science, Shanghai 201620, China.
[††] Artificial Intelligence Industry Research Institute, Shanghai University of Engineering Science, Shanghai 201620, China.
a) E-mail: 949713872@qq.com

DOI: 10.1587/transinf.2024EDL8041

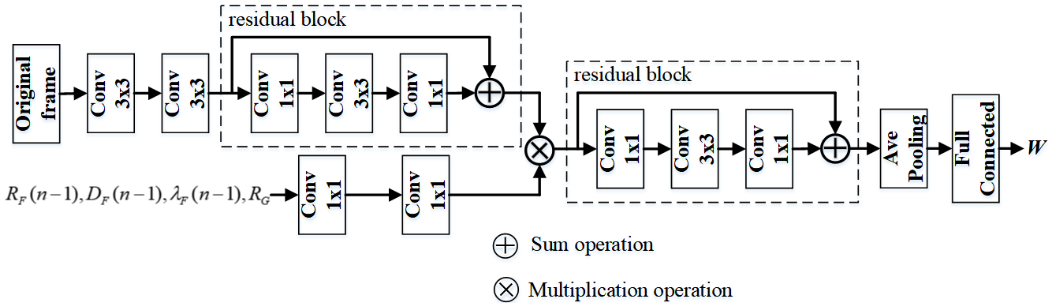

Fig. 2 Two-branch residual-based network.

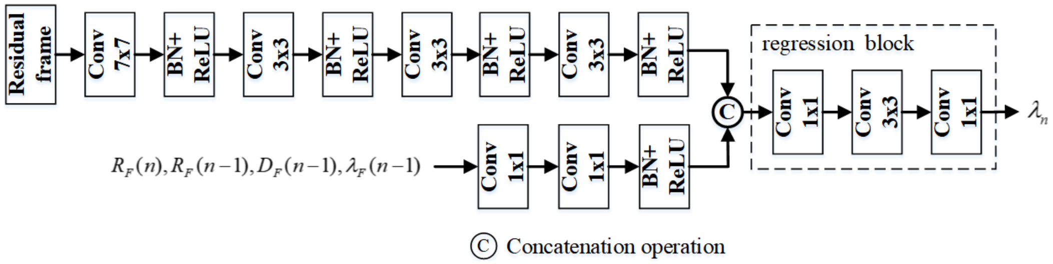

Fig. 3 Two-branch regression-based network.

ture of the network is used, as shown in Fig. 2. In Fig. 2, $R_F(n-1)$, $D_F(n-1)$ and $\lambda_F(n-1)$ are the bit rate, distortion and Lagrangian multiplier of the previous encoded frame, respectively. R_G is the target bit rate of the current group of pictures (GoP). W is the predicted bit rate ratio from the network. For the up branch of the network, the low-level features of the frame are extracted by the two convolutional layers with the 3×3 kernel. Then, the residual block extracts the high-level features. For the down branch of the network, the encoded information of the previous frame is input into the network. Since the features of the two branches have strong temporal correlation, a multiplication operation is used to fuse the temporal correlation. Finally, the fusion features are extracted and converted to predict the bit rate ratio W.

The GoP bit allocation R_G can be expressed as

$$R_G = \frac{R_{\text{target}} \cdot (n_{\text{encoded}} + N_{SW}) - R_{\text{encoded}}}{N_{SW}} \cdot N_G \tag{1}$$

where R_{target} and R_{encoded} are the target bit rate and total used bit rate, respectively; N_G is the number of frames in the GoP; n_{encoded} is the encoded frames; N_{SW} is the smooth window. Then, the bit allocation of frame n can be expressed as

$$R_F(n) = \frac{R_G - R_{\text{encoded–G}}}{\sum_{i=n}^{N_G} W_i} \cdot W_n \tag{2}$$

where $R_{\text{encoded–G}}$ is the used bit rate of the frames in the current GoP; W_n is the bit rate ratio of frame n, which can be predicted from the two-branch residual-based network. The loss function of the network is defined as

$$Loss_{ratio} = \frac{1}{N} \cdot \sum_{i=1}^{N} (W_i - \hat{W}_i)^2 \tag{3}$$

where W_i is the predicted bit rate ratio, $\hat{W}_i$ is the actual bit rate ratio, and N is the number of frames for training.

2.3 Optimal λ Decision

Figure 3 shows the structure of the two-branch regression network to predict λ. Since the residual feature, which is the difference between predicted frame and original frame, can indicate the correlation of the adjacent frames, the residual frame will be used as the up input. The bit allocation of the current frame $R_F(n)$ is calculated using Eq. (2), and the bit cost $R_F(n-1)$, distortion $D_F(n-1)$, and $\lambda_F(n-1)$ of the previous encoded frame are used as the down input. Then, the fusion feature of the two branches is input into the regression block. Finally, the network can predict the optimal λ.

Unlike the loss function of the two-branch residual-based network, the two-branch regression-based network for λ is trained by a multi-tasking loss function, which is defined as

$$Loss_\lambda = \gamma \left(\frac{|R_F - \hat{R}_F[\lambda]|}{R_F} \right)^2 + (1-\gamma)\hat{D}_F[\lambda] \tag{4}$$

where γ is set as 0.4 empirically; R_F is the calculated bit in Eq. (2); $\hat{R}_F[\lambda]$ and $\hat{D}_F[\lambda]$ are the actual bit and distortion, respectively; $[\lambda]$ denotes parameter λ in the range between $\hat{R}_F$ and $\hat{D}_F$.

Table 2 Experimental comparisons of Li *et al.* [5], Li *et al.* [6] and the proposed algorithm.

Class	Sequence	Li *et al.* [5] BD-rate (PSNR)	Li *et al.* [5] BD-rate (SSIM)	Li *et al.* [6] BD-rate (PSNR)	Li *et al.* [6] BD-rate (SSIM)	proposed BD-rate (PSNR)	proposed BD-rate (SSIM)
Class A1	Tango2	-3.63	-0.77	-0.20	-0.03	-0.42	-0.27
	FoodMarket4	0.83	0.10	0.08	0.02	0.74	0.07
	Campfire	-2.02	-0.65	-0.32	-0.20	-1.66	-0.42
Class A2	CatRobot1	-2.27	-0.68	-0.32	-0.03	-0.63	-0.40
	DaylightRoad2	-0.82	-0.12	-0.22	-0.06	-0.07	-0.12
	ParkRunning3	0.40	0.04	0.34	0.05	0.29	-0.05
Class B	MarketPlace	1.03	0.12	0.14	-0.02	-1.60	-0.67
	RitualDance	-0.93	-0.31	-0.82	-0.36	-1.30	-0.58
	Cactus	-0.49	-0.19	-0.40	-0.18	-1.13	-0.51
	BasketballDrive	-2.52	-0.68	-1.07	-0.67	-2.56	-0.72
	BQTerrace	-1.83	-0.60	-1.16	-0.65	-1.52	-0.60
Class C	BasketballDrill	-0.06	-0.01	-0.51	-0.32	-1.14	-0.43
	BQMall	0.15	0.00	-0.22	-0.03	-0.50	-0.34
	PartyScene	0.23	-0.02	0.18	0.02	-1.96	-0.50
	RaceHorses	-0.59	-0.08	-0.61	-0.28	-1.12	-0.46
Class D	BasketballPass	-0.89	-0.30	-1.13	-0.44	-0.05	-0.08
	BQSquare	1.03	0.13	-0.93	-0.41	-0.80	-0.30
	BlowingBubbles	1.02	0.10	1.04	0.32	-0.01	-0.06
	RaceHorses	-0.45	-0.20	-0.10	-0.01	-0.46	-0.22
Class E	FourPeople	-1.33	-0.43	-0.69	-0.23	-1.29	-0.37
	Johnny	-0.43	-0.12	-0.17	-0.10	-0.43	-0.33
	KristenAndSara	-1.59	-0.61	-0.50	-0.21	-0.82	-0.38
	Average	**-0.69**	**-0.24**	**-0.35**	**-0.17**	**-0.84**	**-0.35**

Table 1 Bit rate accuracy comparisons of DCVC, Li *et al.* [5], Li *et al.* [6] and the proposed algorithm.

Class	DCVC M%	Li *et al.*[5] M%	Li *et al.*[6] M%	proposed M%
Class A1	4.21	5.41	7.60	2.13
Class A2	4.18	5.38	7.56	2.12
Class B	1.43	3.24	6.27	3.36
Class C	1.31	2.75	4.71	2.80
Class D	2.65	3.24	3.83	1.76
Class E	1.95	3.30	5.60	1.35
Average	**2.62**	**3.89**	**5.93**	**2.25**

3. Experimental Results

The proposed algorithm is implemented in DCVC. Li *et al.* [5] and Li *et al.* [6] are used for comparison. The Vimeo-90k [7] and BVI-DVC [8] datasets are used to train the two designed networks. One hundred frames are used to encode every test sequence. DCVC is used as an anchor, and four RD points are selected: λ = 256, 512, 1024 and 2048. The bit rate accuracy is defined as

$$M = \frac{|R - \hat{R}|}{R} \tag{5}$$

where R is the target bit rate, and $\hat{R}$ is the actual bit rate. Table 1 shows the bit rate accuracy results.

Table 1 shows that the average bit rate accuracy results are 2.62%, 3.89%, 5.93% and 2.25%, respectively. The proposed algorithm has better control accuracy than the other algorithms. Since controlling the bit rate is a highly challenging task for end-to-end coding, the accuracies of the four algorithms remain high. Table 2 shows a comparison of the coding quality of the algorithms.

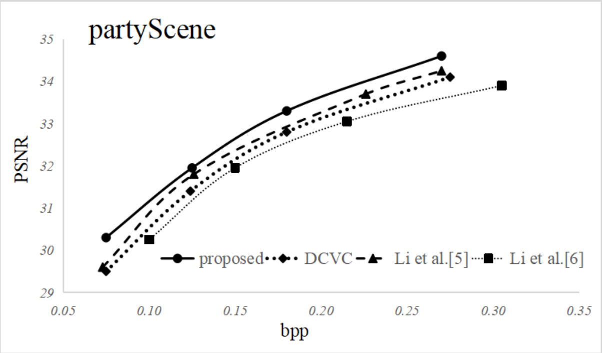

Fig. 4 RD curve comparisons of DCVC, Li *et al.* [5], Li *et al.* [6] and the proposed algorithm.

In Table 2, the average BD-rate (PSNR) indices of Li *et al.* [5], Li *et al.* [6] and the proposed algorithm are −0.69, −0.35 and −0.84, respectively. This result indicates that the proposed algorithm uses the lowest bit rate but improves the coding quality the most. For the BD-rate (SSIM) indices, the proposed algorithm achieves −0.35. Li *et al.* [5] and Li *et al.* [6] achieve values of −0.24 and −0.17, respectively. Thus, the proposed algorithm mostly improves the subjective coding quality. Since the temporal coding information is used by the proposed algorithm to train the network for coding, the bit rate can be more reasonably allocated to satisfy the changing frame feature, and λ will be more effectively selected to decrease the RD cost.

Figure 4 shows the RD comparisons of DCVC, Li *et al.* [5], Li *et al.* [6] and the proposed algorithm. The proposed algorithm has better RD performance than the other algorithms, which indicates the effectiveness of the proposed algorithm. In summary, the proposed end-to-end rate control can improve both objective and subjective coding performance with good control accuracy.

4. Conclusions

In this work, a two-branch residual-based network and a two-branch regression-based network are designed to obtain the bit rate ratio and λ for end-to-end rate control. By fully utilizing the temporal coding correlation, the rate control parameters are appropriately selected to satisfy the coding feature. Experimental results show that the proposed algorithm can significantly improve the coding performance with a high rate control accuracy.

References

[1] F. Jiang, W. Tao, S. Liu, J. Ren, X. Guo, and D. Zhao, "An end-to-end compression framework based on convolutional neural networks," IEEE Trans. Circuits Syst. Video Technol., vol.28, no.10, pp.3007–3018, 2017.

[2] G. Lu, W. Ouyang, D. Xu, X. Zhang, C. Cai, and Z. Gao, "DVC: An end-to-end deep video compression framework," Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.10998–11007, 2019.

[3] J. Li, B. Li, and Y. Lu, "Deep contextual video compression," Adv. Neural Inf. Process. Syst., vol.34, pp.18114–18125, 2021.

[4] E. Çetin, M.A. Yılmaz, and A.M. Tekalp, "Flexible-rate learned hierarchical bi-directional video compression with motion refinement and frame-level bit allocation," Proc. 2022 IEEE International Conference on Image Processing (ICIP), pp.1206–1210, 2022.

[5] Y. Li, X. Chen, J. Li, J. Wen, Y. Han, S. Liu, and X. Xu, "Rate control for learned video compression," Proc. ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp.2829–2833, 2022.

[6] B. Li, H. Li, L. Li, and J. Zhang, "λ domain rate control algorithm for high efficiency video coding," IEEE Trans. Image Process., vol.23, no.9, pp.3841–3854, 2014.

[7] T. Xue, B. Chen, J. Wu, D. Wei, and W.T. Freeman, "Video enhancement with task-oriented flow," Int. J. Comput. Vis., vol.127, pp.1106–1125, 2019.

[8] D. Ma, F. Zhang, and D.R. Bull, "BVI-DVC: A training database for deep video compression," IEEE Trans. Multimed., vol.24, pp.3847–3858, 2021.