Joint Global and Local Hierarchical Priors for Learned Image Compression – Supplementary Material –

Jun-Hyuk Kim[1] Byeongho Heo[2] Jong-Seok Lee[1]
[1]School of Integrated Technology, Yonsei University [2]NAVER AI Lab
{junhyuk.kim, jong-seok.lee}@yonsei.ac.kr bh.heo@navercorp.com

A. Effectiveness of Informer

In our main paper, we evaluated our entropy model (Informer) with the encoder-decoder structure in Minnen *et al.* [8]. Additionally, in this supplementary material, we evaluate Informer with another encoder-decoder structure in Cheng *et al.* [4], which consists of residual blocks and attention modules. For the baseline entropy model [8], we use the result values reported in the CompressAI Github page[1].

As shown in Fig. A.1, Informer shows superiority across all PSNR levels on Kodak [6] with performance gaps from 3.03% up to 5.25%. From this result, we observe that our proposed Informer effectively models remaining dependencies in the quantized latent representation regardless of the encoder-decoder structure.

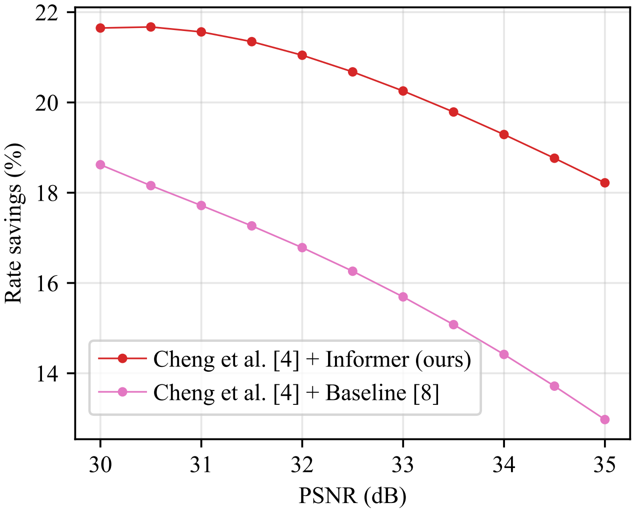


Figure A.1. Performance of different image compression methods. Each curve means the rate savings (%) relative to BPG [3] at different PSNR levels. Larger values mean better performance. The results of the MSE-optimized methods are averaged over Kodak [6].

[1] https://github.com/InterDigitalInc/CompressAI/blob/80ffc32/results/kodak/compressai-cheng2020-attn_mse_cuda.json

B. More implementation details

We illustrate the overall diagram of the learned image compression method with Informer in Fig. B.1. In addition, architectural details of Informer are summarized in Tab. B.1.

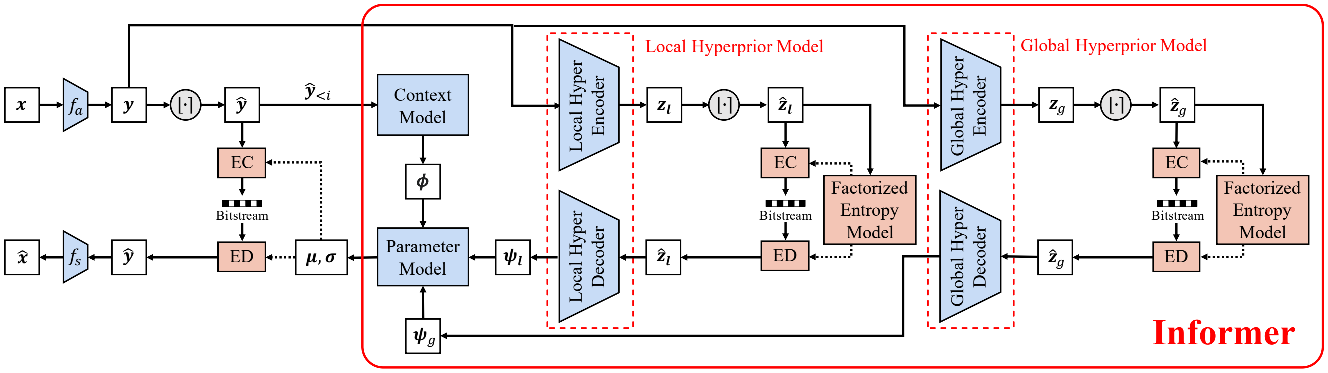


Figure B.1. Overall diagram of the learned image compression method with the proposed entropy model, Informer. The white blocks are data tensors, the blue blocks are learned models, the red blocks are entropy coding, and the gray circles are quantization operations. Informer can be combined with any transformation part (consisting of an encoder f_a and a decoder f_s). Informer jointly optimizes *Context Model* (capturing dependencies in the previously decoded local context $\hat{\boldsymbol{y}}_{<i}$), *Local Hyperprior Model* (capturing inter-channel dependencies at each spatial location), *Global Hyperprior Model* (capturing global dependencies across the whole image region), and *Parameter Model* (predicting the distribution parameters $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$). By utilizing the predicted $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ by Informer, the quantized latent representation $\hat{\boldsymbol{y}}$ is encoded into a bitstream using an entropy encoder (EC) and the bitstream is decoded by an entropy decoder (ED). In addition to $\hat{\boldsymbol{y}}$, the proposed local hyperprior $\hat{\boldsymbol{z}}_l$ and global hyperprior $\hat{\boldsymbol{z}}_g$ are also encoded using entropy coding. For this, we employ the factorized entropy model [2], where all elements of $\hat{\boldsymbol{z}}_l$ with the same channel index are assumed to follow the same distribution and all elements of $\hat{\boldsymbol{z}}_g$ are assumed to follow different distributions.

Context Model	**Local Hyper Encoder**	**Local Hyper Decoder**	**Global Hyper Encoder**	**Global Hyper Decoder**	**Parameter Model (1)**	**Parameter Model (2)**
Masked: 5×5 c384 s1	Conv: 1×1 c48 s1 Leaky ReLU Conv: 1×1 c12 s1	Conv: 1×1 c98 s1 Leaky ReLU Conv: 1×1 c384 s1	MHA: c192 h4 Linear: c768 GELU Linear: c192 Conv: 1×1 c24 s1	Conv: 1×1 c384 s1	MHA: c384 h8 Linear: c1536 GELU Linear: c384	Conv: 1×1 c640 s1 Leaky ReLU Conv: 1×1 c512 s1 Leaky ReLU Conv: 1×1 c384 s1

Table B.1. Architectural details of Informer when $C = 192$ and $N = 8$. Convolutional layers are represented by the "Conv" prefix followed by the kernel size, number of channels and stride (*e.g.*, the first layer of the local hyper encoder uses 1×1 kernels with 48 channels and a stride of one). The "Masked" prefix means the masked convolutional layer as in [9]. The "MHA" prefix means the multi-head attention blocks as in [5], followed by the number of channels and the number of heads. The "Linear" prefix means the linear layers followed by the number of channels.

C. Rate–distortion curves

Fig. C.1 shows rate–distortion curves corresponding to Fig. 6 in the main paper. For traditional methods, we use JPEG [12], JPEG2000 [11], and BPG [3]. For learned methods, we use those of Minnen *et al*. [8], Lee *et al*. [7], and Qian *et al*. [10].

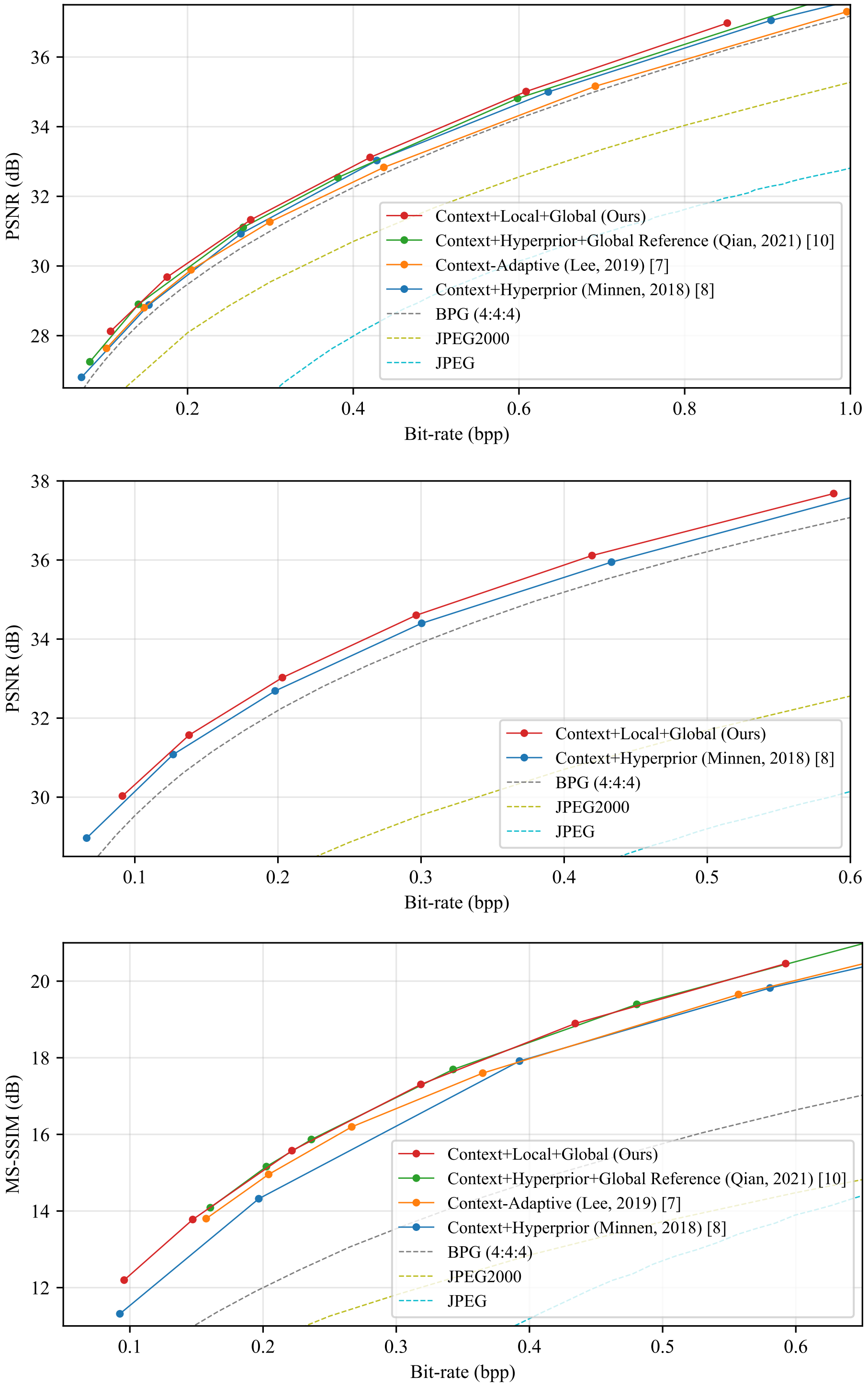
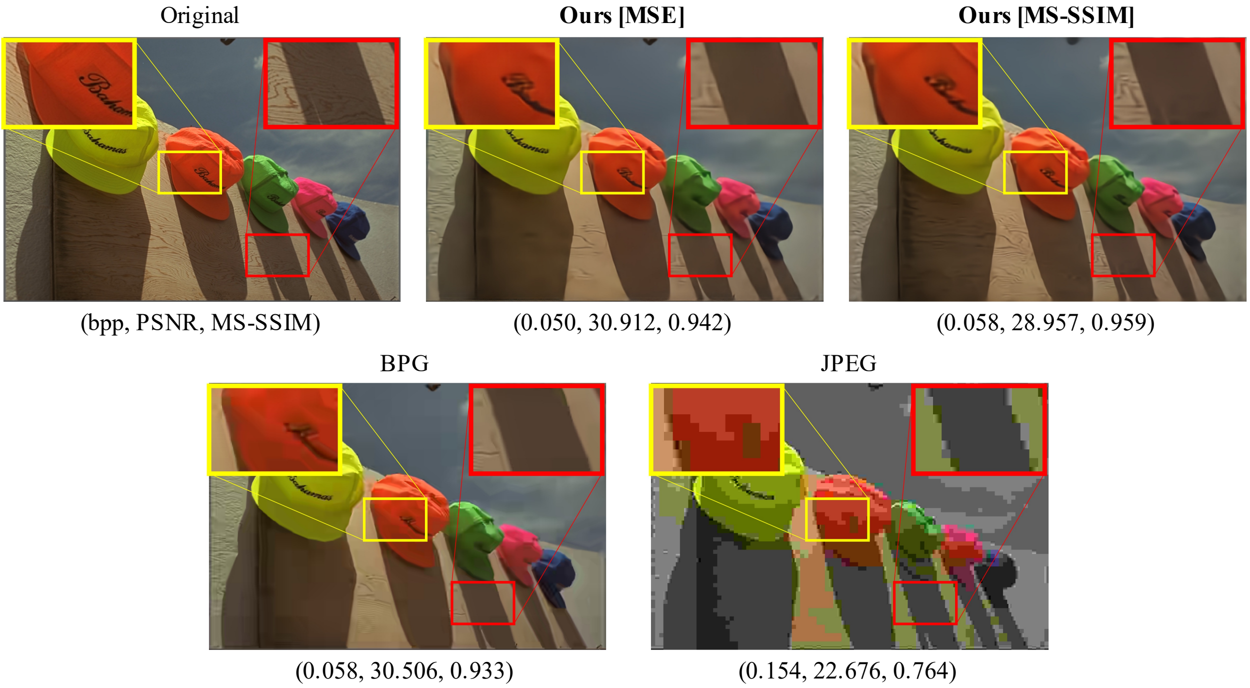


Figure C.1. Rate–distortion curves. Top: MSE-optimized models on Kodak [6]. Middle: MSE-optimized models on Tecnick [1]. Bottom: MS-SSIM-optimized models on Kodak [6].

D. More qualitative results

Figs. D.1 to D.5 show qualitative performance comparison between our method and other traditional image codecs. As in the main paper, for fair comparisons, we encode the images under compression settings for as similar bpp values as possible. Overall, our methods exhibit better qualitative performance by restoring images more clearly, not reconstructing wrong content that does not exist in the original image, and producing patterns more similar to those of the original image.



Original

Ours [MSE]

Ours [MS-SSIM]

(bpp, PSNR, MS-SSIM)

(0.050, 30.912, 0.942)

(0.058, 28.957, 0.959)

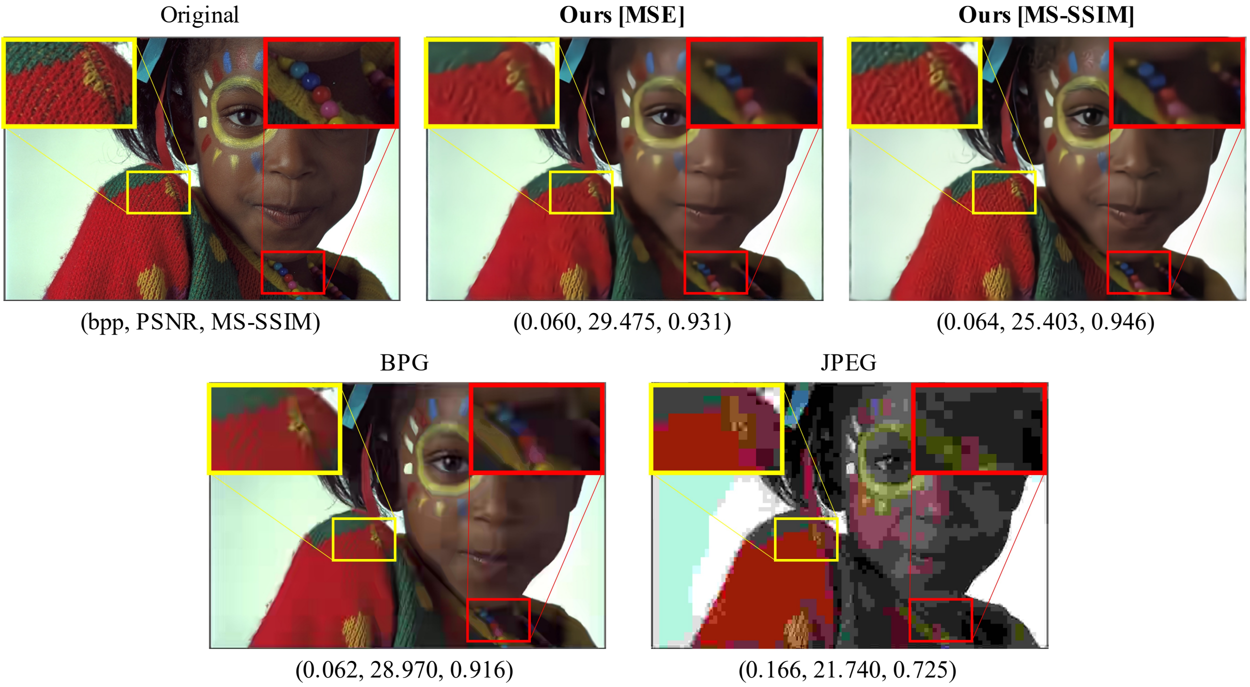
BPG

JPEG

(0.058, 30.506, 0.933)

(0.154, 22.676, 0.764)

Figure D.1. Visual comparison of the decoded images by our methods and the other image codecs on “Kodim03” from Kodak [6].



Original

Ours [MSE]

Ours [MS-SSIM]

(bpp, PSNR, MS-SSIM)

(0.060, 29.475, 0.931)

(0.064, 25.403, 0.946)

BPG

JPEG

(0.062, 28.970, 0.916)

(0.166, 21.740, 0.725)

Figure D.2. Visual comparison of the decoded images by our methods and the other image codecs on “Kodim15” from Kodak [6].

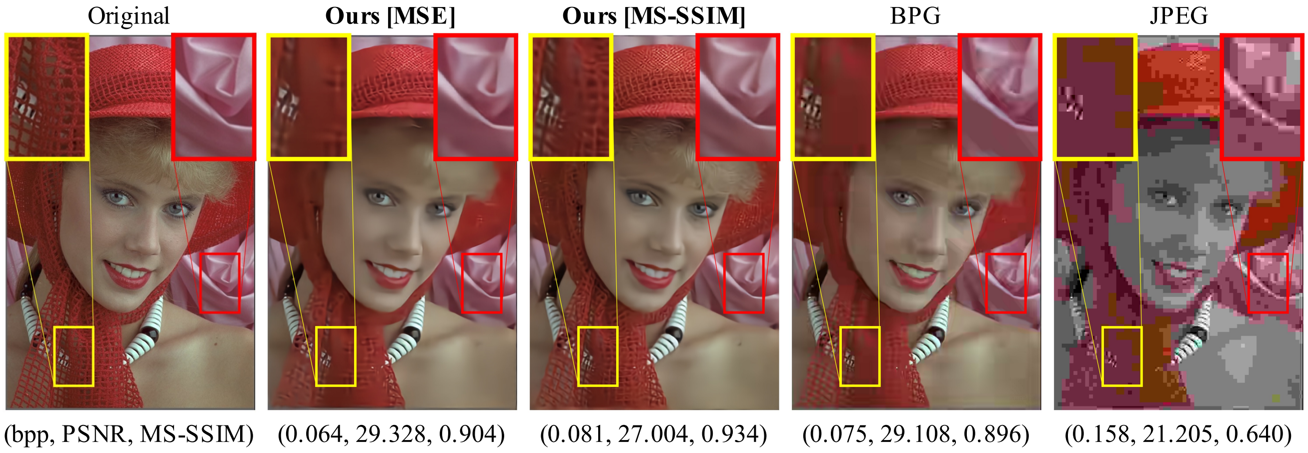


Figure D.3. Visual comparison of the decoded images by our methods and the other image codecs on “Kodim04” from Kodak [6].

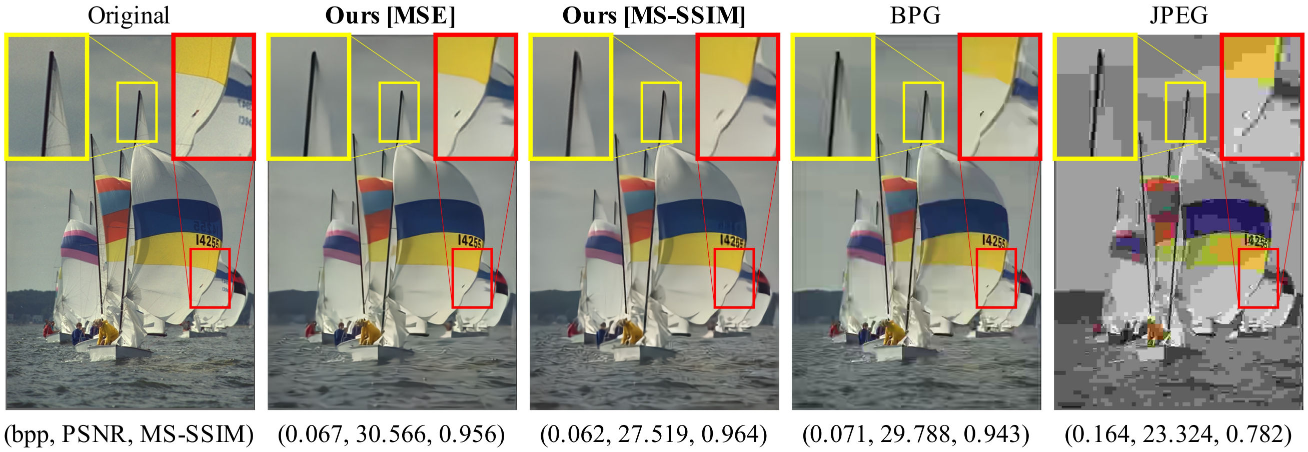


Figure D.4. Visual comparison of the decoded images by our methods and the other image codecs on “Kodim09” from Kodak [6].

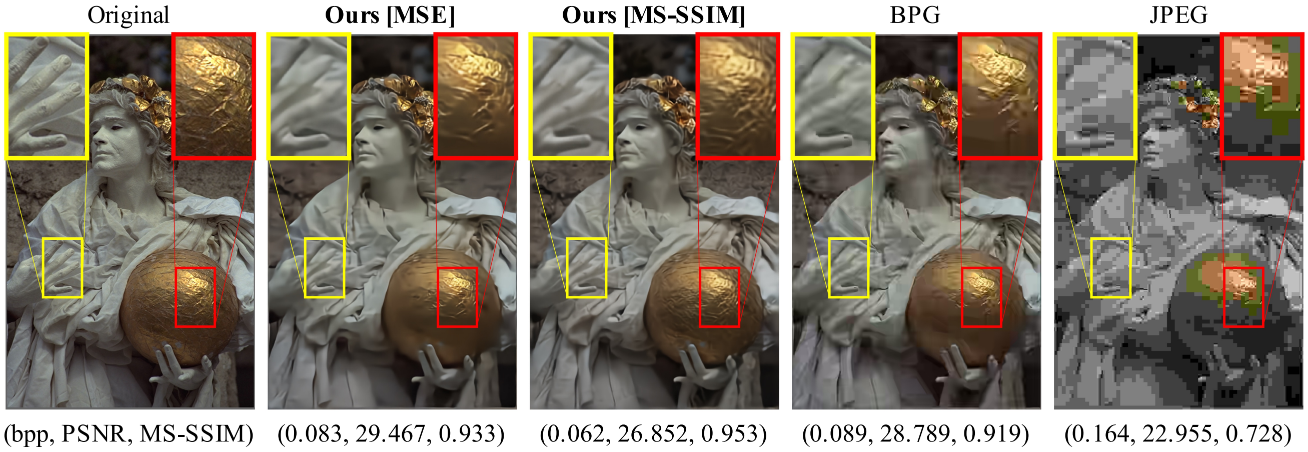


Figure D.5. Visual comparison of the decoded images by our methods and the other image codecs on “Kodim17” from Kodak [6].

References

[1] Nicola Asuni and Andrea Giachetti. TESTIMAGES: a large-scale archive for testing visual devices and basic image processing algorithms. In *Proceedings of the Smart Tools and Apps for Graphics (STAG)*, 2014.

[2] Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. Variational image compression with a scale hyperprior. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.

[3] Fabrice Bellard. BPG image format. http://bellard.org/bpg/.

[4] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto. Learned image compression with discretized gaussian mixture likelihoods and attention modules. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.

[6] The Kodak PhotoCD dataset. http://r0k.us/graphics/kodak/.

[7] Jooyoung Lee, Seunghyun Cho, and Seung-Kwon Beack. Context-adaptive entropy model for end-to-end optimized image compression. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.

[8] David Minnen, Johannes Ballé, and George Toderici. Joint autoregressive and hierarchical priors for learned image compression. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[9] Aäron van den Oord, Nal Kalchbrenner, Oriol Vinyals, Lasse Espeholt, Alex Graves, and Koray Kavukcuoglu. Conditional image generation with pixelcnn decoders. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[10] Yichen Qian, Zhiyu Tan, Xiuyu Sun, Ming Lin, Dongyang Li, Zhenhong Sun, Li Hao, and Rong Jin. Learning accurate entropy model with global reference for image compression. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

[11] Majid Rabbani. JPEG2000: Image compression fundamentals, standards and practice. *Journal of Electronic Imaging*, 11(2):286, 2002.

[12] Gregory K. Wallace. The JPEG still picture compression standard. *IEEE Transactions on Consumer Electronics*, 38(1):xviii–xxxiv, 1992.