EFFICIENT LEARNED IMAGE COMPRESSION WITH SELECTIVE KERNEL RESIDUAL MODULE AND CHANNEL-WISE CAUSAL CONTEXT MODEL

Haisheng Fu[⋆†], *Feng Liang*[⋆], *Jie Liang*[†], *Zhenman Fang*[†] , *Guohe Zhang*[⋆], *Jingning Han*[‡]

[⋆] Xi'an Jiaotong University, China, {fhs4118005070, fengliang, zhangguohe}@xjtu.edu.cn
[†] Simon Fraser University, Canada, {jiel, zhenman}@sfu.ca
[‡]Google LLC, Mountain View, CA, USA, jingning@google.com

ABSTRACT

Recently, learning-based image compression approaches have achieved superior performance over classical image compression methods. However, their complexities remain quite high. In this paper, we propose two efficient modules to reduce the complexity. First, we introduce a selective kernel residual module into the core network, which effectively expands the receptive field and captures global information. Second, we present an improved channel-wise causal context model, designed to not only reduce encoding and decoding time but also ensure rate-distortion performance. Experimental results demonstrate that our proposed method achieves better trade-off than recent leading learned image compression methods, and also outperforms the latest H.266/VVC (4:4:4) in terms of PSNR and MS-SSIM metrics.

Index Terms— Learning-based image compression, Selective kernel residual module, Channel-wise causal context model

1. INTRODUCTION

Image compression is a fundamental technology for many multimedia applications. The traditional image compression standards, including JPEG, JPEG 2000, BPG (intra-frame encoding of HEVC/H.265) [1], and the intra coding of VVC/H.266, include the following key components: linear transforms such as Discrete Cosine Transform (DCT) and wavelet transform, quantization, and entropy coding.

Recently, deep-learning-based image compression methods have begun to outperform traditional approaches, including the latest VVC intra coding. Most leading end-to-end learned image compression schemes follow a similar pipeline to traditional approaches: a core network that extracts low-dimensional latent representations from the input image, quantization, and a hyper coding network for entropy coding.

In the core network part, different modules are developed to obtain a more efficient and compact latent representation, such as generalized divisive normalization (GDN) [2], non-local attention module [3], residual block [4, 5], generative adversarial network (GAN) [6], and transformer networks [7, 8].

For entropy coding, the hyperprior approach was first introduced in [9], using a zero-mean Gaussian scale model. The Gaussian mixture model (GMM) for the hyperprior is proposed in [10]. The GMM is also used in [11, 12, 13].

In [14], the Gaussian-Laplacian-Logistic mixture model (GLLMM) is proposed to replace the GMM to further reduce redundancy in latent representations. A concatenated residual block (CRB) is also developed to improve rate-distortion performance. The scheme in [14] outperforms other learning-based approaches and VVC intra coding (4:4:4) in PSNR and MS-SSIM.

However, the complexity of the approach presented in [14] is quite high. In this paper, we propose two efficient modules. First, we introduce a selective kernel residual module (SKRM) into the core network, offering lower complexity compared to the previous concatenated residual blocks (CRB) in [14]. Second, we employ an improved channel-wise causal context model (CWCCM), which splits the latent representation into two parts, with each being encoded and decoded separately. This not only preserves rate-distortion performance but also reduces encoding/decoding time. To further reduce complexity, we replace the more involved GLLMM model with the simpler GMM model for entropy coding.

Thanks to the contributions of SKRB and CWCCM, our method achieves a better trade-off between complexity and performance compared to [14]. The BD-Rate performance of the proposed method is about 2% worse than [14], but it still outperforms other leading learning-based methods by more than 5%. On the other hand, the decoding speed of the proposed method is approximately 27 times faster than the GLLMM method [14], and also faster than other learning-based methods. Experimental results in this paper also demonstrate that our method achieves a better performance than VVC (4:4:4).

2. THE PROPOSED IMAGE COMPRESSION FRAMEWORK

Fig. 1 depicts the proposed framework. Similar to [10, 11, 14], our framework follows the same VAE structure [9], which consists of two sub-networks: the core subnetworks and the hyper subnetworks.

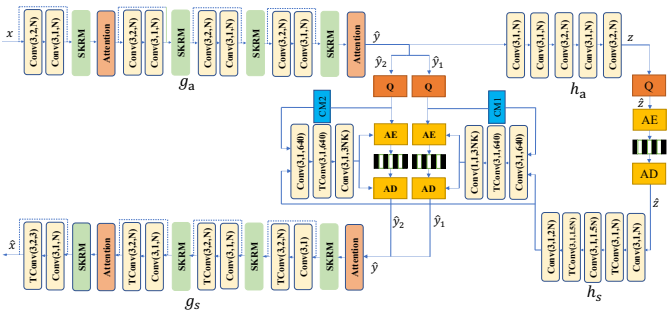

Fig. 1. The network architecture of the proposed method. Convolution **Conv(k,s,n)** and its transposed version **TConv(k,s,n)** use the kernel with a size of k × k and a stride of s, and N is the number filters. AE and AD represent arithmetic encoder and arithmetic decoder, respectively. Similar to [11, 14], the dotted lines represent shortcut connections with size change. Both $CM1$ and $CM2$ utilize the same context model as in [11], which are implemented by a 5 ×5 mask convolution.

In the core encoder network, three stages of selective kernel residual blocks (SKRB) are used, which will be described in Sec. 2.2. As in [14, 11], our decoder network is symmetric to the encoder, with down-sampling operations being replaced by up-sampling operations.

To improve encoding and decoding efficiency, the hyper networks were developed to learn the distribution of the latent representation y. Unlike [11, 14], the latent representation is uniformly divided into two parts along the channel direction, denoted as y_1 and y_2, and the hyper networks are used to learn the probability parameters of these two parts separately. Unlike [15], there is no information interaction between y_1 and y_2, i.e., these two parts can be encoded and decoded independently. To reduce the complexity, we use the GMM model to estimate the probability distribution of y_1 and y_2.

In the ablation experiments in Sec. 4, we will conduct extensive experiments to demonstrate the effectiveness of the proposed modules.

2.1. Selective Kernel Residual Module

In [16], a Selective Kernel (SK) unit was designed for object recognition by adaptive kernel selection in a soft-attention manner, as shown in Fig. 2.2. It was motivated by the adaptive receptive field (RF) sizes of neurons in visual cortex, where convolutions at different scales are fused to extract a more efficient feature representation of the input information by using softmax attention.

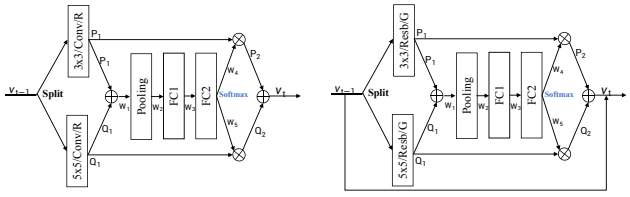

Fig. 2. (a) The selective kernel unit in [16]. (b) The proposed improved selective kernel residual module. G represents GDN operator. k is the convolution kernel size.

The structure in [16] is shown in Fig. 2(a). Given an input feature map $V_{t-1} \in R \times H \times C$, two branches, namely features P_1 and Q_1, can be obtained by utilizing two convolutional layers with kernels of 3×3 and 5×5. Both P_1 and Q_1 use ReLU as the activation function. Next, we fuse P_1 and Q_1 to obtain the fused feature w_1 through element-wise summation. To aggregate global information, a global average pooling layer is used to obtain channel-wise statistics w_2. Subsequently, a fully connected layer $FC1$ is employed to capture the relationships between features of different scales and to adaptively select kernels. Additionally, a fully connected layer $FC2$ is used to reduce the complexity of w_3, with the number of channels in w_3 being 1/16 of w_2. Two softmax layers are then employed to learn the weights of the two scales, denoted as w_4 and w_5. Finally, we obtain the final feature v_t, which contains information from multiple scales.

Based on [16], we propose an improved SKRM and apply it to image compression, as illustrated in Fig. 2(b). In comparison to 2(a), we have made the following three improvements. Firstly, the SK in [16] only uses simple convolutions. In our SKRM, we replace it with the residual block from ResNet [4], which makes it easier to acquire global information at different scales. Secondly, we have added a shortcut between the input v_{t-1} and the final output v_t, which facilitates the convergence of the network. Thirdly, we have also incorporated the GDN operator to enhance feature extraction.

2.2. Channel-wise Causal Context Module

Earlier learning-based image compression approaches use causal context models to improve rate-distortion performance. However, this approach requires sequential decoding, which is very slow. To address this issue, in [15], a channel-wise autoregressive entropy model is proposed to minimize element-level serial processing in the context model. However, the R-D performance drops by 0.2-0.3 dB compared with the causal context models on the Kodak dataset.

We propose an improved channel-wise causal context model (CWCCM) by combining the channel-wise autoregres-

sive entropy model with the serial causal context-adaptive model. This approach achieves a better trade-off between complexity and performance. As shown in Fig. 1, the latent representation y is evenly split into y_1 and y_2 along the channel direction. As in [11], we use the GMM model to estimate the probability distribution of y_1 and y_2. Compared with GLLMM, GMM saves considerable encoding and decoding time. We also find that the channels of y_1 and y_2 become increasingly sparse, allowing us to skip the all-zero channels when encoding and decoding these latent representations.

3. THE LOSS FUNCTION

When training our end-to-end learned image compression network architectures, we need to jointly optimize two terms: the bitrate R (representing the number of bits in the bitstream) and distortion D (such as Mean Squared Error or MS-SSIM [17]), which measures the discrepancy between the origin image and the reconstructed image. The trade-off between rate and distortion is determined by the Lagrange multiplier λ. The loss function of our framework is defined as follows:

$$\begin{aligned} L &= R + \lambda D, \\ D &= E_{x \sim P_x}[d(x, \hat{x})], \end{aligned} \tag{1}$$

where p_x corresponds to the unknown distribution of the input images.

Table 1. Comparisons of encoding and decoding time and model sizes.

Method	Encode	Decode	Model(L)	Model(H)
VVC	402.3 s	0.6 s	7.2 MB	7.2 MB
[18]	10.7 s	37.9 s	123.8 MB	292.6 MB
[3]	402.3 s	2405.1 s	84.6 MB	290.9 MB
[11]	20.9 s	22.1 s	50.8 MB	175.2 MB
[14]	467.90 s	468.0 s	77.08 MB	241.0 MB
Ours	**15.5 s**	**17.4 s**	**60.9 MB**	**206.3 MB**

4. EXPERIMENT

4.1. Training Details

The networks are trained on color PNG images collected from the CLIC dataset [1] and the LIU4K dataset [19].

Models are trained for various bit rates. When optimizing for the PSNR metric, λ is set to the elements of 0.0016, 0.0032, 0.0075, 0.015, 0.03, 0.045, 0.06. The number of channels, denoted by N, is set to 128 for the first three elements and 256 for the last four elements. When optimizing for MS-SSIM, λ is set to 12, 40, 80, and 120. N is set to 128 for the first two cases, and 256 for the other two cases. Each model is trained for 1.5×10^6 iterations. We use the Adam solver with a batch size of 8. The learning rate is set to 1×10^{-5} during training.

[1] http://www.compression.cc/

4.2. Comparisons

We compare our method with some recent leading learned image compression schemes as well as traditional image codecs in terms of PSNR and MS-SSIM metrics. MS-SSIM is measured in decibels (dB) using the formula $-10\log_{10}(1-MS\text{-}SSIM)$. The learned schemes included GLLMM [14], Chen2021 [3], Cheng2020 [11], and Lee2019 [18]. The classical image codecs are comprised of the latest H.266/VVC-Intra (4:4:4) (VTM 8.0) [2] and BPG (H.265/HEVC-Intra).

The average rate-distortion curves for the 24 Kodak images are depicted in Fig. 3. When optimized for PSNR, GLLMM achieves the best performance. Our approach achieves very similar performance to GLLMM across a wide bit rates. Compared to VVC (4:4:4), our method has similar performance at low bit rates, but better performance at high bit rates. our method also outperforms other learning-based methods, including Chen2021 [3] and Cheng2020 [11], by up to 0.5 dB at higher bit rates. When optimized for MS-SSIM, GLLMM still achieves the best results. Our method is slightly worse than GLLMM, but it still outperforms other compared approaches.

4.3. Encoding and Decoding Complexity

Table 1 compares the complexities of different methods. Since some methods can only run on CPU, we evaluate the running times of all methods on a 2.9GHz Intel Xeon Gold 6226R CPU. The average time across all Kodak images is used. We also report the average model size at both low and high bit rates.

From Table 1, it can be seen that compared to the state-of-the-art GLLMM [14], the proposed method is approximately 27 times faster in both encoding and decoding. Our model size is also smaller than the GLLMM method and is comparable to Cheng2020. Compared to VVC, our encoding speed is about 25 times faster, but the decoding speed is much slower. From the results from Table 1 and Fig. 3, we could obtain the following conclusion. Our scheme achieves an improved tradeoff between coding performance and decoding speed. Since we split the channel into two parts, each of which is modeled with a separate causal entropy model, and the two parts can be processed in parallel. This not only reduces the decoding time but also ensures performance.

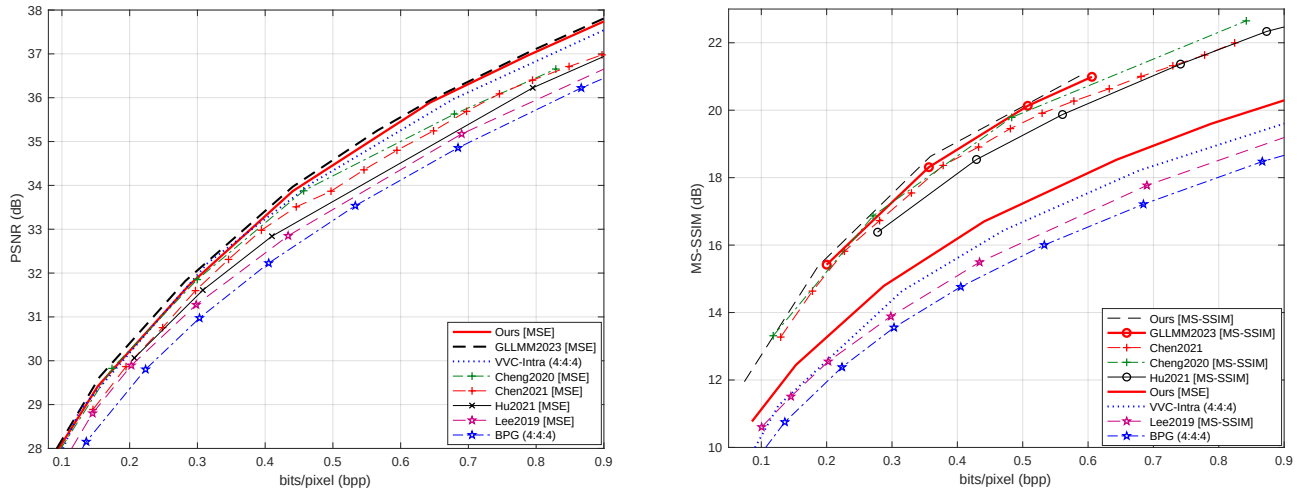

Fig. 3. The average PSNR and MS-SSIM performance of the 24 images in the Kodak dataset.

Table 2. The performance of different modules

Module	Bit rate	PSNR	MS-SSIM
Ours	0.1542	29.40 dB	12.45 dB
w/o CWCCM	0.1579	29.32 dB	12.40 dB
w/o SKRM	0.1647	29.20 dB	12.36 dB
Ours	0.6434	35.89 dB	18.53 dB
w/o CWCCM	0.6547	35.75 dB	18.48 dB
w/o SKRM	0.6589	35.70 dB	18.39 dB

Table 3. The performance of different groups

Groups	Bit rate	PSNR	MS-SSIM	model Size
2	0.1542	29.40 dB	12.45 dB	60.9 MB
4	0.1621	29.32 dB	12.40 dB	62.3 MB
2	0.6434	35.89 dB	18.53 dB	206.3 MB
4	0.6542	35.76 dB	18.48 dB	208.7 MB

4.4. Performance Improvement of Different Modules

Table 2 compares the results when the CWCCM and SKRM modules are removed from our network architecture respectively, and the other modules remain the same. It can be seen that the performance drops by about 0.1 dB without CWCCM. The performance drops by about 0.2 dB at both low and high bit rates without SKRM. The reason is as follows. Since our proposed SKRM can expand the receptive field and capture global information at different scales, it could extract more compact and efficient latent representations, thus improving rate-distortion performance. The proposed CWCCM can split the latent representation into two parts. Each part utilizes a causal entropy model, which not only makes the latent representation sparser but also reduces the spatial redundancy.

4.5. Performance Comparison of Different Group Partitions

We also explore the case of dividing the latent representations evenly into two and four groups, respectively, and the results are shown in Table 3. It can be seen that the performance drops about 0.1 dB when the latent representations are divided into four groups. The model size also increases slightly. We cannot divide the latent presentation into too many parts. The reason is that when the channels are divided into too many groups, there are fewer channels in each group, making it difficult to remove spatial redundancy between pixels.

[2]https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM/tree/VTM-5.2

5. CONCLUSION

In this paper, we propose two efficient modules to reduce the complexity of the state-of-the-art learned image coding scheme in [14], namely the selective Kernel residual module (SKRM) and the channel-wise causal context model (CWCCM). We also use the simple Gaussian mixture model in entropy coding, instead of the more complicated GLLMM model. Experimental results show that the encoding and decoding of our method are about 26 times faster than [14]. Although there is a slight drop in the rate-distortion performance, it still outperforms H.266/VVC (4:4:4) and other compared learning-based methods.

6. ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (No. 61474093), Industrial Field Project - Key Industrial Innovation Chain (Group) of Shaanxi Province (2022ZDLGY06-02), the Natural Sciences and Engineering Research Council of Canada (RGPIN-2020-04525), Google Chrome University Research Program, NSERC Discovery Grant RGPIN-2019-04613, DGECR-2019-00120, Alliance Grant ALLRP-552042-2020; CFI John R. Evans Leaders Fund.

7. REFERENCES

[1] Gary J. Sullivan, Jens-Rainer Ohm, Woo-Jin Han, and Thomas Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, 2012.

[2] Johannes Ballé, Valero Laparra, and Eero P. Simoncelli, "End-to-end optimization of nonlinear transform codes for perceptual quality," in *2016 Picture Coding Symposium (PCS)*, 2016, pp. 1–5.

[3] Tong Chen, Haojie Liu, Zhan Ma, Qiu Shen, Xun Cao, and Yao Wang, "End-to-end learnt image compression via non-local attention optimization and improved context modeling," *IEEE Transactions on Image Processing*, vol. 30, pp. 3179–3191, 2021.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of CVPR*, June 2016, pp. 770–778.

[5] Haisheng Fu, Feng Liang, Bo Lei, Nai Bian, Qian Zhang, Mohammad Akbari, Jie Liang, and Chengjie Tu, "Improved hybrid layered image compression using deep learning and traditional codecs," *Signal Processing: Image Communication*, vol. 82, pp. 115774, 2020.

[6] M. Akbari, J. Liang, and J. Han, "Dsslic: Deep semantic segmentation-based layered image compression," in *The 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2019, pp. 2042–2046.

[7] Y. Zhu, Y. Yang, and T. Cohen, "Transformer-based transform coding," in *International Conference on Learning Representations*, 2022.

[8] Renjie Zou, Chunfeng Song, and Zhaoxiang Zhang, "The devil is in the details: Window-based attention for image compression," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 17492–17501.

[9] J. Ballé, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston, "Variational image compression with a scale hyperprior," in *International Conference on Learning Representations*, 2018, pp. 1–23.

[10] David Minnen, Johannes Ballé, and George D Toderici, "Joint autoregressive and hierarchical priors for learned image compression," in *Advances in Neural Information Processing Systems*, 2018, pp. 10794–10803.

[11] Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, "Learned image compression with discretized gaussian mixture likelihoods and attention modules," in *Proceedings of CVPR*, 2020, pp. 7939–7948.

[12] J. Lee, S. Cho, and M. Kim, "Joint autoregressive and hierarchical priors for learned image compression," *arXiv:1912.12817*, 2020.

[13] Haisheng Fu, Feng Liang, Jie Liang, Binglin Li, Guohe Zhang, and Jingning Han, "Asymmetric learned image compression with multi-scale residual block, importance scaling, and post-quantization filtering," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 8, pp. 4309–4321, 2023.

[14] Haisheng Fu, Feng Liang, Jianping Lin, Bing Li, Mohammad Akbari, Jie Liang, Guohe Zhang, Dong Liu, Chengjie Tu, and Jingning Han, "Learned image compression with gaussian-laplacian-logistic mixture model and concatenated residual modules," *IEEE Transactions on Image Processing*, vol. 32, pp. 2063–2076, 2023.

[15] David Minnen and Saurabh Singh, "Channel-wise autoregressive entropy models for learned image compression," in *2020 IEEE International Conference on Image Processing (ICIP)*, 2020, pp. 3339–3343.

[16] Xiang Li, Wenhai Wang, Xiaolin Hu, and Jian Yang, "Selective kernel networks," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 510–519.

[17] Z. Wang, E.P. Simoncelli, and A.C. Bovik, "Multiscale structural similarity for image quality assessment," in *The Thrity-Seventh Asilomar Conference on Signals, Systems Computers, 2003*, 2003, vol. 2, pp. 1398–1402.

[18] Jooyoung Lee, Seunghyun Cho, and Seung-Kwon Beack, "Context-adaptive entropy model for end-to-end optimized image compression," in *International Conference on Learning Representations*, 2019.

[19] Jiaying Liu, Dong Liu, Wenhan Yang, Sifeng Xia, Xiaoshuai Zhang, and Yuanying Dai, "A comprehensive benchmark for single image compression artifact reduction," *IEEE Transactions on Image Processing*, vol. 29, pp. 7845–7860, 2020.