Content Adaptive Latents and Decoder for Neural Image Compression (Supplementary Materials)

Guanbo Pan[1], Guo Lu[2], Zhihao Hu[1], and Dong Xu[3](✉)

[1] School of Software, Beihang University
[2] School of Computer Science and Technology, Beijing Institute of Technology
[3] Department of Computer Science, The University of Hong Kong
dongxudongxu@gmail.com

A Experiments on CLIC 2022 Validation Dataset

We also evaluate our methods on the newly released CLIC [1] 2022 validation dataset, which contains 30 high-resolution images collected from unsplash.com. We compare our methods with the baseline method [6] and traditional codec BPG [3]. As shown in Fig. S1, our methods improve the corresponding baseline method by about 0.3 dB at all bit-rates in terms of PSNR. Our methods on the context version outperform the baseline method [6] and BPG [3], which demonstrates the effectiveness of our proposed methods.

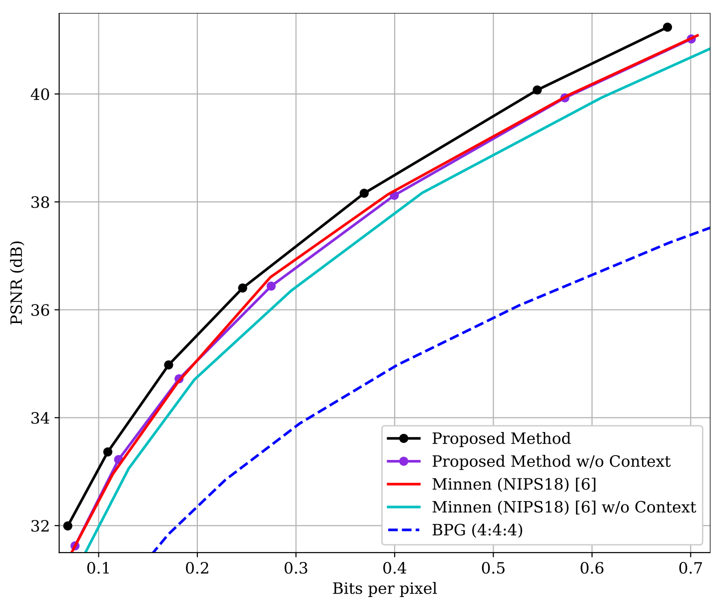

Fig. S1. Rate-distortion evaluation results on the CLIC 2022 validation dataset.

B More Qualitative Comparison Results

We provide more qualitative comparison results to demonstrate that our methods can adapt to the image content in order to preserve more details with smaller bit-rate in Fig. S2 and Fig. S3. In Fig. S2, we observe that the details of the plants are more clear in our methods. In Fig. S3, it is observed that the shallow letters on the sail of the right boat reconstructed by our methods are clearer than those reconstructed by BPG or [6] and our methods also preserve more details for the waves raised by the left boat than other two methods.

Fig. S2. Additional qualitative comparison results of the traditional codes BPG [3], the neural image compression method Minnen et al. [6] and our proposed methods.

C Test Settings of Traditional Video Codecs

For the rate-distortion performance evaluation in neural video compression, we follow the settings in [5], which uses the FFmpeg to compress the videos by H.264 and H.265 with the default mode. Given an uncompressed video *video.yuv* with the resolution of $W \times H$, we compress it to *output.mkv* with H.264 using the following command line as

ffmpeg -pix_fmt yuv420p -s WxH -r FR -i video.yuv -vframes N -c:v libx264 -tune zerolatency -crf Q -g GoP -sc_threshold 0 output.mkv

And the command line for H.265 is

Ground Truth

BPG
0.1175bpp, 30.90dB, 0.9421

Minnen et al. [6]
0.1328bpp, 31.81dB, 0.9550

Ours
0.1162bpp, 32.12dB, 0.9578

Fig. S3. Additional qualitative comparison results of the traditional codes BPG [3], the neural image compression method Minnen et al. [6] and our proposed methods.

ffmpeg -pix_fmt yuv420p -s WxH -r FR -i video.yuv -vframes N -c:v libx265 -tune zerolatency -x265-params "crf=Q:keyint=GoP:verbose=1" output.mkv

where *FR, N, Q, GoP* represent the frame rate, the number of frames to be encoded, the quality and the GoP size. Q is set as 20, 23, 26, 29. GoP is set as 10 in our experiments.

D Runtime Analysis

We conduct experiments on the machine with a single NVIDIA RTX 2080Ti GPU to evaluate the coding speed of our proposed content adaptive feature transformation (CAFT) method on the Tecnick [2] dataset, which is averaged over the dataset in millisecond. The result is provided in Table S1.

Table S1. BDBR(%) results and decoding speed (ms) on the Tecnick dataset. Negative values indicate bit-rate saving. *Synthesis time* calculates the inference time from latents to reconstructed image and *Decoding time* calculates that for both entropy parameter estimation and image reconstruction.

Decoder	BDBR(%)	Synthesis Time	Decoding Time
Minnen *et al.* [6]	0	20.30	596.33
CAFT	-7.98	82.22	658.89

The results demonstrate that our CAFT is effective to improve the baseline method with the BDBR [4] results of -7.98%. Although it increases the decoding time by 10.49%, it is acceptable in the whole decoding procedure comparing to the time-consuming context model which accounts for a large proportion.

For the content adaptive channel dropping (CACD) method, it increases computational costs because it forwards the decoder network K times, while that only happens during the encoding process where CACD costs much less than SGA [7] requiring 2000 iterations with both forward and backward.

References

1. CLIC · Challenge on Learned Image Compression (2022), URL http://compression.cc/ accessed on March 14, 2022
2. Asuni, N., Giachetti, A.: TESTIMAGES: a large-scale archive for testing visual devices and basic image processing algorithms. In: Giachetti, A. (ed.) Italian Chapter Conference 2014 - Smart Tools and Apps in computer Graphics, STAG 2014, Cagliari, Italy, September 22-23, 2014. pp. 63–70. Eurographics (2014)
3. Bellard, F.: Bpg image format. URL https://bellard.org/bpg (2015)
4. Bjontegaard, G.: Calculation of average psnr differences between rd-curves. ITU-T VCEG-M33, April, 2001 (2001)
5. Lu, G., Ouyang, W., Xu, D., Zhang, X., Cai, C., Gao, Z.: DVC: an end-to-end deep video compression framework. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019. pp. 11006–11015. Computer Vision Foundation / IEEE (2019)
6. Minnen, D., Ballé, J., Toderici, G.: Joint autoregressive and hierarchical priors for learned image compression. In: Bengio, S., Wallach, H.M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R. (eds.) Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada. pp. 10794–10803 (2018)
7. Yang, Y., Bamler, R., Mandt, S.: Improving inference for neural image compression. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H. (eds.) Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual (2020)