Perceptual in-Loop Filter for Image and Video Compression

Huairui Wang[1*] Guangjie Ren[1*] Tong Ouyang[1*] Junxi Zhang[1] Wenwei Han[1]
Zizheng Liu[2] Zhenzhong Chen[1†]
[1]Wuhan University, Wuhan, China [2]Tencent Media Lab, Shenzhen, China
zzchen@whu.edu.cn

Abstract

In this paper, we introduce our hybrid image and video compression scheme enhanced by CNN-optimized in-loop filter. Specifically, a Structure Preserving in-Loop Filter (SPiLF) is incorporated in the hybrid video codec Enhanced Compression Model (ECM), where two branches, i.e., gradient branch and pixel branch, are developed based on the dense residual unit (DRU). To provide pleasant visual quality, the Generative adversarial networks (GAN) loss and LPIPS loss are further considered. Therefore, the proposal is mainly focusing on perceptual-friendly image compression for human vision, whilst video compression could be further investigated. The experiments show that the proposed method achieves advanced visual quality when compared to the traditional methods.

1. Introduction

Nowadays, the explosive growth of multimedia data poses a huge challenge for storage and transmission. There are many image/video compression algorithms that have been developed, e.g. JPEG, JPEG2000, BPG (based on H.265/HEVC intra coding), and AV1. The latest video coding standard Versatile Video Coding (VVC) has been finished in July 2020. It aims at yet another 50% bit-rate reduction compared to H.265/HEVC, and provides a range of additional functionalities [3]. Although the block-based prediction and quantization in VVC are indeed effective, flexible, and popular, there are some consequences in the reconstructed frames using these methods, known as blocking artifacts. The visual quality of reconstructed frames can be severely damaged due to the discontinuities at the edges between blocks. To eliminate the artifacts, some conventional tools are added to the in-loop filter to alleviate visual discomfort without adding too many bits, including deblocking filter (DBF), Sample Adaptive Offset (SAO), and Adaptive Loop Filter (ALF), which are implemented sequentially in VVC. However, under the circumstances of low bit-rate, the decoded videos still suffer from noticeable compression artifacts. Moreover, the optimization distortion of the hybrid video codec is mean square error (MSE), which may cause a mismatch between the subjective and objective perceptions. With the development of the deep neural network, learning based image/video compression has drawn more attention. Convolutional Neural Networks (CNN) based in-loop filter has been investigated and demonstrated its advantage to improve perceptual quality further.

In this paper, we propose a perceptual-oriented structure-preserving in-loop filter (SPiLF) method for image/video compression. The network of SPiLF consists of two branches, one is the Gradient branch (GB) for enhancing the information in high frequency than the conventional networks [6] and the other one is a conventional pixel branch that restores the reconstructed frames with the enhanced feature map of gradient map. Besides, the Generative adversarial networks (GAN) [7] loss and LPIPS [9] loss are further considered to improve subjective visual quality. The experimental results demonstrate the superior performance of the proposed method.

2. Proposed Method

In the proposed method, we develop our codec based on the Enhanced Compression Model (ECM) platform [1], which is developed based on Versatile Video Coding Test Model (VTM). The SPiLF is integrated into ECM to enhance the coding efficiency. The proposed framework is shown in Figure 1.

2.1. Structure Preserving in-Loop Filter

The overall framework is shown in Figure 1 where the proposed SPiLF consists of two branches. One is a conventional compression artifact removal branch in the pixel domain, and the other is a gradient branch. The decoded frames are taken as inputs for both branches, and the recovered frames are generated with the guidance provided by the gradient branch, which aims to produce gradient maps as similar as possible to their uncompressed counterparts.

*Co-first author. †Corresponding author.

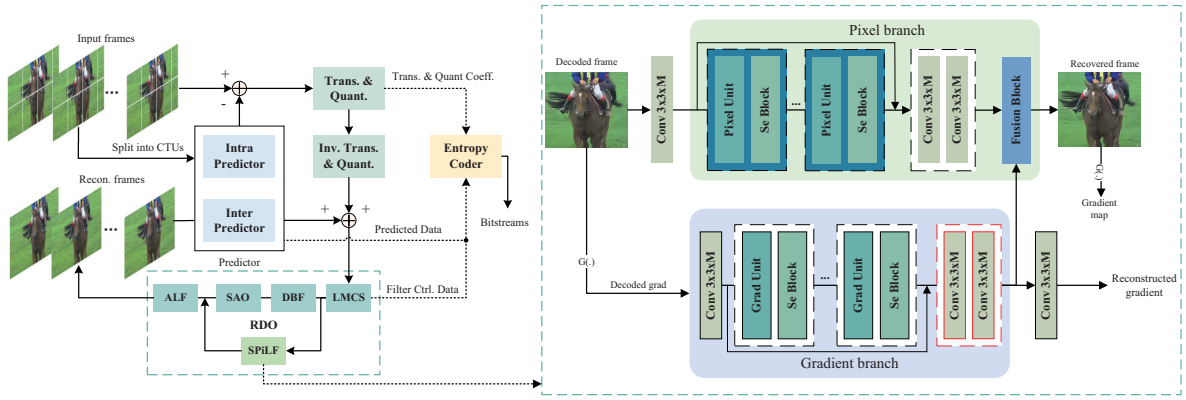


Figure 1. The scheme of the Perceptual in-Loop Filter enhanced image and video compression method and the detailed architecture of our proposed SPiLF.

2.1.1 Gradient Branch

The gradient branch is proposed to generate better gradient maps with compressed textures and provide professional guidance to the pixel branch in restoring the decoded frames as well. A function for extracting the gradient maps of frames denoted as $G(\cdot)$, is designed as follows:

$$G(f_{x,y}) = \sqrt{(f_{x-1,y} - f_{x+1,y})^2 + (f_{x,y-1} - f_{x,y+1})^2}$$

where $f_{x,y}$ denotes a pixel located at (x, y) in a frame. The gradient maps obtained from decoded frames are fed into a residual network, which consists of multiple grad blocks and squeeze-and-excitation block (SE block) [4]. In our approach, the dense residual unit (DRU) [8] is adopted to reproduce the lost textures in decoded grad. Before outputting the reconstructed gradient maps, the feature maps of restored gradients are incorporated into the pixel branch.

2.1.2 Pixel Branch

As for the pixel branch, the network is designed on the basis of structure preserving and artifacts removing architecture, which mainly consists of two parts. The first part is a sequential residual structure with multiple pixel units that could be replaced by any basic neural architecture. We adopt DRU as the pixel unit, in consideration of its superior performance in denoising tasks. Every pixel unit is followed with a SE block to enhance the output feature maps, which assign greater weight automatically to channels that are more important in the output feature maps. The second part uses the output features from the first part and gradient branches. More specifically, the information from two branches is concatenated and fed into a fusion block, which adopts the DRN in this work. Some convolution blocks are also utilized to generate the final reconstructed frames.

2.2. Perceptual Loss Function

We adopt a perceptual loss function for the CNN-based in-Loop Filter in the hybrid compression framework to further enhance the subjective visual quality. The whole loss function consists of four parts: distortion, perceptual, adversarial, and structural loss. We use L1 loss as the basic distortion loss in the pixel domain to force low-frequency correctness and use LPIPS loss to improve high-level feature fidelity.

As for the adversarial loss, we follow PatchGAN [5] to adopt Markovian discriminator to focus on modeling high-frequencies at the scale of patches and utilize Relativistic GAN loss [7] which helps to learn sharp edges and more detailed textures. We denote the Patch-wise Relativistic average Discriminator as D_{Pat}, which can be formulated as:

$$D_{Pat}(x_r, x_f) = \sigma(C_{Pat}(x_r - \mathbb{E}_{x_f}[C_{Pat}(x_f)])),$$

where σ is the sigmoid function, $C_{Pat}(x)$ is the patch-wise discriminator output, x_f and x_r are the output images of the SPiLF and the ground truth, respectively. The GAN loss function can be stated as follows:

$$\begin{aligned}\mathcal{L}_D = &- \mathbb{E}_{x_r}[log(D_{Pat}(x_r, x_f))] \\ &- \mathbb{E}_{x_f}[log(1 - D_{Pat}(x_f, x_r))], \\ \mathcal{L}_G = &- \mathbb{E}_{x_r}[1 - log(D_{Pat}(x_r, x_f))] \\ &- \mathbb{E}_{x_f}[log(D_{Pat}(x_f, x_r))],\end{aligned}$$

The structural loss $\mathcal{L}_{ST}$ is designed for improving the structural similarity between the output and the ground truth with gradient maps. In summary, the overall loss function can be formulated as:

$$\mathcal{L}_{total} = \lambda_1 \times \mathcal{L}_1 + \lambda_2 \times \mathcal{L}_{LPIPS} + \lambda_3 \times \mathcal{L}_{GAN} + \lambda_4 \times \mathcal{L}_{ST}.$$

where λ_i is the weight of each loss. We modify the λ parameter list for different optimization target, and the detailed

settings are illustrated in Section 3.

2.3. Resource Allocation

The task of this challenge is to choose the allocation rate of each compressed image to minimize the overall distortion under a given rate constraint. Under the constrained optimization purpose, we find that the task can be transformed into a constrained programming problem:

$$\arg\min \sum_{i=1}^{N}\sum_{j=1}^{M} D_i(Q_j) \times x_{ij}$$
$$s.t. \sum_{i=1}^{N}\sum_{j=1}^{M} R_i(Q_j) \times P_i \times x_{ij} \leq T$$

where Q_j denotes the j_{th} quantization parameter (QP) in our search space, $D_i(Q_j)$ and $R_i(Q_j)$ are the distortion and rate cost when the image is compressed with QP Q_j. P_i denotes the pixel number of the i_{th} image, x_{ij} represents the flag whether quality j_{th} is chosen for compressing the i_{th} image, which subjective to $\forall i \sum_{j=1}^{M} x_{ij} = 1$. T means the total target bits.

We use the Linear Integer Programming method to solve the problem. Specifically, After establishing the constraint equation and the objective equation, we use the public solver to get the optimal index combination. In order to facilitate the calculation, the solver requires the coefficients to be integers, so we scale all the parameters uniformly to meet the requirements.

3. Experiments

3.1. Dataset

We adopt DIV2K [2] as the training dataset which contains 800 images for training and 100 images for validation with 2K resolution. For training set generation, we compress DIV2K training set using eight different QPs = $\{22, 27, 32, 37, 42, 47, 52, 57\}$ with DBF, SAO and ALF disabled under all-intra configuration.

3.2. Training Setting

3.2.1 Progressive Training

GAN-based image compression task often suffers from unstable training and undesired objective performance, especially at low bit-rate. So we perform a two-stage training strategy to achieve stable training. First, we only use L_1 and L_{ST} to optimize the model. After the entire network converges, the total loss is applied for the final finetune.

3.2.2 Loss Function Weights

For different optimization targets (PSNR and Perceptual), we apply different loss functions on the in-loop filter model of the same architecture. Table 1 shows the detailed weights for the proposed loss functions. We remove perceptual oriented loss on Arabica to focus on the improvement of PSNR, and set λ_2, λ_3 to 50 and 0.05 for ArabicaPerceptual to improve high-level feature fidelity and generate more realistic results. The gradient branch and the structural loss are utilized to restore the structural information of the image, so its existence mainly improves the subjective quality.

Team Name	λ_1	λ_2	λ_3	λ_4
Arabica	1	0	0	5e-3
ArabicaPerceptual	1	50	5e-2	5e-3

Table 1. The loss function weight setting of the proposed Arabica and ArabicaPerceptual methods.

3.2.3 Training Details

We set the batch size to 32, and randomly crop the training sequences into the resolution of 256×256. The Adam optimizer is adopted whose parameters β_1 and β_2 are set as 0.9 and 0.999. The learning rate is initialized to 1×10^{-4} and decreased by a factor of 2 when evaluation performance becomes stable. The entire network converges after 450 epochs. All experiments are conducted using the PyTorch with NVIDIA GTX 3090 GPUs.

3.3. Performance Analysis

We compare our methods with two traditional hybrid compression methods: BPG and ECM, and all experiments are conducted on CLIC 2022 validation set for the image track. To verify the performance of our proposed schemes, the objective results of different compression methods are shown in Table 2. It can be observed that the Arabica will surpass ECM by 0.15 to 0.2 dB at the same bpp on PSNR, and the ArabicaPerceptual performs the best in terms of LPIPS among all methods.

4. Concluding Remarks and Discussions

In this paper, we propose the hybrid compression method with perceptual in-loop filter. To be specific, considering structural deformation caused by the GAN-based loss function, we propose the Structure-Preserving in-Loop Filter and the gradient loss for constraining the output to maintain accurate structure. For perceptual optimization, LPIPS and PatchGAN are utilized to generate more plausible results. The proposal is mainly focusing on perceptual-friendly image compression for human vision. For video compression, as the GAN-loss optimized video frame might cause the coding efficiency degradation for inter-frame motion compensation, it worthy further study instead of simply apply the GAN-loss optimization for non-reference video frames.

Method Name	0.075 bpp			0.15 bpp			0.3 bpp		
	PSNR↑	MS-SSIM↑	LPIPS↓	PSNR↑	MS-SSIM↑	LPIPS↓	PSNR↑	MS-SSIM↑	LPIPS↓
BPG444	27.989‡	0.9201‡	-	30.214‡	0.9493‡	-	32.827‡	0.9695‡	-
ECM	28.233	0.9345	0.308	30.423	0.9603	0.235	32.692	0.975	0.170
Arabica	28.352‡	0.9349‡	0.304	30.604‡	0.9609‡	0.231	32.890‡	0.9762‡	0.169
ArabicaPerceptual	28.464‡	0.9333‡	0.254	29.945‡	0.9541‡	0.197	32.391‡	0.9733‡	0.128

‡: Results copied from CLIC 2022 Validation set Leaderboards.

Table 2. Objective results comparison on CLIC 2022 Validation set.

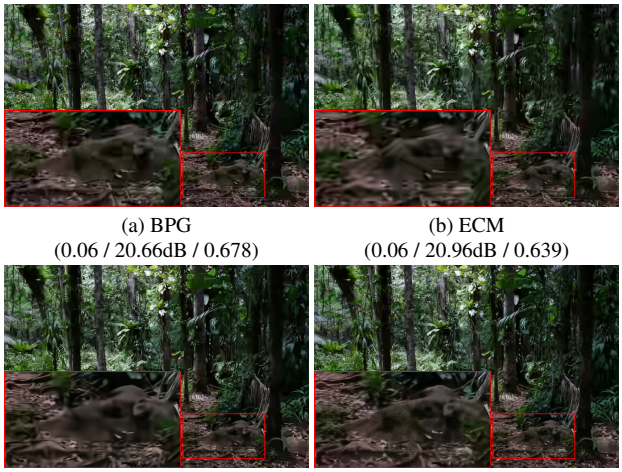
(a) BPG (0.02 / 27.02dB / 0.552)

(b) ECM (0.02 / 28.76dB / 0.524)

(c) Arabica (0.02 / **28.96dB** / 0.531)

(d) ArabicaPerceptual (0.02 / 28.65dB /**0.464**)

Figure 2. The human content visual results and their enlarged details of traditional hybrid codecs BPG, ECM, the proposed Arabica and ArabicaPerceptual. The parentheses contain (BPP / PSNR / LPIPS).



(a) BPG (0.06 / 20.66dB / 0.678)

(b) ECM (0.06 / 20.96dB / 0.639)

(c) Arabica (0.06 / **21.04dB** / 0.642)

(d) ArabicaPerceptual (0.06 / 20.89dB / **0.543**)

Figure 3. The forest content visual results and their enlarged details of traditional hybrid codecs BPG, ECM, the proposed Arabica and ArabicaPerceptual. The parentheses contain (BPP / PSNR / LPIPS).

References

[1] Enhanced Compression Model (ECM ver. 3.1). https://vcgit.hhi.fraunhofer.de/ecm/ECM.git Accessed: March 20, 2022. 1

[2] Eirikur Agustsson and Radu Timofte. NTIRE 2017 challenge on single image super-resolution: Dataset and study. In *CVPR Workshop*, 2017. 3

[3] Benjamin Bross, Jianle Chen, Jens-Rainer Ohm, Gary J. Sullivan, and Ye-Kui Wang. Developments in International Video Coding Standardization After AVC, With an Overview of Versatile Video Coding (VVC). *Proceedings of the IEEE*, pages 1–31, 2021. 1

[4] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, 2018. 2

[5] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017. 2

[6] Cheng Ma, Yongming Rao, Yean Cheng, Ce Chen, Jiwen Lu, and Jie Zhou. Structure-preserving super resolution with gradient guidance. In *CVPR*, 2020. 1

[7] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *ECCV Workshop*, 2018. 1, 2

[8] Yingbin Wang, Han Zhu, Yiming Li, Zhenzhong Chen, and Shan Liu. Dense residual convolutional neural network based in-loop filter for hevc. In *VCIP*, 2018. 2

[9] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 1