CNN BASED POST-PROCESSING TO IMPROVE HEVC

Chen Li[†], *Li Song*[*†], *Rong Xie*[†], *Wenjun Zhang*[*†]

[*]Institute of Image Communication and Network Engineering, Shanghai Jiao Tong University
[†]Cooperative Medianet Innovation Center, Shanghai, China

ABSTRACT

In this paper, we propose a frame-based dynamic metadata post-processing scheme in HEVC. Video sequence is classified into different categories contains complexity of video content and quality indicator for each frame, an up-to-one byte flag embedded in the bitstream is transferred as side information. Meanwhile dynamic metadata contains classification information indicates the offline training of separate network models. Specifically, we adopt a 20-layers CNN (Convolutional Neural networks) model to extract more meaningful information from the reconstructed error and improve the filtering performance. Experimental results shows that our proposed post-processing scheme leads on average 1.6% BD-rate reduction compared with HEVC baseline on the six sequences given in *2017 ICIP Grand Challenge*.

***Index Terms*—** HEVC, CNN, dynamic metadata, post-processing, out-of-loop

1. INTRODUCTION

In lossy video compression algorithms, there exist lots of visually annoying artifacts. In order to alleviate these artifacts, two post-processing techniques are incorporated into the state-of-the-art High Efficiency Video Coding (HEVC) standard, namely deblocking filter and sample adaptive offset (SAO). Both techniques reduce the prediction residual of the subsequent coding pixels, and effectively improve the subjective and objective quality of video.

Recently, there have been many progresses achieved using Convolutional Neural networks (CNN) for video filtering. For example, Park and Kim [1] replace SAO with a modified SRCNN network. Dai *et al.* [2] propose a Variable-filter-size Residue-learning CNN (VRCNN) to replace both deblocking filter and SAO. However, the techniques mentioned above haven't consider the influence of features in video content. In the meantime, single network model cannot be suitable for all frames in a video sequence with high complexity.

In this paper, we adopt a CNN forward network to assist image reconstruction and a new post-processing approach that can well accommodate to multiple complexity. Firstly, we propose to utilize an up-to-one byte flag with side information contains the complexity information of video content and the dynamic QP allocated by the encoder, which is transferred out-of-loop to the post-processing module for selecting separate trained models. And the video complexity is reflected in both temporal domain and spatial domain. Secondly, a deeper network architecture [3] is adopted in this research, which could efficiently extract information from input image. Details of these techniques are given in the next section.

2. THE FRAMEWORK OF THE PROPOSED SCHEME

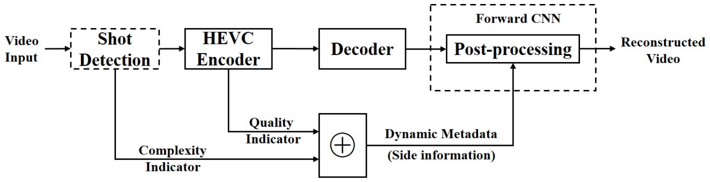

Fig. 1: Our proposed coding scheme.

The proposed encoding framework is shown in Fig. 1, where post-processing module is implemented using forward network. In this section, we will introduce the proposed scheme by detailed description of incorporated modules including shot detection, complexity analysis, dynamic metadata and CNN based post-processing.

2.1. Shot Detection

Different video content reflect to different complexity, and video information has a rapid growth in quantity. If we simply compute the complexity for the sequence with diverse content, the accuracy of complexity classification is doubtful. Therefore, it's necessary to segment the video into several subsequence with content before complexity analysis. A video shot is defined to be a sequence of images which are captured by a single camera in an uninterrupted run [4], and a shot could be consider as the basic unit of complexity computation.

In this scheme, the input video firstly flow into the shot detection module to implement an adaptive GOP. For each classified shot, a complexity information will be computed, then the complexity information will be transferred with the side

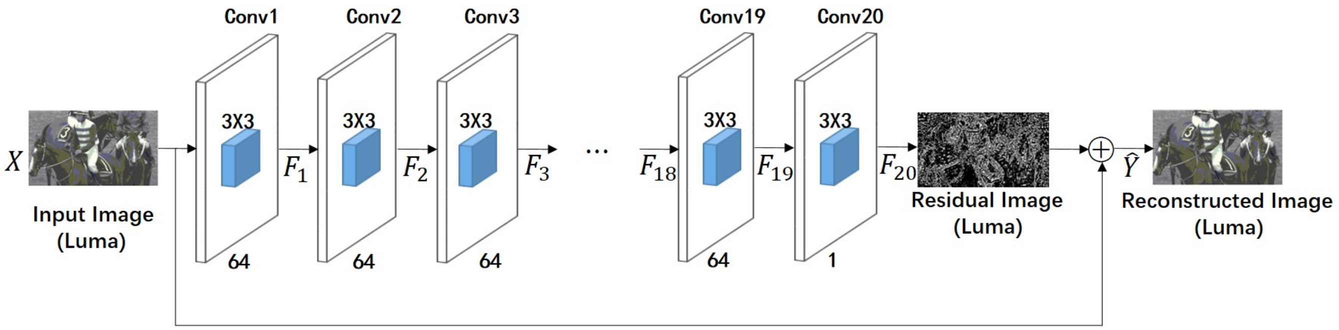

Fig. 2: VDSR Architecture.

information (the representation of different dynamic metadata) for the further partition. And all frames in one shot will adopt a same complexity information.

2.2. Complexity Indicator

There exist a variety of complexities in video sequences, which may be quiet different between two frames even in the same sequence. Therefore, if we simply train the network with indiscriminately natural image set and test with a sequence of complex content, the model will not be able to maximize the effectiveness of each frame.

We use the SI and TI [5] indexes to represent spatial and temporal complexity. SI is the maximum standard deviation after sobel filtering, TI stands for the differences between two consecutive frames based on pixels. For the *n* th frame F_n:

$$SI = max_{time}\{std_{space}[Sobel(F_n)]\} \tag{1}$$

$$TI = max_{time}\{std_{space}[F_n(i,j) - F_{n-1}(i,j)]\} \tag{2}$$

Applying to this structure, SI and TI indexes is computed through a shot rather than the whole sequence. In the aspect of complexity categories, the luminance component contains more structural information than the chrominance component, so we only analyse the luminance component of the sequences adopted in [5], and implement three specific classification criteria to represent High, Medium and Low complexity using K-means algorithm by SI and TI. According to these classified classes, it's easy to find a corresponding category for each shot.

2.3. Dynamic Metadata

Dynamic metadata is the representation of video category, it is transmitted as a side information in the bitstream contains the complexity and quantification information. As mentioned above, complexity is computed based on one shot, and all the frames in one shot share the same complexity category. Meanwhile quality is viewed as another classification metric, which is evaluated with QP in this paper.For the encode mode with fixed bitrate, HEVC encoder will allocate different QP for each frame. As we can see, quantification error is an important component of reconstruction error. Therefore, the network models will be trained out based on QP, which is a reasonable way to improve the reconstructed quality. After preprocessing the video data, it can be observed that most of allocated QPs exist in the range of 20 to 40. Considering the influence of quantification, we select six groups of frequently-used QPs (20, 24, 28, 32, 36, 40) and three ancillary QPs (15, 45, 50) to furthest simulate the distribution of QP. For each frame, the model with nearest QP is adopted and a side information contains categories is transmitted with a flag, whose length is up-to-one byte.

2.4. CNN-Based Post-processing

Previous research on image classification and image restoration has proved that CNN can efficiently extract the feature from input data. Therefore, using CNN to learn the residuals between the reconstructed image and original image on pixel level can approximate to the ground truth to the maximum extent.

In this research, we adopt a deep network called VDSR [3], which is showed in Fig. 2. Because in the task of image restoration, the image resolution is fixed, thus the interpolation before CNN is released. In other words, a compressed image flows into the module and processed by neural network, then a high quality image with the same resolution will be written to reconstructed video.

VDSR is a fully convolutional network with 20 layers, except the first and the last layer, the filter type of other layers is $3 \times 3 \times 64$, where filter size is 3×3 and implement 64 feature maps. The first layer extract the information of input image, and the last layer incorporate all the features for image reconstruction. In order to accelerate the convergence and solve the vanishing gradients problem, Residual-Learning technique is adopted. Input image is added to the reconstructed image directly, thus CNN only need to learn the distance between input and output. As input image is similar to origin image, the residual could be small and easy to approximate.

In VDSR, the output of the *i*-th layer can be represent as:

$$F_i = max(0, W_i * F_{i-1} + B_i), i \in \{1, 2..., 19\} \tag{3}$$

when *i*=1, F_0 represent the input image, so W_1 is of the size $3 \times 3 \times 64$, and the size of other W_i is $64 \times 3 \times 3 \times 64$. For the last layer, there exist no activation unit, it can be expressed as:

$$\hat{Y} = F_{20} = W_{20} * F_{19} + B_{20} \tag{4}$$

where $\hat{Y}$ is the predicted residues learned from the distance between original and reconstructed image, W_{20} is of a size $64 \times 3 \times 3$ and '*' represent 3-D linear convolution.

3. EXPERIMENTS

As previously mentioned, separate modules will be trained for each category. Referring to classified information, we cluster the luminance component of the 8 sequences in HEVC standard test sequence ClassC and ClassD into three categories. And these images could be viewed as the source of training set. According to the encode mode given in Grand Challenge [6], the bitrate is fixed. In a sequence, allocated QP may fluctuate on a large range compressed with fixed bitrate, and the difference between the reconstructed error of each frame will also be great, which will make the training process uncontrollable. Consider the relationship between bitrate and QP, we preliminarily select a compromised method to generate training data: the classified images are compressed using the same mode but with fixed QPs defined in the section 2.3 by x265. In this way, the distance between QPs and bitrates in two modes won't be very large. Then we use HM decoder to decode the output bitstream file and get the reconstructed video. In order to extend the training data, reconstructed and original images are divided into 35 × 35 sub-images on pairs without overlap. Finally, the order of the prepared image is shuffled and randomly choose 46784 images as training set.

3.1. Training

Let X_n denote the original training set, and Y_n denote the compressed image and input data, where $n \in \{1, 2..., N\}$ is the index of each image. The network learns the mapping function *F* by minimize the Euclidean loss *L*:

$$L(\Theta) = \frac{1}{N} \sum_{n=1}^{N} ||F(Y_n|\Theta) - X_n||^2 \tag{5}$$

where Θ denote the parameter set in VDSR including *W* and *B* of each layer, and these parameters are updated by the method of stochastic gradient descent. The adjustable gradient clipping method [3] is also adopted in this research. In which the gradient update is restricted in [τ/α, -τ/α], α is learning rate and τ is a constant. The gradient update correlate inversely to the learning rate, so this technique is beneficial to restrain exploding problem.

In this experiment, we use deep learning software CAFFE [7] to train the network, the weights are initialized using M-SRA [8], the mini-batch size is 64. The momentum parameter is set to 0.9, and weight decay is 0.0001. And the gradient clipping constant τ is 0.01. The base learning rate is set to decay from 0.1 to 0.0001 with the type of step, changing every 9620 iterations. Thus, it takes 38480 iterations totaly to train the network, which use a little more than one hour on GTX 980 Ti GPU.

3.2. Test in Given Sequence

For the purpose of verifying the performance of the proposed coding scheme, some modification should be applied to codec first. The category classified module is incorporated into x265 encoder and the side information is transferred with bitstream. Then the output binary file is decoded by the modified HM decoder which integrate the CNN forward network, while the side information is extracted by CNN based post-processing module. Based on the extracted side-information, corresponding module will be select to execute the task of filtering.

The test sequence is given in *2017 ICIP Grand Challenge* [6], there exist two resolutions: 854 × 480 and 352 × 288. For each sequence, six bitrate points is set. Compared with HEVC baseline, The reconstructed quality is measured by P-SNR and expressed by BD-rate [9]. In addition, we use only Y component of test sequence for simplicity.

Table 1: Performance of proposed scheme of BD-rate

Size	Sequence	BD-rate(%)
854 × 480	controlled_burn	-1.7
	park_joy	-1.1
	red_kayak	-1.4
352 × 288	football_cif	-1.9
	foreman_cif	-2.4
	flower_cif	-1.1
Overall	**All**	-1.6

3.3. Experiment Results

We use 6 sequences (three of high resolution, three of low resolution) in *2017 ICIP Grand Challenge* for the test, which include football_cif, foreman_cif, flower_cif (352 × 288) and controlled_burn, park_joy, red_kayak (854 × 480). All the luminance of 60 frames in each sequence is tested. Table 1 shows the performance compared between proposed coding scheme and HEVC baseline. As shown in Table 1, the coding

Table 2: Performance of football sequence of PSNR

Bitrate(kb/s)	4400	2800	1600	1200	600	200
PSNR(dB)	39.36	36.38	33.04	31.41	28.02	23.67
Baseline(dB)	39.28	36.27	32.85	31.28	27.95	23.64
Gain(dB)	0.08	0.1	0.19	0.13	0.07	0.03
BD-PSNR(dB)	0.1					

scheme shows its effectiveness for improving average 1.6% BD-rate, and for the sequence foreman, 2.4% BD-rate reduction is achieved.

Table 2 shows the performance of PSNR on the football sequence, this coding scheme significantly improve the coding performance at medium bitrates, and relatively speaking, the scheme performs worse at high or low bitrate. The video compressed with high bitrates are usually allocated with lower QPs, which produce smaller reconstructed error so that it is difficult for CNN to learn meaningful information. As for low bitrates, the sequence will be compressed with a poor quality. Although the reconstructed error is large enough, but serious artifacts make some important features blurry. The neural network can not extract enough information to restore the image compared with medium bitrate.

4. CONCLUSION

This paper introduces a new post-processing method to improve HEVC adaptive to different video content and quality. An out-of-loop flag is adopted to transfer the side information contains SI, TI indexes and allocated QP at fixed bitrate for selecting the optimum model. Meanwhile VDSR with 20 convolutional layers is adopted in this paper, which could extract more meaningful feature to improve the restoration performance. Compared to the HEVC baseline, on average 1.6% BD-rate (in luminance) is achieved on the six sequences given in Grand Challenge. To some extent, the experimental results verify the effectiveness of our proposed coding scheme.

Our future work is in two directions. First, it is necessary to develop a optimum method to implement the training data which can associate QP closely with bitrate, so that the network could be trained out effectively. Second, we will pay close attention to the state-of-the-art technique in deep learning and investigate how to develop a simple but effective network for image restoration.

5. ACKNOWLEDGEMENTS

This work was supported by NSFC (61671296 and 61521062), the 111 Project (B07022 and Sheitc No.150633) and the Shanghai Key Laboratory of Digital Media Processing and Transmissions.

6. REFERENCES

[1] W. S. Park and M. Kim, "Cnn-based in-loop filtering for coding efficiency improvement," in *Image, Video, and Multidimensional Signal Processing Workshop (IVMSP), 2016 IEEE 12th*. IEEE, 2016, pp. 1–5.

[2] Y. Dai, D. Liu, and F. Wu, "A convolutional neural network approach for post-processing in hevc intra coding," in *International Conference on Multimedia Modeling*. Springer, 2017, pp. 28–39.

[3] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1646–1654.

[4] Jingwei Xu, Li Song, and Rong Xie, "Shot boundary detection using convolutional neural networks," in *Visual Communications and Image Processing (VCIP), 2016*. IEEE, 2016, pp. 1–4.

[5] Y. Zhu, L. Song, R. Xie, and W. Zhang, "Sjtu 4k video subjective quality dataset for content adaptive bit rate estimation without encoding," in *Broadband Multimedia Systems and Broadcasting (BMSB), 2016 IEEE International Symposium on*. IEEE, 2016, pp. 1–4.

[6] Grand Challenge ICIP 2017, "Grand challenge on the use of image restoration for video coding efficiency improvement," `https://storage.googleapis.com/icip-2017/index.html`.

[7] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, 2014, pp. 675–678.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[9] G Bjøntegaard, "Calculation of average psnr differences between rd-curves (vceg-m33)," in *VCEG Meeting (ITU-T SG16 Q. 6)*, 2001, pp. 2–4.