A MACHINE LEARNING APPROACH TO ACCURATE SEQUENCE-LEVEL RATE CONTROL SCHEME FOR VIDEO CODING

Yangfan Sun, Mouqing Jin, Li Li, and Zhu Li

{ysb5b, mqjwb8}@mail.umkc.edu, {lil1, lizhu}@umkc.edu

University of Missouri, Kansas City

ABSTRACT

In this paper, we propose a two-pass encoding framework to handle the problem of sequence-level rate control. We consider the sequence-level encoding parameter constant rate factor (CRF) as the factor to be adjusted. The proposed framework mainly has two key contributions. First, we provide a second order model to characterize the relationship between the bitrate and CRF. The proposed second order model outperforms the traditional linear model significantly. Second, we adopt a shallow neural network to train the relationship between the content-dependent features with the second-order model parameters. The proposed neural network is quite simple but able to estimate the model parameters accurately. We implement the proposed algorithm under tensorflow. Experimental results show that our proposed method obviously outperforms the state-of-the-art method.

Index Terms— Machine learning, Rate control, constant rate factor, sequence-level, second order model, video coding

1. INTRODUCTION

Rate control is quite an effective coding tool to decrease the expense of encoding. It is capable of adjusting parameters to match the target bitrate to avoid multi-pass coding. Since we have a common sense that the quantization parameters (QP) have significant influences on the residue bitrate and distortion, the early researches always build a R-Q model between the bitrate R and QP or QStep [1] [2]. Since the R-Q model may be with quite complex forms, He et al. [3] later propose a rho-domain based rate control algorithm and build a linear relationship between bitrate and the percentage of non-zero coefficients after transform and quantization. However, as mentioned in [5], both models can only characterize the residue bitrate. Along with the fast development of coding standards especially the newest video coding standard High Efficiency Video Coding (HEVC) [4], the non-residue bits can no longer be ignored. Recently, Li et al. [5] propose that the lambda is the key factor to determine the bitrate and develop a R-lambda model based rate control algorithm.

All the previous researches can adjust encoding parameters from block to block or frame to frame so that the bitrate can be controlled in a very precise way. There are also some use cases where we need to deal with the sequence-level rate control problem, in which we can only adjust one sequence-level parameter, for example, the constant rate factor (CRF) [6], to achieve the target bitrate. A straightforward method of rate control on CRF is to apply multi-pass encoding scheme to approximate the target bitrate as much as possible. However, multi-pass encoding may lead to much more computational resources and encoding time. Therefore, our objective is to find a robust model to characterize the content-related relationship between the target bitrate and CRF.

One recent work [7] proposes a linear model between the bitrate and CRF to solve this problem. A shallow neural network is also provided to estimate the content-related linear model parameters and CRF. However, according to our analysis, the linear model is quite inaccurate to describe the relationship between the bitrate and CRF. Besides, as [7] tries to combine all the resolutions together, it makes the model even more inaccurate. As a result, the achieved target bitrate is far from satisfiable.

In this paper, we follow the approaches in [7] and propose a resolution-free two-pass sequence-level rate control scheme to significantly improve the bitrate accuracy. Our proposed framework mainly has two key contributions.

- First, for each spatial resolution, we build a second order model between the bitrate and CRF. Compared with the linear model, the proposed second order model can increase the model accuracy significantly partially due to the second order model, and partially thanks that we build an independent model for each resolution.
- Second, we establish a simple network structure to estimate the content-related second order model parameters accurately. Compare with [7], our network structure is not only with simpler structure but also with better accuracy.

This paper is organized as follows. In Section 2, we will introduce the overview of our proposed framework. Then in Section 3 and 4, the proposed second order model between bitrate and CRF, and the machine-learning based

training process will be presented, respectively. After that, many experiments of each resolution will be performed to show the benefits brought by the proposed framework in Section 5. At last, we will conclude the whole paper in Section 6.

2. THE PROPOSED FRAMEWORK

In this section, we will introduce the overview of our proposed two-pass coding framework, as shown in Fig, 1. We intend to achieve the target bitrate that uses only two pass coding to derive the expected CRF for encoding. From Fig. 1, we can see that there are mainly three steps including the offline training process, first-pass encoding process, and second-pass encoding process. We obtain the network from offline training process and feed it in the first-pass coding to generate the content-dependent model parameters. The model parameters will then be used to calculate the CRF according to the target bitrate.

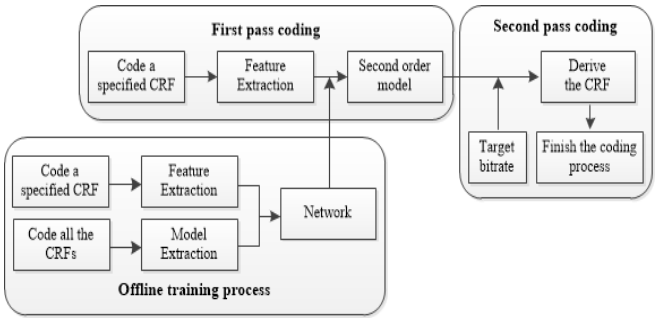

Fig. 1. The framework of the proposed two-pass scheme

3. THE PROPOSED SECOND ORDER MODEL

In this section, we will deduce the second order content-dependent model between the bitrate and CRF from the linear model.

As shown in [3], there is a quite robust relationship between the bitrate R and the Lagrange Multiplier lambda,

$$\lambda = \alpha(v) R^{\beta(v)} \tag{1}$$

where alpha and beta are the model parameters related to the video coding. In the meantime, there is also quite a good relationship between QP and lambda as follows [8].

$$QP = c \ln \lambda + d \tag{2}$$

where c and d are constant numbers. If we combine (1) and (2), we can obtain the following relationship between QP and ln(R),

$$QP = a(v) \ln R + b(v) \tag{3}$$

Where $a(v)$ and $b(v)$ are content-related model parameters.

Besides the relationship shown in (3), the relationship between the QP and ln(R) in [7] is also a linear relationship, although it also takes the resolution into consideration in the model. Therefore, we first try to apply the linear model as shown in (3). We apply the linear model on the test sequences with different resolutions. We first many pairs of CRFs and bitrates from 10 to 42 and then use the linear model to fit the data. The average bitrate errors on 720 sequences with various contents and resolutions can be seen from Table 1. The bitrate error in Table 1 is calculated using the following equation.

$$error = \frac{|R_A - R_t|}{R_t} \times 100\% \tag{3}$$

Table 1 The average bitrate error using the linear model

	error within 20%	**error within 10%**
270p	98%	87%
480p	96%	80%
720p	89%	62%
1080p	82%	49%
Avg.	**91%**	**69%**

From Table 1, we can see that the linear model can only achieve 91% and 69% of total samples within 20% and 10% bitrate errors in average. A typical failure case of the linear model can be seen from Fig. 2. From Fig. 2, we can see that the fitted curve of our linear model does not match the actual dotted line. It seems the linear model is not good enough to make the model adaptable to a large range of CRFs.

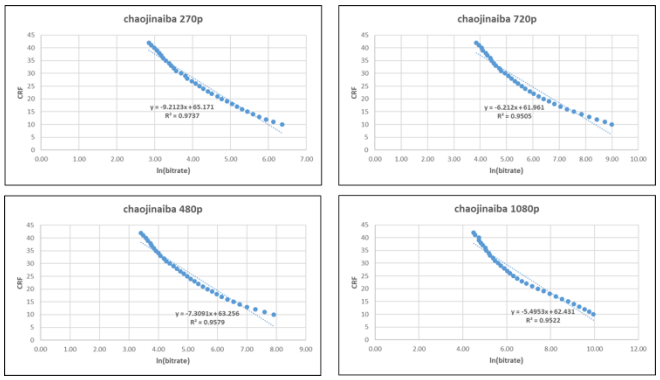

Fig. 2. Linear model estimation between CRFs and bitrates

Therefore, we need to find a more robust relationship between the bitrate and CRF. Based on some experiences that the QP is with the second model with the bitrate if a wide variety of QPs are needed to cover. This phenomenon motivates us to try the second order model between the

CRF and ln(R). The second order model we used can be described as follows.

$$QP = a(v)(\ln R)^2 + b(v)(\ln R) + c(v) \qquad (4)$$

where $a(v)$, $b(v)$, and c(v) are content-related model parameters. To test the performance of the second order model, we also test the proposed model on the same 720p sequences as we used for the linear model. The fitting results are shown in Table 2.

Table 2 The average bitrate error using the linear model

	error within 20%	Improve on linear model	error within 10%	Improve on linear model
270p	100%	2%	99%	12%
480p	100%	4%	99%	19%
720p	99%	10%	97%	35%
1080p	97%	15%	89%	40%
Avg.	**99%**	**8%**	**96%**	**27%**

From Table 2, we can see that the second order model can achieve that 99% and 96% of the total samples are within 20% and 10% bitrate errors, respectively. It outperforms the linear model by 8% and 27% accordingly. This result obviously demonstrates the effectiveness of the proposed second order model.

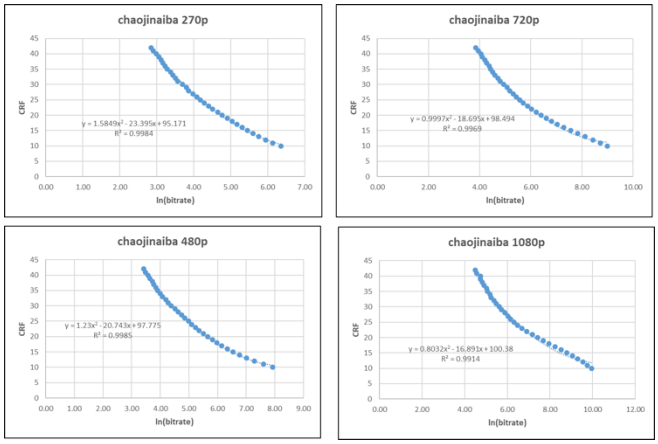

Fig. 3. Fitting curves between CRFs and bitrates, testing second order model

We also fit the failure case of the linear model using the proposed second order model as shown in Fig. 3. As shown in Fig. 3, the determining coefficient are all as high as 0.99 over all the resolutions. This can also partially explain the benefits of the proposed second order model.

4. PROPOSED MACHINE-LEARNING BASED TRAINING PROCESS

After determining the second order model, the remained problem is to estimate the content-related model parameters. In this work, we adopt a shallow network to estimate the model parameters. The proposed network structure is shown in Fig. 4. We derive 14 features from a specified CRF as input. The bitrate and CRF are also derived to fit the second order model to get the label. Only two-hidden layers are employed in our machine learning based model. The output will be the three parameters of the second order model.

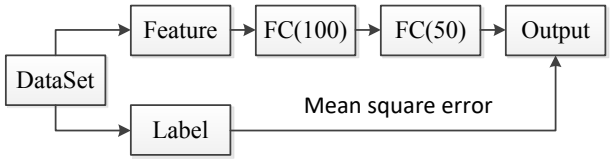

Fig. 4. The structure of machine-learning based network

To make the proposed network structure more obvious, we list of all the derived 14 features as follows.

- Average PSNR of each channel & only Y channel
- Encoded Bitrate
- Percentage of texture bits on I & P macroblocks (MB) relative to the total bits
- Percentage of texture bits relative to total bits
- Average number of bits for texture for each I & P MBs
- Average number of bits for texture for each MBs
- Percentage of I & P MB
- Percentage of zero motion vector
- Average of motion vector for P frames and B frames
- Average MVD bits

These features include both the frame level and MB level features to fully reflect the characteristic of the specified content.

We implement our network on TensorFlow with Python. First, after extracting the 14 features and the 3 labels from many training sequences, we do pre-process for them with MinMaxScaler package supported by Python on features and StandardScaler package on labels. The MinMaxScaler package is used to re-distribute the data to the range between 0 and 1, and the StandardScaler package is an operation to distribute data by subtracting the mean and dividing the standard variance. Then we feed groups of features after initialization in our 2 fully-connected layers network, with 100 neutrons and 50 neutrons on the first and the second layer. In the proposed network model, as the dimension of the input features is very low, the fully-connected layers structure is suitable to characterize the purity feature extraction for the labels we expected. On both layers, the hyperbolic tangent algorithm (Tanh) is used as the motivation function. We set up our gradient descent optimizer with learning rate equal to 0.05 to train our network. The loss function used is the mean square error between the predicted model parameters and the

actual model parameters. After the whole network is trained, the network will then be used to generate the model parameters for each specified sequence. The CRF calculation and encoding process will be finished afterwards.

5. EXPERIENTIAL RESULT

We adopt totally 27,732 samples of 5 second sequences as our training dataset. These test samples are cut from over 1000 2min test sequences with different video content to ensure the variety of the test sequences. In the meantime, another 4,787 samples with totally different contents are used as our test dataset to evaluate the performance of the network trained from the training samples. To fully test the performance of the proposed framework, we test the performance on 4 resolutions including 270p, 480p, 720p, and 1080p and different CRFs ranging from 10 to 42.

During training process, we employ only one specific CRF setting on each resolution to obtain the network since we can only perform one pass coding to obtain the model parameters. However, in the test process, we need to adapt the model to all the CRFs for a specified sequence. All samples split into three sets, 80% of 27,372 as training set, the rest of 20% as validation set and another 4,787 as test samples. The detailed CRF selections are from a number of tests on various CRFs. We choose CRF settings shown the best performance from all the test CRFs from 10 to 42. For a specified resolution 480p, we show the changing of the bitrate errors from 20 to 30 as shown in Fig. 5. From Fig. 5, we can see that, there are not that many differences for all the CRFs for both the bitrate errors under 10% and 20%, we select the one which optimizes the average performance for the test samples.

We compare the percentages of the test samples under 20% and 10% bitrate errors of the proposed framework with the linear model. The detailed experimental results are shown in Table 3. From Table 3, we can see that the proposed algorithm can achieve about 90.9% and 72.3% of total samples within 20% and 10% bitrate errors, respectively. Compared with the linear model, the proposed algorithm can achieve about 4.7% and 13.4% more examples under 20% and 10% bitrate errors accordingly.

For each specified resolution, we can see that the linear model can achieve slightly better performance compared with the second order model for the 270p case. Comparing the 270p resolution results in Table 1 and Table 2, we can see that the difference between the linear model and the second order model is relatively small compared with other resolutions. However, the linear model parameters are easier to estimate through the network. That is why the linear model achieves slightly better performance. For the other resolutions, the proposed framework provides much better results compared with the linear model. The higher the resolution, the better the performance. Especially, under the 1080p case, the proposed algorithm can achieve 11.5% and 24.4% increase under 20% and 10% cases.

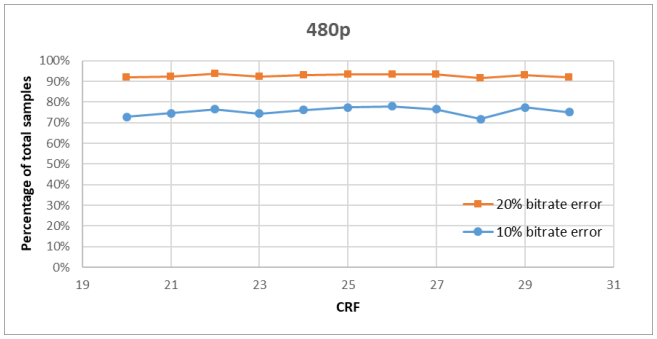

Fig. 5 The changing of 20% and 10% average bitrate errors for the 480p test sequences

Table 3. The performance of the proposed framework and its improvement over the linear model

	error within 20%	Improve on linear model	error within 10%	Improve on linear model
270p	93.5%	95.0%(**-1.5%**)	72.9%	70.0%(**-2.9%**)
480p	93.4%	91.7%(**1.7%**)	77.3%	66.4%(**10.9%**)
720p	91.6%	84.5%(**7.1%**)	73.5%	52.4%(**21.1%**)
1080p	85.2%	73.7%(**11.5%**)	65.2%	40.8%(**24.4%**)
Avg.	**90.9%**	**86.2%(4.7%)**	**72.3%**	**58.9%(13.4%)**

6. CONCLUSION AND FUTURE WORK

In this paper, we provide a two-pass coding framework to solve the problem of sequence level rate control. In the framework, we first propose a second order model to model the relationship between the bitrate and constant rate factor. We also propose to use a simple network structure to estimate the model parameters accurately. The proposed algorithm is implemented under the tensorflow framework. The experimental results show that the proposed algorithm can achieve 4.7% and 13.4% more samples within 20% and 10% bitrate errors, respectively.

In the future, we will further optimize the network by choosing some block or region-based features. Also, we can use some pixel information if the reconstructed videos can be used. Finally, under the block-based features, we will further try some convolution neural networks in the network structure.

7. **REFERENCES**

[1] T. Chiang and Y. Zhang, “A new rate control scheme using quadratic rate distortion model,” *in IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, no. 1, pp. 246-250, Feb. 1997.

[2] S. Ma, W. Gao and Y. Lu, “Rate-distortion analysis for H.264/AVC video coding and its application to rate control,” *in IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 12, pp. 1533-1544, Dec. 2005.

[3] Z. He, Y. K. Kim and S. K. Mitra, “Low-delay rate control for DCT video coding via ρ-domain source modeling,” *in IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 8, pp. 928-940, Aug. 2001.

[4] G. J. Sullivan, J. R. Ohm, W. J. Han and T. Wiegand, “Overview of the High Efficiency Video Coding (HEVC) Standard,” *in IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649-1668, Dec. 2012.

[5] B. Li, H. Li, L. Li and J. Zhang, " λ Domain Rate Control Algorithm for High Efficiency Video Coding," *in IEEE Transactions on Image Processing*, vol. 23, no. 9, pp. 3841-3854, Sept. 2014.

[6] http://slhck.info/video/2017/02/24/crf-guide.html.

[7] M. Covell, M. Arjovsky, Y. Lin, and A. Kokaram, “Optimizing Transcoder Quality Targets Using a Neural Network with an Embedded Bitrate Model”, *Electronic Imaging*, Visual Information Processing and Communication VII, pp. 1-7(7).

[8] B. Li, J. Xu, D. Zhang, and H. Li, “QP refinement according to Lagrange Multiplier for High Efficiency Video Coding”, *Circuits and Systems, 2013 IEEE International Symposium on*, pp. 477-480.