

# A Rate-Quality Model for Learned Video Coding

Sang NguyenQuang\*, Cheng-Wei Chen\*, Xiem HoangVan† and Wen-Hsiao Peng\*

\* National Yang Ming Chiao Tung University, Taiwan

† VNU University of Engineering and Technology, Vietnam

E-mail: xiemhoang@vnu.edu.vn

**Abstract**—Learned video coding (LVC) has recently achieved superior coding performance. However, there is a lack of research on the rate-quality (R-Q) model which is important in real-time LVC applications. In this paper, we propose a parametric model to characterize the R-Q relationship in LVC systems. In the proposed model, a neural network, termed RQNet is introduced to characterize the relationship between bitrate and quality level according to video content and coding context. The predicted (R,Q) results are further integrated with those from previously coded frames using the least-squares method to determine the parameters of our R-Q model on-the-fly. Compared to the conventional approaches, our method accurately estimates the R-Q relationship, enabling the online adaptation of model parameters to enhance both flexibility and precision. Experimental results show that our R-Q model achieves significantly smaller bitrate deviations than the baseline methods on commonly used datasets with minimal additional complexity.

**Index Terms**—Learned video coding, Rate-Quality characteristic, Rate Control

## I. INTRODUCTION

Learned video coding has been considered as a promising coding solution for a wide range of video transmission and storage. Similar to traditional video coding standards, rate control plays a crucial role in deploying LVC for practical video applications. An effective rate control system must deliver high rate-distortion performance while minimizing deviation between actual and target bitrates. The encoder's efficiency depends on bitrate allocation across frames, while bitrate deviation is determined by encoding parameters. To achieve this, a well-designed model that accurately captures the relationship between bitrate and coding parameters (i.e., quantization parameter or quality level) is indispensable.

In learned video coding [1]–[4], accurately estimating the rate-quality (R-Q) relationship poses a significant challenge. Unlike traditional codecs [5]–[7] with well-calibrated analytical models, learned video codecs exhibit distinct R-Q characteristics heavily influenced by model architectures, training protocols, and optimization strategies.

In recent years, capturing R-Q relationships in image and video coding has emerged as an important research topic. Jia *et al.* [8] utilize exponential and logarithmic functions to characterize the relationship among rate (R), distortion (D), and the Lagrange multiplier ( $\lambda$ ), which indicates the slope of the R-D curve. Xue *et al.* [9] propose an exponential function with an additional bias term to model the R- $\lambda$  relationship and enhance the rate-fitting accuracy. Similar approaches have been applied to video coding [10], where the R- $\lambda$  and D- $\lambda$  curves are fitted using exponential models, and  $\lambda$  values are

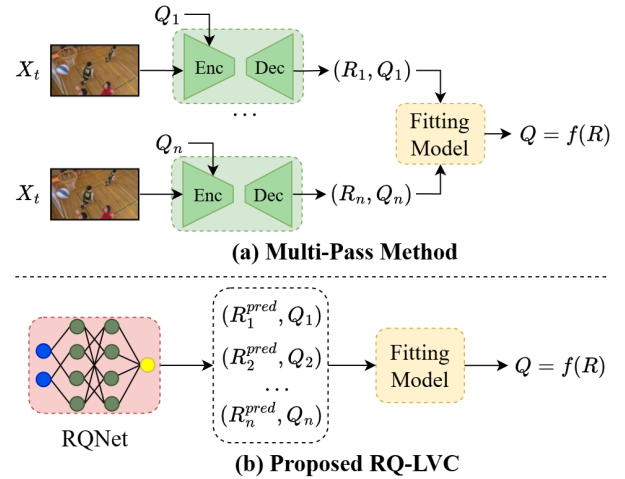


Fig. 1: Comparison of the conventional method and our RQ-LVC in predicting rate-quality relationship.

updated dynamically based on encoding results of previous frames. Inspired by [10], Li *et al.* [11] propose a learned R-D- $\lambda$  model and parameter updating mechanism for learned video coding. Jia *et al.* [12] explore the relationship between bitrate and quantization parameter, while Chen *et al.* [13] introduce the scaling factor as a hyperparameter in the R-D model for learned video codecs. Additionally, Gu *et al.* [14] design two neural network models to predict (R, $\lambda$ ) and (D, $\lambda$ ) points, leveraging them to model the relationship between R and D. In contrast, Zhang *et al.* [15] propose a fully neural network-based rate control system including a rate allocation model and rate implementation network to perform the rate-parameter mapping. Although these methods have proven effective, they remain fundamentally constrained by predefined  $\lambda$  values, limiting their adaptability for rate control in learned video coding. Moreover, most methods either update model parameters iteratively by adjusting parameters based on rate-distortion statistics collected from previously encoded frames or use neural networks to predict coding parameters. These approaches operate separately and do not take full advantage of available information.

In this paper, we introduce a rate-quality (R-Q) model for learned video coding to estimate the R-Q relationship. In our approach, termed RQ-LVC, the quality level acts as a coding parameter to regulate the variable-rate video coding process. Compared with conventional approaches that independently model the R- $\lambda$  or D- $\lambda$  relationships, our method directly

TABLE I: Our method vs. prior works

Method	Rate Control Model	Updating Mechanism	Application
TIP'14 [10]	$\lambda = \alpha R^\beta$	Iterative Mode	Traditional Codec
ICASSP'24 [12]	$R = C \times Q^{-K}$	Iterative Mode	Learned Codec
ICLR'24 [15]	Neural Network	Neural Network	Learned Codec
TCSVT'24 [13]	$R = \alpha \times r^\beta$	Iterative Mode	Learned Codec
DCC'25 [14]	$R = \alpha \times \lambda^\beta$	Neural Network	Learned Codec
Our RQ-LVC	$Q = \alpha \times \ln(R) + \beta$	Batch Mode	Learned Codec

\*R: bitrate; Q: quality level;  $\lambda$ : Lagrange multiplier;  $\alpha, \beta, C, K$ : model parameter;  $r$ : rescale ratio.

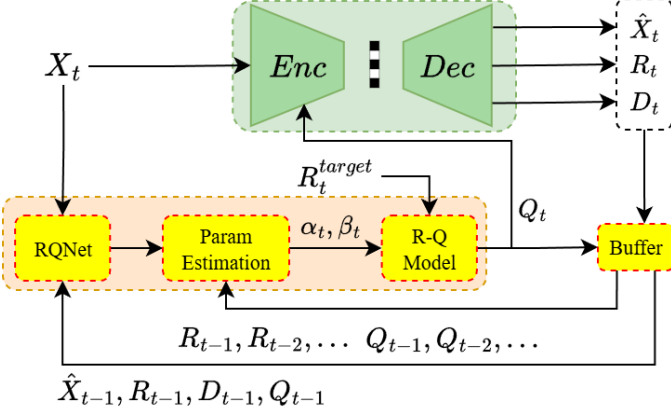


Fig. 2: System overview of our RQ-LVC framework with the RQNet, parameter estimation, and R-Q model. RQNet extracts spatiotemporal information to predict (R,Q) points, which are combined with those from previously coded frames to estimate the parametric R-Q model. Then, the predicted  $Q_t$  is estimated based on the target bitrate  $R_t^{target}$  to regularize the encoding of an input video frame.

models the link between bitrate and quality, enabling precise rate control for learning-based video codecs. Furthermore, our proposed framework dynamically updates the R-Q model on-the-fly in a batch manner, allowing online adaptation by using information from previously coded frames and predictions from a deep neural network.

Our contributions in this work are best illustrated in Fig. 1. As depicted, multi-pass approaches collect (R,Q) points by encoding a video frame multiple times with different quality levels to fit the R-Q relationship, which is highly time-consuming. In contrast, our proposed RQ-LVC efficiently models the R-Q relationship with low complexity. By leveraging RQNet, our approach significantly reduces encoding complexity while ensuring accurate rate control, offering a more computationally efficient and scalable solution. In Table I, we compare the proposed method with several prior works in terms of the rate control model, model parameters updating mechanism, and application scenario.

## II. PROPOSED METHOD

Fig. 2 illustrates the overall architecture of the proposed rate-quality model for learned video coding (RQ-LVC). RQ-LVC encodes each input frame  $X_t$  at variable quality level  $Q_t$

under a rate constraint  $R_t^{target}$  by modeling the rate-quality (R-Q) relationship. Specifically, our proposed plug-in module, RQNet, predicts (R,Q) points based on the coding context—such as rate and distortion of reference frames—and content of  $X_t$ . These predictions serve as prior knowledge, which is further refined using the observed (R,Q) points collected from previously coded frames to update the R-Q model. The updated R-Q model determines the appropriate quality level  $Q_t$  to satisfy  $R_t^{target}$ .

### A. Learning a Variable-Rate Neural Video Codec

Our framework features a variable-rate neural video codec that adapts bitrate and quality based on quality level  $Q$ . Instead of training separate models for different rates, we develop a variable-rate version using MaskCRT [2] as the base codec.

To train this variable-rate version of MaskCRT [2], we adopt the standard training loss function for learned video coding, namely,  $Loss_{RD} = R + \lambda D$ , where  $R$ ,  $D$ , and  $\lambda$  represent the bitrate, distortion, and Lagrange multiplier, respectively. To support wide bitrate ranges with a single model,  $\lambda$  is randomly sampled from a predefined range during each training iteration. Following [4], we relate  $\lambda$  to a (continuous) quality parameter  $Q$  within the interval  $[0, Q_{num} - 1]$ :

$$\lambda = \exp \left\{ \ln \lambda_{min} + \frac{Q}{Q_{num} - 1} (\ln \lambda_{max} - \ln \lambda_{min}) \right\}, \quad (1)$$

where  $\lambda_{min}, \lambda_{max}$  define the lowest and highest Lagrange multipliers, respectively. To sample a  $\lambda$  between  $\lambda_{min}$  and  $\lambda_{max}$ , we simply choose a  $Q$  from  $[0, Q_{num} - 1]$ .  $Q_{num} = 64$  in our current implementation.

### B. Parametric Rate-Quality (R-Q) Models

We adopt a parametric approach to construct an R-Q model, enabling the prediction of  $Q$  for a given target bitrate  $R^{target}$ . To analyze the relationship between the bitrate  $R$  and quality level  $Q$ , we collect frame-level (R,Q) statistics from the UVG [16] and HEVC Class B [17] datasets. This is achieved by encoding some sequences from these datasets using various quality levels  $Q$  and collecting the resulting bitrates  $R$ . Three parametric models—linear, exponential, and logarithmic models—are employed to fit these frame-level (R,Q) data points:

$$Q = \alpha \times R + \beta, \quad (2)$$

$$Q = \alpha \times e^\beta, \quad (3)$$

$$Q = \alpha \times \ln(R) + \beta, \quad (4)$$

where  $\alpha, \beta$  are obtained by the least-squares method. In this work, the model achieving the highest  $R^2$  score is deemed the best among the three. As shown in Table II, the logarithmic model consistently outperforms the linear and exponential models. Furthermore, Fig. 3 confirms that the logarithmic model effectively captures the relationship between bitrate ( $R$ ) and quality level ( $Q$ ). Thus, we choose the logarithmic model as our parametric solution for modeling the R-Q relationship.

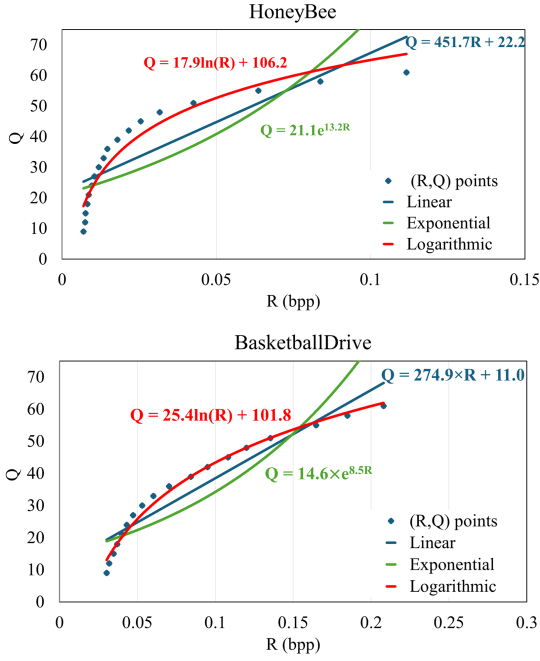


Fig. 3: R-Q models for HoneyBee and BasketballDrive sequences.

TABLE II:  $R^2$  scores for various parametric models.

Sequence	Fitting Model		
	Linear	Exponential	Logarithmic
Kimono1	0.943	0.806	0.997
BasketballDrive	0.898	0.767	0.988
ReadySteadyGo	0.925	0.781	0.995
Jockey	0.801	0.680	0.964
HoneyBee	0.684	0.547	0.929
Beauty	0.810	0.713	0.948
Average	0.844	0.716	0.970

### C. Online Estimation of Model Parameters

The coefficients  $\alpha$  and  $\beta$  of the R-Q model are highly dependent on both video content and the backbone codec (e.g., MaskCRT). A straightforward approach to estimate their values involves encoding an input video multiple times using various quality levels and fitting  $\alpha, \beta$  to the resulting frame-level (R,Q) data points. However, this approach is computationally expensive and impractical. Drawing inspiration from Bayesian inference [18], we learn a neural network, termed RQNet, to predict a number of (R,Q) points based on video content and contextual information. Because RQNet is trained offline on a large dataset, these network-predicted (R,Q) points provide the prior knowledge of the R-Q relationship for the current coding context. To address the amortized inference issue (i.e., the RQNet predictions are not optimal for individual videos), we further refine this prior knowledge by incorporating the (R,Q) points collected from previously coded frames. These (R,Q) points are then combined in a least-squares framework to dynamically update  $\alpha$  and  $\beta$  at inference time.

1) *RQNet*: Fig. 4 depicts the U-Net architecture of RQNet. The inputs consist of the current coding frame  $X_t$ , frame difference  $X_t - \hat{X}_{t-1}$  between  $X_t$  and the previously decoded

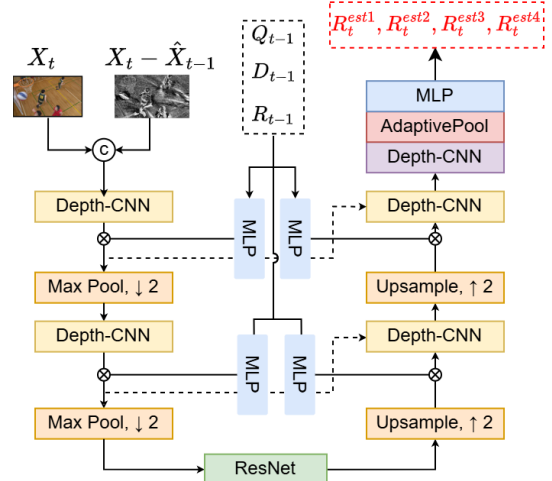


Fig. 4: The architecture of the proposed RQNet. Depth-wise separable convolution (Depth-CNN) [19] and MLP layers are used to extract spatiotemporal information.

frame  $\hat{X}_{t-1}$ , as well as the rate-distortion statistics  $R_{t-1}$ ,  $D_{t-1}$ ,  $Q_{t-1}$  from the previous frame. The model outputs four bitrates corresponding to the predefined quality levels of 10, 17, 43 and 60, respectively. Instead of predicting the bitrate for every possible quality level, we simplify the task to focus on only four bitrates for these fixed quality levels. Consequently, RQNet provides four (R,Q) points that are both content- and context-dependent.

To train RQNet, video frames are encoded at predefined quality levels to collect the actual bitrates, denoted as  $R_t^{enc}$ . The model then predicts the estimated bitrates  $R_t^{pred}$  for the four quality levels. The training objective is to minimize the mean absolute deviation between  $R_t$  and  $R_t^{pred}$ :

$$L = \frac{1}{4} \sum_{n=1}^4 \left| R_t^{enc(n)} - R_t^{pred(n)} \right|. \quad (5)$$

2) *Least-Squares for Parameter Estimation*: There are two sets of (R,Q) points. One is derived from the output of RQNet and the other is collected from the actual encoding of previous frames. The (R,Q) points from RQNet serve as prior knowledge while those sampled from coded frames are additional observations.

Consider the data points  $\{R_t^{pred(n)}, Q_t^{pred(n)}\}_{n=1}^4$  from RQNet and  $\{R_j^{enc}, Q_j^{enc}\}_{j < t}$  collected from the coded frames up to the time index  $t$  within a group-of-pictures (GOP);  $\alpha_t$  and  $\beta_t$  are estimated by minimizing the sum of squared errors:

$$\arg \min_{\alpha_t, \beta_t} \sum_{n=1}^4 \left( Q_t^{pred(n)} - f \left( R_t^{pred(n)}; \alpha_t, \beta_t \right) \right)^2 + \sum_{j=1}^{t-1} \left( Q_j^{enc} - f \left( R_j^{enc}; \alpha_t, \beta_t \right) \right)^2, \quad (6)$$

where  $f(\cdot)$  represents the logarithmic R-Q function in Eq. (4).

### III. EXPERIMENTAL RESULTS

#### A. Experimental Settings

**Datasets:** We use Vimeo-90K [20] to train RQNet and the variable-rate MaskCRT. We randomly crop  $256 \times 256$  patches for training. The test datasets are HEVC Class B [17] and UVG [16].

**Implementations:** We evaluate our proposed method with three variants: RQ-LVC, RQ-LVC w/ RQNet Only, and RQ-LVC w/o RQNet. Our RQ-LVC combines (R,Q) points collected from both RQNet and previously coded frames to estimate the parameters of the R-Q model. In contrast, RQ-LVC w/ RQNet relies solely on RQNet predictions, while RQ-LVC w/o RQNet uses only previously coded frames without utilizing any predictions from RQNet.

**Evaluation Methodologies:** In our evaluation, we incorporate proposed RQ-LVC into a rate allocation algorithm, as in [10], [12]. First, we determine the average target bitrates per frame  $R_s$  using MaskCRT [2] with a fixed quality level  $Q \in \{10, 25, 40, 55\}$ . Based on the given  $R_s$ , we compute the target bitrate  $R_{mg}$  for a miniGOP, which includes a set of  $N_m$  consecutive frames  $\{X_i, X_{i+1}, \dots, X_t, \dots, X_{i+N_m-1}\}$ :

$$R_{mg} = \frac{R_s \times (N_{coded} + SW) - \hat{R}_s}{SW} \times N_m, \quad (7)$$

where  $N_{coded}$  is the number of encoded frames,  $\hat{R}_s$  is the total bitrate already consumed, and  $SW$  refers to the sliding window size, which is set to 40 in our implementation. Afterward, the target bitrate  $R_t$  (in bits) for frame  $X_t$  is given by:

$$R_t = \frac{R_{mg} - \hat{R}_{mg}}{\sum_{j=t}^{i+N_m-1} w_j} \times w_t, \quad (8)$$

where  $\hat{R}_{mg}$  is the bitrate consumed by previously coded frames within the current miniGOP, and  $w_t$  denotes the rate allocation weight for frame  $X_t$ . In this work, we set the number  $N_m$  of frames in a miniGOP to 4 with empirically chosen weights  $\{1.9, 1.6, 1.3, 1.0\}$ .

The experiments are conducted by encoding the first 96 frames of each video sequence. We pre-encode the video sequences using a single quality level across all frames, and then use the resulting bitrate as the target bitrate for RQ-LVC. We measure  $\Delta R^{RC}$  to assess RQ-LVC rate control accuracy:

$$\Delta R^{RC} = \left| \frac{R^{target} - R^{enc}}{R^{target}} \right| \times 100\%, \quad (9)$$

where  $R^{target}$  and  $R^{enc}$  denote the target bitrate and the actual bitrate resulting from encoding the video sequence with our RQ-LVC, respectively. We evaluate with 4 quality levels in  $\{10, 25, 40, 55\}$  and report the average per-sequence rate deviation over these quality levels.

The accuracy  $P^{RQNet}$  of RQNet in predicting the bitrates at predefined quality levels is defined as the average absolute difference between the predicted bitrate  $R_t^{pred}$  and the actual bitrate  $R_t^{enc}$  across  $n$  frames:

$$P^{RQNet} = \frac{\sum_{t=1}^n \frac{|R_t^{enc} - R_t^{pred}|}{R_t^{pred}}}{n} \times 100\%, \quad (10)$$

To evaluate the rate-distortion performance, we calculate BD-rate savings [21], with the anchor adopting constant quality levels without rate control.

**Baseline Methods:** Our method is compared with the multi-pass encoding approach, which pre-encodes frames with four quality levels  $\{10, 17, 43, 60\}$  and estimates model parameters using the least-squares algorithm. We adopt Li *et al.*'s update strategy [10] as our baseline. Following [10], we use the same initial  $\alpha_{t=1}$  and  $\beta_{t=1}$  for each test sequence, obtained by averaging their respective values over the first frames of all sequences in both datasets. Based on Eq. (11), the quality level  $Q_t^{real}$  is derived from the target bitrate  $R_t^{target}$  and used to encode the current coding frame  $X_t$  with MaskCRT, resulting in a bitrate of  $R_t^{real}$ :

$$Q_t^{real} = \alpha_t \times \ln(R_t^{target}) + \beta_t. \quad (11)$$

With  $R_t^{real}$ , the corresponding quality level  $Q_t^{est}$  is estimated as:

$$Q_t^{est} = \alpha_t \times \ln(R_t^{real}) + \beta_t. \quad (12)$$

By adopting the Adaptive Least Mean Square method, the model parameters  $\alpha, \beta$  are updated iteratively by Eq. (13) and Eq. (14), respectively:

$$\alpha_{t+1} = \alpha_t + \mu \times (Q_t^{real} - Q_t^{est}) \times \ln(R_t^{real}), \quad (13)$$

$$\beta_{t+1} = \beta_t + \eta \times (Q_t^{real} - Q_t^{est}), \quad (14)$$

where  $\mu$  and  $\eta$  are the learning rates, which are set to 0.01 in our experiments. Besides, we adopt the R-Q model and updating mechanism proposed by Liao *et al.* [12] as a competing method.

#### B. Experimental Results

**Rate Control Performance of RQ-LVC:** Table III evaluates the rate control performance of our RQ-LVC framework. As shown in the table, RQ-LVC consistently achieves significantly smaller bitrate deviations than the baseline methods [10], [12]. This improvement is attributed to the fact that RQ-LVC leverages the (R,Q) points output by RQNet, along with those from previously coded frames, to estimate model parameters  $\alpha, \beta$ . In contrast, the approaches presented by Li *et al.* [10] and Liao *et al.* [12] rely exclusively on the (R,Q) points from the past frames. Fig. 5 visualizes the R-Q functions with the predicted  $\alpha, \beta$  for frame 16 across various video sequences, alongside the actual (R,Q) points obtained by encoding the video with various quality levels. The R-Q function predicted by our RQ-LVC closely aligns with the actual (R,Q) points, whereas the R-Q functions predicted by Li *et al.* [10] exhibit significant deviations from these ground truths. In some sequences, such as Bosphorus, HoneyBee, Jockey, and ReadySteadyGo, our proposed RQ-LVC even outperforms the time-consuming multi-pass encoding strategy. In addition, the application of hierarchical quality-level patterns allows efficient compression. Our RQ-LVC ensures that actual bitrates closely match the allocated target bitrates for every frame, thereby achieving consistently high compression performance across all test sequences. Besides, the additional overhead is

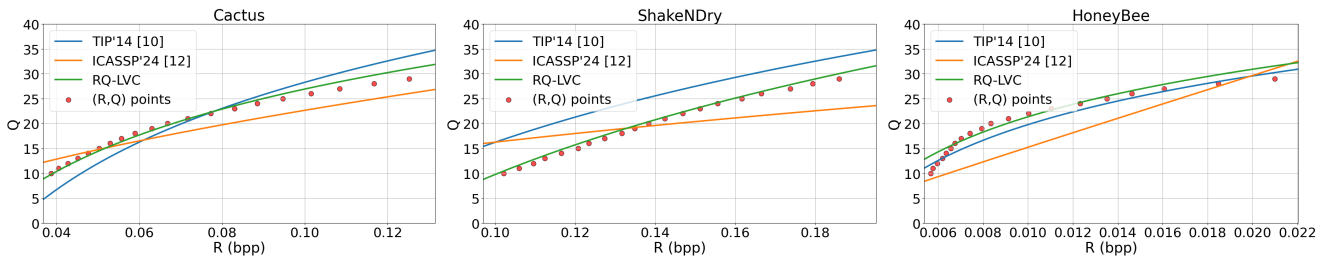


Fig. 5: The predicted R-Q functions for frame 16 versus the ground-truth (R,Q) points collected by multi-pass encoding.

TABLE III: Comparison of the competing rate control schemes in terms of bitrate deviation  $\Delta R^{RC}$  and BD-rate saving (%). Notation:  $\Delta R^{RC}$  / BD-rate saving (%).

Dataset	Sequence	Four-pass	One-pass				
			TIP'14 [10]	ICASSP'24 [12]	RQ-LVC w/ RQNet Only	RQ-LVC w/o RQNet	RQ-LVC
UVG	Beauty	1.27 / 7.2	2.15 / 0.4	3.47 / 2.5	6.47 / 7.2	1.70 / 1.2	2.07 / -0.1
	Bosphorus	2.42 / -5.8	4.38 / -2.9	1.41 / -5.9	7.50 / -5.8	0.82 / -2.7	0.46 / -6.1
	HoneyBee	4.28 / -7.8	6.63 / -3.2	34.09 / -3.3	2.88 / -4.8	13.25 / -20.5	2.23 / -7.7
	Jockey	1.69 / -2.9	4.99 / -0.4	1.71 / 0.0	4.18 / -2.0	0.74 / -3.7	1.58 / -3.5
	ReadySteadyGo	1.80 / -4.3	2.96 / -2.3	1.97 / -4.5	4.47 / -2.8	1.15 / -3.0	0.84 / -4.8
	ShakeNDry	0.19 / -2.6	5.97 / -1.9	2.65 / -0.8	2.21 / -2.1	1.31 / -3.4	0.88 / -2.6
	YachtRide	0.42 / -2.3	3.21 / -2.2	2.27 / -0.8	1.09 / -2.0	1.38 / -2.2	1.04 / -2.3
	Average	1.72 / -2.6	4.33 / -1.8	6.80 / -1.8	4.11 / -1.8	2.90 / -4.9	1.30 / -3.9
HEVC-B	BasketballDrive	1.00 / -2.3	2.24 / -2.2	3.65 / -2.1	1.77 / -2.7	2.21 / -2.2	1.63 / -2.9
	BQTerrace	1.64 / -2.1	11.69 / 7.3	1.96 / -1.1	1.10 / -1.6	3.58 / 0.2	0.88 / -1.8
	Cactus	1.42 / -4.0	1.08 / -5.8	2.72 / -3.7	2.27 / -3.0	1.88 / -3.7	0.57 / -4.1
	Kimono1	0.20 / 0.5	2.12 / -0.3	2.52 / 1.1	2.60 / -0.4	0.81 / -0.3	0.40 / -0.4
	ParkScene	1.89 / -5.0	0.96 / -8.1	2.53 / -4.6	2.04 / -3.2	1.45 / -4.7	0.59 / -4.8
	Average	1.23 / -2.6	3.62 / -1.8	2.67 / -2.1	1.96 / -2.2	1.99 / -2.1	0.81 / -2.8

TABLE IV: Prediction accuracy  $PRQNet$  of RQNet in estimating bitrates at predefined quality levels.

Dataset	Sequence	Q=10	Q=27	Q=43	Q=60	Average
UVG	Beauty	7.01	4.50	24.01	24.12	14.91
	Bosphorus	41.21	28.77	27.54	23.29	30.20
	HoneyBee	11.63	17.44	44.59	38.90	28.14
	Jockey	20.75	12.00	12.93	17.20	15.72
	ReadySteadyGo	11.87	13.70	20.66	24.94	17.79
	ShakeNDry	12.99	12.72	15.82	15.45	14.25
	YachtRide	7.40	10.86	8.27	12.30	9.71
	Average	16.12	14.28	21.98	22.31	16.87
HEVC-B	Kimono1	5.99	7.94	6.26	19.17	9.84
	BQTerrace	23.11	12.77	23.82	25.29	21.25
	Cactus	18.30	12.90	20.34	21.12	18.16
	BasketballDrive	8.50	6.82	13.90	5.57	8.70
	ParkScene	21.04	17.03	22.64	24.84	21.39
	Average	15.39	11.49	17.39	19.20	15.87

minimal compared to MaskCRT, the base codec. Specifically, the additional complexity introduced by RQNet amounts to an 8% increase in encoding multiply-accumulate operations per pixel (kMAC/pixel) and a 14% increase in encoding time, while the RQNet model size represents only a fraction of MaskCRT's.

**Model parameter estimation with RQNet Only:** We evaluate the rate control performance of RQNet by estimating model parameters solely from RQNet-predicted (R,Q) points, in a way similar to [14], which uses neural networks to predict (R, $\lambda$ ) and (D, $\lambda$ ) data points to estimate model parameters for rate control. From Table III, it is evident that relying exclusively on the (R,Q) points from RQNet is ineffective. To further analyze the performance of RQNet, Table IV presents RQNet's prediction accuracy. RQNet achieves average bitrate deviations of 16.87% and 15.87% on UVG and HEVC Class B datasets, respectively. The deviations are relatively larger at

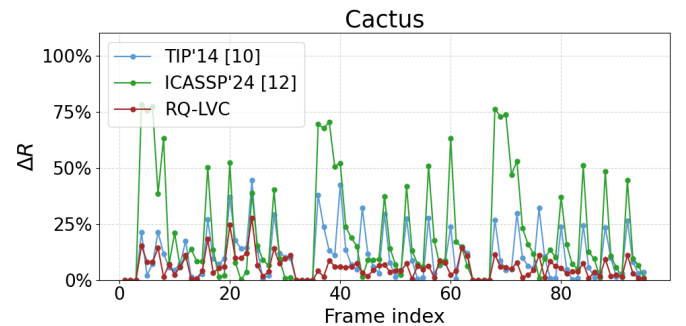


Fig. 6: Comparison of per-frame rate deviation  $\Delta R$  between our RQ-LVC and competing methods for Cactus sequence.

higher bitrates, indicating that the predicted (R,Q) points are less reliable at higher rates, making it beneficial to also use the (R,Q) points from previously coded frames.

**Model parameter estimation without RQNet:** The results in Table III also demonstrate that model parameter estimation using only previously coded (R,Q) points can still yield effective results. However, this approach has two significant limitations: (1) it requires a substantial number of historical (R,Q) points, and (2) it struggles when the target bitrate range maps to an extremely narrow QP range, as observed in the HoneyBee sequence results.

**Per-frame rate deviation:** Maintaining precise rate control is critical, as deviations impact both current and subsequent frame quality. Exceeding the target bitrate constrains the bitrate budget for remaining frames, while under-encoding compromises the quality of the current frame. As demonstrated in Fig. 6, RQ-LVC maintains this accuracy even for the first

initial frames within a GOP, where historical coding data is limited. This is achieved by utilizing predictions from RQNet. In contrast, conventional methods [10] [12] frequently struggle in such scenarios.

#### IV. CONCLUSION

This paper proposes a rate-quality model for learned video coding and an adaptive strategy for on-the-fly parameter updates. We introduce RQNet, a lightweight neural network that predicts encoding bits at predefined quality levels, removing the need for multiple encoding passes. By combining RQNet's predictions with the encoding results from previously coded frames, we explore the least-squares method to estimate the model parameters for each frame. Experimental results show our method achieves significantly lower bitrate deviations than the baseline method, with minimal extra computational overhead. For future work, we will improve RQNet's effectiveness, weight (R,Q) data point contributions to improve parameter estimation, and developing neural approaches for bitrate allocation to improve compression performance.

#### REFERENCES

- [1] Y.-H. Ho, C.-P. Chang, P.-Y. Chen, A. Gnutti, and W.-H. Peng, "CANF-VC: Conditional Augmented Normalizing Flows for Video Compression," in *2022 European Conference on Computer Vision (ECCV)*, 2022, pp. 207–223.
- [2] Y.-H. Chen, H.-S. Xie, C.-W. Chen, *et al.*, "MaskCRT: Masked Conditional Residual Transformer for Learned Video Compression," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 11, pp. 11 980–11 992, 2024.
- [3] J. Li, B. Li, and Y. Lu, "Neural Video Compression with Diverse Contexts," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 22 616–22 626.
- [4] J. Li, B. Li, and Y. Lu, "Neural Video Compression with Feature Modulation," in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 26 099–26 108.
- [5] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC Video Coding Standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, 2003.
- [6] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. 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.
- [7] B. Bross, Y.-K. Wang, Y. Ye, *et al.*, "Overview of the Versatile Video Coding (VVC) Standard and Its Applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 10, pp. 3736–3764, 2021.
- [8] C. Jia, Z. Ge, S. Wang, S. Ma, and W. Gao, "Rate Distortion Characteristic Modeling for Neural Image Compression," in *2022 Data Compression Conference (DCC)*, 2022, pp. 202–211.
- [9] N. Xue and Y. Zhang, "Lambda-Domain Rate Control for Neural Image Compression," *Proceedings of the 5th ACM International Conference on Multimedia in Asia*, 2023.
- [10] B. Li, H. Li, L. Li, and J. Zhang, " $\lambda$  Domain Rate Control Algorithm for High Efficiency Video Coding," *IEEE Transactions on Image Processing*, vol. 23, pp. 3841–3854, 2014.
- [11] Y. Li, X. Chen, J. Li, *et al.*, "Rate Control for Learned Video Compression," in *2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 2829–2833.
- [12] S. Liao, C. Jia, H. Fan, J. Yan, and S. Ma, "Rate-Quality Based Rate Control Model for Neural Video Compression," in *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024, pp. 4215–4219.
- [13] J. Chen, M. Wang, P. Zhang, S. Wang, and S. Wang, "Sparse-to-Dense: High Efficiency Rate Control for End-to-End Scale-Adaptive Video Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 5, 2024.
- [14] B. Gu, H. Chen, M. Lu, J. Yao, and Z. Ma, "Adaptive Rate Control for Deep Video Compression with Rate-Distortion Prediction," in *2025 Data Compression Conference (DCC)*, 2025, pp. 33–42.
- [15] Y. Zhang, G. Lu, Y. Chen, *et al.*, "Neural Rate Control for Learned Video Compression," in *The Twelfth International Conference on Learning Representations*, 2023.
- [16] A. Mercat, M. Viitanen, and J. Vanne, "UVG Dataset: 50/120fps 4K Sequences for Video Codec Analysis and Development," in *Proceedings of the 11th ACM Multimedia Systems Conference*, 2020, pp. 297–302.
- [17] F. Bossen *et al.*, "Common test conditions and software reference configurations," *JCTVC-L1100*, 2013.
- [18] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin, *Bayesian Data Analysis*, 3rd. Chapman and Hall/CRC, 2013. DOI: 10.1201/b16018.
- [19] F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1800–1807.
- [20] T. Xue, B. Chen, J. Wu, D. Wei, and W. T. Freeman, "Video Enhancement with Task-Oriented Flow," *International Journal of Computer Vision*, vol. 127, no. 8, pp. 1106–1125, 2019.
- [21] "Working Practices Using Objective Metrics for Evaluation of Video Coding Efficiency Experiments," *Standard ISO/IEC TR 23002-8, ISO/IEC JTC 1*, Jul, 2020.