HIERARCHICAL FEATURE AGGREGATION NETWORK FOR DEEP IMAGE COMPRESSION

Wenfeng Li[⋆†], *Zongcai Du*[⋆†], *Hao He*[⋆], *Jie Tang*[⋆‡], *Gangshan Wu*[⋆]

[⋆]State Key Laboratory for Novel Software Technology, Nanjing University, China
{wenfengli, 151220022, haohe}@smail.nju.edu.cn, {tangjie, gswu}@nju.edu.cn

ABSTRACT

Existing CNN-based methods for image compression extract features through serially connected high-to-low (encoder) or low-to-high (decoder) resolution stages, leading to insufficient utilization of hierarchical features. To solve this problem, we present a hierarchical feature aggregation network (HFAN) for generating more informative latent representations. In detail, we propose two strategies, namely inter-stage feature aggregation and intra-stage feature aggregation. The inter-stage feature aggregation integrates multi-scale information thereby producing more contextual features. The intra-stage aggregation fuses features within the same stage to enrich representations of one specific resolution. Besides, we incorporate a lightweight pixel-wise attention mechanism to further enhance the discriminative ability of our network. Extensive experiments demonstrate that our HFAN achieves superior performance over state-of-the-art methods without a hyperprior variational autoencoder.

Index Terms— deep image compression, feature aggregation, attention mechanism

1. INTRODUCTION

A deep image compression system generally includes four parts, e.g., encoder, decoder, quantizer and entropy model. The performance can be improved by carefully considering the four parts. Some works focus on the quantization [1, 2, 3], where the concern is how to design differentiable quantization. Some works such as [4, 5, 6, 7] are dedicated to the entropy estimation of quantized latent representations to achieve best trade-off between reconstruction errors and required bits. Some works pay attention to the architecture design in order to generate compact representation. For example, The work [8] proposed a three-layer convolutional network for encoder and decoder with generalized divisive normalizaion (GDN) for activation. The work [9] pursued variant recurrent neural networks to compress the residuals recursively. The work [10] adopted generative adversarial network (GAN) to realize an extreme image compression system, while obtaining better subjective reconstructions at the same time. Recently, residual learning and attention mechanism are further introduced to image compression [11], for easing the training difficulty and emphasizing on important features.

Although these methods have made great progress, we notice that interests of the en/decoder design are shifted to creating powerful feature extraction blocks such as residual blocks [11], attention-based blocks [12], ignoring to utilize the intermediate information of different stages, which leads to performance decrease. Typical en/decoder of deep image compression involves several high-to-low or low-to-high resolution stages, acting as a coarse-to-fine manner. Each stage performs deep feature extraction at a specific resolution, and every intermediate feature contributes to the final compressed or reconstructed representation. Motivated by this, we propose two strategies, called inter-stage feature aggregation and intra-stage feature aggregation, to make full use of the hierarchical features to produce more powerful representations. The inter-stage aggregation integrates multi-scale information to produce more contextual features and the intra-stage feature aggregation fuses features of the same resolution to prepare more powerful features for next stage. Furthermore, we decompose pixel-wise attention mechanism into channel-wise and spatial-wise attention mechanism to boost the performance. It is worth mentioning that the proposed feature aggregation strategies and the attention mechanism are lightweight thus can be easily incorporated into other en/decoder architectures.

The rest of the paper is organized as follows. Section 2 introduces the related work. Section 3 describes the proposed HFAN. Experiments are explained in Section 4. We conclude with a brief summary in Section 5.

2. RELATED WORK

2.1. Hand-crafted Image Compression

Conventional image compression standards such as JPEG[13], JPEG2000[14] and HEVC/H.265[15] rely on hand-crafted module design individually. For instance, these modules include intra prediction, discrete cosine transform or wavelet transform, quantization and entropy coder such as Huffman coder or content adaptive binary arithmetic coder (CABAC). Multiple modes are considered in these methods and the rate-distortion optimization is used to determine the best mode. These compression methods are robust to different images but may produce visually unpleasant reconstructions with artifacts like blocking and ringing.

2.2. Deep Image Compression

Existing CNN-based methods such as [1, 8, 3, 11] tend to make end-to-end training possible by creating differential quantization and dealing with rate estimation. The work [11] further proposes a channel-level variable quantization network to dynamically allocate more bitrates for significant channels and withdraw bitrates for negligible channels. In addition, the architecture design of encoder and decoder are paid more and more attention because the latent representation generated by encoder has a direct influence on the quantization and entropy model, which largely determines the reconstruction quality. For example, Some works [16, 17] adopts recurrent neural networks to compress the images, but they relied heavily on binary representation to achieve scalable coding. Some approaches [10, 18, 19] uses generative adversarial network to obtain realistic reconstructions at a extremely low bit rate. The work [20]

[†] Equal Contribution
[‡] Contact Author

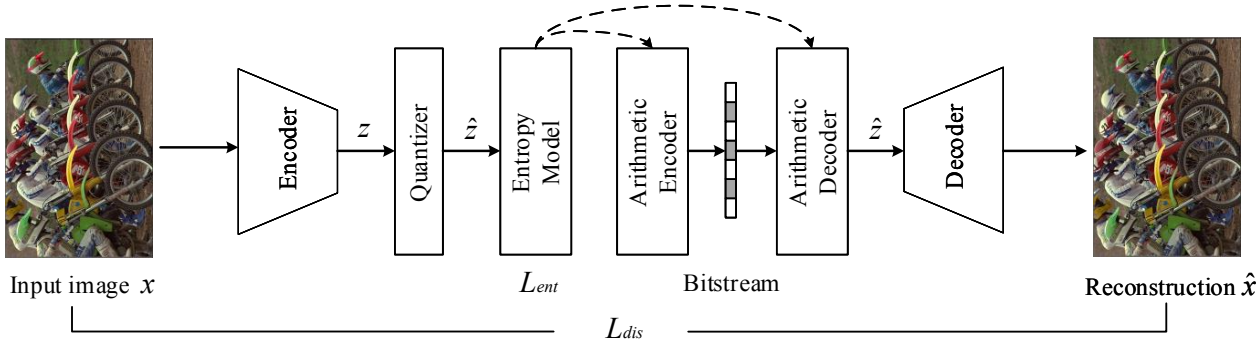


Fig. 1. The overall architecture of our proposed image compression system.

applys principle component analysis to include a content-weighted strategy or de-correlating different channels. The work [21] introduces residual [22] units to train deeper network and the work [12] incorporates attention mechanism to make the network more discriminative. Recently, some works [4, 5, 6, 7] reveal that entropy estimation has also a large impact on the final performance and propose hyperprior model and joint model, which are the most representative methods and have achieved best tradeoff between reconstruction errors and required bits. Although these methods have made great progress, they ignore to utilize the intermediate information of different resolutions, which leads to performance decrease.

3. PROPOSED METHOD

3.1. Overall Architecture

As shown in Fig. 1, our image compression system comprises four components which are encoder, quantizer, entropy model and decoder. We map an image $\boldsymbol{x}$ to a latent representation $\boldsymbol{z}$ with encoder and then discretize $\boldsymbol{z}$ to $\hat{\boldsymbol{z}}$ with quantizer. After quantization, we use the entropy model to generate the conditional probability of $\hat{\boldsymbol{z}}$ and the decoder to reconstruct the image $\hat{\boldsymbol{x}}$ from the quantized latent representation $\hat{\boldsymbol{z}}$. The details of these parts are illustrated in the following subsections, respectively.

3.2. En/Decoder

The encoder of our proposed method are composed of three parts: head, body, and tail. The head module which contains one convolutional layer and a residual group (RG) as depicted in Fig. 3(c) extracts initial feature maps from the original image. Each RG is a stack of residual pixel attention block (RPAB) which is depicted in Fig. 3. The tail module which also contains one convolutional layer generates the compressed latent representation $\hat{\boldsymbol{z}}$ with C channels, where C can be manually varied for different bit per pixels (BPPs). As for the body module, previous deep learning-based approaches such as [11, 12] merely extract features through serially connected high-to-low resolution residual stages as shown in Fig. 2(a), consequently leading to insufficient utilization of the intermediate features. To solve this issue, we propose a hierarchical feature aggregation network (HFAN) to help generate more informative and contextual latent representations. In detail, we propose two strategies, namely intra-stage feature aggregation and inter-stage aggregation, to achieve this goal. Besides, we incorporate a lightweight pixel-wise attention mechanism by decomposing it into channel-wise and spatial-wise attention mechanism, which can better enhance the discriminative ability of the network. Similarly, the architecture of the decoder is simply the inverse version of the encoder.

3.2.1. Intra-stage Feature Aggregation

In order to utilize the informative intermediate features more efficiently, we propose intra-stage feature aggregation strategy. The principle behind intra-stage feature aggregation is that features at various levels gradually focus on different aspects of the image and they can efficiently complement each other. Similar ideas are also investigated in some single image super resolution networks [23, 24]. Specifically, as shown in Fig. 2(c), we concatenate the output feature maps of all the RG within the same stage. After concatenation, the feature maps are passed through two convolutional layers with ReLU as activation function. Finally, a skip connection is adopted as the vanilla residual structure does, which can ease the training difficulty. The effectiveness will be experimentally verified in Sec 4.2.

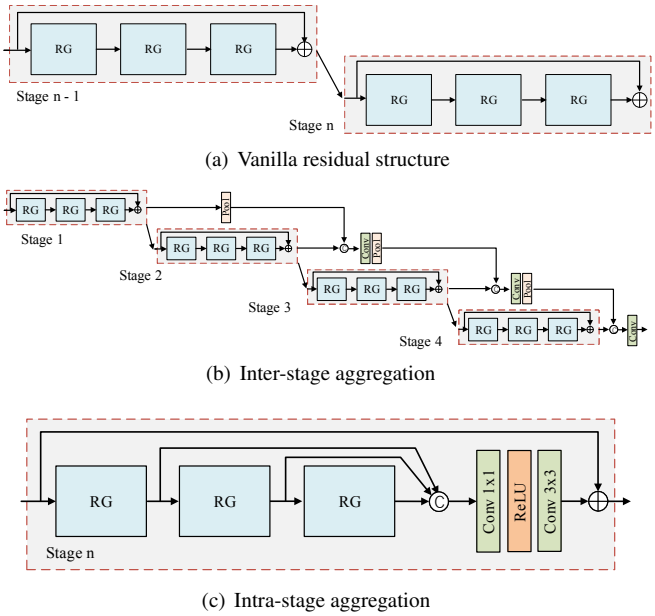
3.2.2. Inter-stage Feature Aggregation

To push multi-scale information fusion and generate more contextual features, we propose a inter-stage feature aggregation strategy. Since low-resolution stage always lacks details and high-resolution stage always lacks large receptive field, we consider their combination to overcome their weaknesses. As shown in Fig. 2(b), for the middle stages, we sample the feature map of the former stage and concatenate it with the output feature map of the current stage. Then we pass the concatenated feature map through a 1×1 convolutional layer to reduce the channel number. For the last stage, we output the feature map without downsampling. The effectiveness will also be experimentally verified in Sec 4.2.

3.2.3. Pixel-wise Attention Module

To further enhance the discriminative ability of our encoder and decoder network, we incorporate a lightweight pixel-wise attention (PA) module which is shown in Fig. 3(a).

Consider the input feature of PA module is $\boldsymbol{X} \in \mathbb{R}^{C\times H\times W}$, we use PA module to generate attention maps $\boldsymbol{X}_{att} \in \mathbb{R}^{C\times H\times W}$. For channel attention module, we use global spatial average pooling layer and spatial max pooing layer to get the overall and max response of each channel simultaneously. Then two convolutional layers are applied to generate the channel attention maps $\boldsymbol{X}_{c,a}, \boldsymbol{X}_{c,m} \in \mathbb{R}^{C\times 1\times 1}$. And the final channel attention maps $\boldsymbol{X}_c \in \mathbb{R}^{C\times 1\times 1}$ is formulated as:

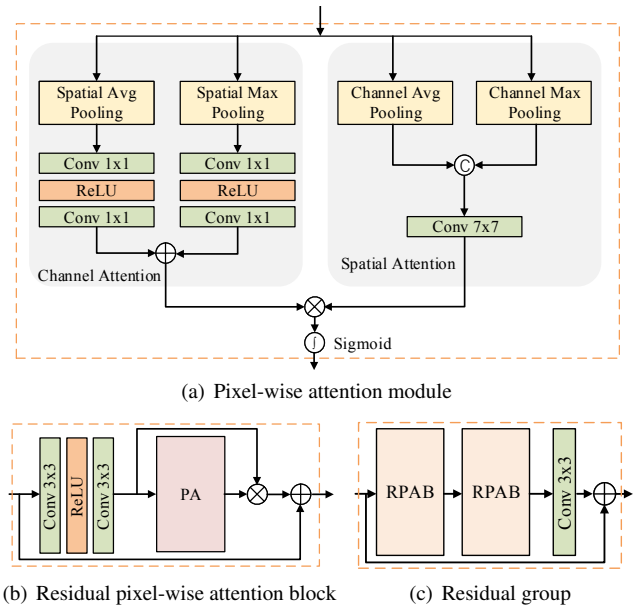


(a) Vanilla residual structure

(b) Inter-stage aggregation

(c) Intra-stage aggregation

Fig. 2. (a) Vanilla residual structure. (b) Inter-stage feature aggregation strategy. (c) Intra-stage feature aggregation strategy. *Best viewed on screen.*



(a) Pixel-wise attention module

(b) Residual pixel-wise attention block

(c) Residual group

Fig. 3. (a) Proposed pixel-wise attention module (PA). (b) Residual pixel-wise attention block (RPAB). (c) Residual group (RG) built with RPABs.

$$\begin{aligned} \boldsymbol{X}_{c,a} &= \boldsymbol{W}_s^2 \times ReLU(\boldsymbol{W}_s^1 \times \boldsymbol{X}_a^C), \\ \boldsymbol{X}_{c,m} &= \boldsymbol{W}_s^2 \times ReLU(\boldsymbol{W}_s^1 \times \boldsymbol{X}_m^C), \\ \boldsymbol{X}_c &= \boldsymbol{X}_{c,a} + \boldsymbol{X}_{c,m}, \end{aligned} \tag{1}$$

where $\boldsymbol{X}_a^C, \boldsymbol{X}_m^C \in \mathbb{R}^C$ are results of $\boldsymbol{X}$ after global spatial average pooling layer and spatial max pooling layer, respectively. $\boldsymbol{W}_s^1$ and $\boldsymbol{W}_s^2$ denote the weights of the first and second convolutional layer. $\times$ stands for matrix multiplication. Note that we share the two convolutional layers in the two branches of channel attention module.

For the spatial attention module, we introduce global cross-channel average pooling and max pooling layers to get the overall and max response in each spatial position. 7×7 kernel size is adopted to grasp spatial information within large receptive field. Then the convolutional layer is applied to generate the spatial attention maps $\boldsymbol{X}_s \in \mathbb{R}^{1\times H\times W}$. It is formulated as:

$$\boldsymbol{X}_s = \boldsymbol{W}_c \times concat(\boldsymbol{X}_a^{H,W}, \boldsymbol{X}_m^{H,W}), \tag{2}$$

where $\boldsymbol{X}_a^{H,W}, \boldsymbol{X}_m^{H,W} \in \mathbb{R}^{H\times W}$ are results of $\boldsymbol{X}$ after global cross-channel average pooling and max pooling layer, respectively.

Finally, the pixel-wise attention maps $\boldsymbol{X}_{att}$ is the product of the spatial attention maps and channel attention maps followed by a $sigmoid$ operation. It is formulated as:

$$\boldsymbol{X}_{att} = sigmoid(\boldsymbol{X}_s \odot \boldsymbol{X}_c). \tag{3}$$

It is worth noting that either our channel attention or spatial attention is lightweight and only introduce negligible overheads, and the multiplication between them brings nearly the same memory cost (e.g., only use channel attention), so our pixel-wise can be easily incorporated into other architectures. Based on proposed PA module, we adapt the widely used residual block to residual pixel-wise attention block (RPAB) which is shown in Fig. 3(b). Besides, we stack two RPABs followed by a convolutional layer and a skip connection which is shown in Fig. 3(c) to construct a residual group (RG).

3.3. Quantization and Entropy Model

We adopt a GMM-based quantizer proposed in the work of [11] to quantize $\boldsymbol{z}$. Specifically, we divide the feature maps $\boldsymbol{z}$ into G groups by channel and allocate different bitrates to different groups. We assume the prior distribution $p(\boldsymbol{z})$ is a mixture of Gaussian distributions:

$$p(\boldsymbol{z}) = \prod_i \sum_{q=1}^{Q} \pi_q \mathcal{N}(z_i|\mu_q, \sigma_q^2), \tag{4}$$

where π_q, μ_q, and σ_q are the learnable mixture parameters and Q is the quantization level.

We obtain the forward quantization result by setting it to the mean that takes the largest responsibility using

$$\hat{z}_i \leftarrow \arg\max_{\mu_j} \frac{\pi_j \mathcal{N}(z_i|\mu_j, \sigma_j^2)}{\sum_q^Q \pi_q \mathcal{N}(z_i|\mu_q, \sigma_q^2)}, \tag{5}$$

but rely on differentiable soft quantization

$$\tilde{z}_i = \sum_{j=1}^{Q} \frac{\pi_j \mathcal{N}(z_i|\mu_j, \sigma_j^2)}{\sum_q^Q \pi_q \mathcal{N}(z_i|\mu_q, \sigma_q^2)} \mu_j, \tag{6}$$

to calculate gradients during backward propagation.

As for entropy model, we use a 3D CNN-based context model following [7]. Finally, the loss function of the whole system can be written as:

$$L = \alpha L_{\text{dis}} + \frac{1}{G} \sum_{g=1}^{G} L_{\text{ent},g}. \tag{7}$$

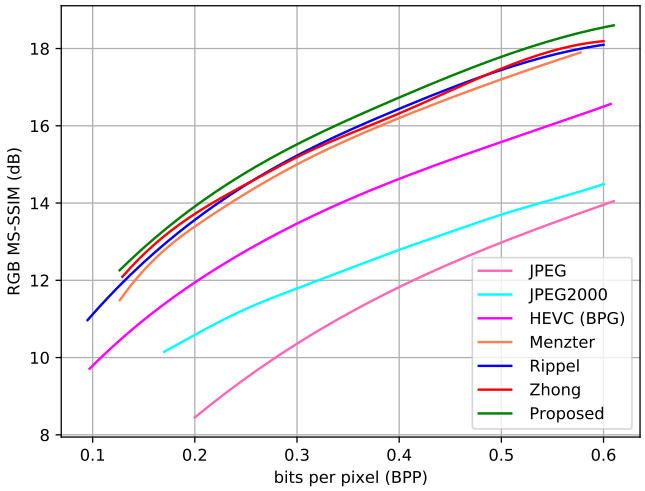


Fig. 4. Comparisons of rate-distortion performance on Kodak dataset.

4. EXPERIMENTS

4.1. Implementation Details

Following [11], we obtain the training data by merging three public datasets, namely DIK2K [25], Flickr2K [26] and CLIC2018[1], which contains approximately 4,000 images in total. In the training process, we vary the quantized feature map $\hat{\boldsymbol{z}}$'s channel number from 16 to 80 to get five models for different compression rate. We randomly crop original input images into 256×256 patches and perform randomly horizontal or vertical flipping. The feature channel number of the head is set to 80. The body of the encoder is composed of four stages where our proposed two feature aggregation strategies and pixel-wise attention module are used. We set the feature channel numbers of these stages to 192, 288, 384, and 480. Similarly, the setting of the decoder is simply the inverse version of the encoder. We use the quantization setting following the best setting as the method by [11] which sets group number $G = 3$, ratio vector $\mathbf{r} = [0.25, 0.5, 0.25]$ with quantization level $\mathbf{q} = [3, 5, 7]$. Our models are trained by Adam [27] optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1 \times 10^{-8}$ and mini-batch size equals to 32. The training epochs are set to 400. The learning rate are initialized as 1×10^{4}, 1×10^{-4}, 5×10^{-5} and 1×10^{-4} for encoder, quantizer, entropy model, decoder and then decrease to one-fifth at 250th, 300th, 350th epoch, respectively. As for loss function, we choose negative MS-SSIM for the distortion loss L_{dis} and set $\alpha = 256$.

We test the proposed method on widely used Kodak lossless image database which has 24 uncompressed 768×512 images. To evaluate the rate-distortion performance, the rate is measured by bits per pixel (BPP) and the reconstruction quality is measured by MS-SSIM [28] which is more consistent with human visual perception than PSNR. We use the rate-distortion (RD) curves to demonstrate the coding efficiency.

4.2. Ablation Study

We study the effect of our proposed two feature aggregation strategies and pixel-wise attention module on the Kodak dataset. All ablation results are obtained over the input mini-batch size of 32 and latent representation channel number of 32.

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

Table 1. Ablation study of our proposed feature aggregation strategies. Inter or Intra means inter-stage or intra-stage feature aggregation is adopted. PA indicates the pixel-wise attention module is adopted.

Method	PSNR	MS-SSIM	bpp
Baseline	27.00	0.9652	0.2563
Intra	27.14	0.9670	0.2563
Inter	27.12	0.9661	0.2564
PA	27.08	0.9662	**0.2552**
Intra + Inter	27.20	0.9673	0.2560
Intra + Inter + PA	**27.24**	**0.9677**	0.2557

As shown in Tab. 1, the model with intra-stage feature aggregation or inter-stage feature aggregation can produce better reconstructions which have higher PSNR and MS-SSIM and compress slightly more effectively. And the performance of our model is further enhanced when the two feature aggregation strategies are used simultaneously, which means that our proposed feature aggregation strategies have the ability to help the generation of more informative and compact latent representations. Besides, the model with PA outperforms the baseline model and the model with our proposed two feature aggregation strategies and pixel-wise attention module achieves the best performance, which achieves a 0.24, 0.0025 increase in terms of PSNR and MS-SSIM comparing to the baseline model, respectively. Note that our PA module is lightweight, which means our PA module doesn't incur too much overhead while improving the performance of the compression system.

4.3. Comparisons

To further evaluate the performance, we compare our proposed method against three most well-known compression standards, JPEG, JPEG2000, and HEVC, as well as several recent deep learning-based methods by [11], [7], and [29].

The rate-distortion performance on Kodak dataset is depicted in Fig. 4. MS-SSIM is converted to decibels by $-10\log_{10}(1-\text{MS-SSIM})$ to illustrate the difference clearly. Our method outperforms the traditional compression standards JPEG, JPEG2000, and HEVC, as well as the deep learning-based methods by [7], and [29]. Additionally, compared with the method by [11], our approach achieves better coding performance when the tested BPP is larger.

5. CONCLUSION

In this paper, we propose a hierarchical feature aggregation network (HFAN) for the en/decoder design of a deep image compression system. Specifically, we propose two strategies, namely inter-stage feature aggregation and intra-stage aggregation. The inter-stage aggregation integrates multi-scale information thereby producing more contextual features. The intra-stage aggregation fuses features within the same stage to enrich representations of one specific resolution. Besides, we incorporate a lightweight pixel-wise attention mechanism to further enhance the discriminative ability of our network and achieve high coding efficiency. Ablation studies validate the effectiveness of our proposed feature aggregation strategies and pixel-wise attention module. Extensive experiments clearly demonstrate that our method achieves superior performance than traditional compression standards including JPEG, JPEG2000, HEVC, and the state-of-the-art deep image compression methods without a hyperprior variational autoencoder.

6. REFERENCES

[1] Lucas Theis, Wenzhe Shi, Andrew Cunningham, and Ferenc Huszár, "Lossy image compression with compressive autoencoders," 2017.

[2] Johannes Ballé, "Efficient nonlinear transforms for lossy image compression," in *2018 Picture Coding Symposium (PCS)*, 2018, pp. 248–252.

[3] Eirikur Agustsson, Fabian Mentzer, Michael Tschannen, Lukas Cavigelli, Radu Timofte, Luca Benini, and Luc V Gool, "Soft-to-hard vector quantization for end-to-end learning compressible representations," in *Advances in Neural Information Processing Systems*, 2017, vol. 30, pp. 1141–1151.

[4] Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston, "Variational image compression with a scale hyperprior," 2018.

[5] 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, vol. 31, pp. 10771–10780, Curran Associates, Inc.

[6] Jooyoung Lee, Seunghyun Cho, and Seung-Kwon Beack, "Context-adaptive entropy model for end-to-end optimized image compression," 2019.

[7] Fabian Mentzer, Eirikur Agustsson, Michael Tschannen, Radu Timofte, and Luc Van Gool, "Conditional probability models for deep image compression," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[8] Johannes Ballé, Valero Laparra, and Eero P. Simoncelli, "End-to-end optimized image compression," *CoRR*, vol. abs/1611.01704, 2016.

[9] George Toderici, Sean M. O'Malley, Sung Jin Hwang, Damien Vincent, David Minnen, Shumeet Baluja, Michele Covell, and Rahul Sukthankar, "Variable rate image compression with recurrent neural networks," 2016.

[10] Eirikur Agustsson, Michael Tschannen, Fabian Mentzer, Radu Timofte, and Luc Van Gool, "Generative adversarial networks for extreme learned image compression," in *Proceedings of the IEEE/CVF ICCV*, October 2019.

[11] Zhisheng Zhong, Hiroaki Akutsu, and Kiyoharu Aizawa, "Channel-level variable quantization network for deep image compression," in *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2020.

[12] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto, "Learned image compression with discretized gaussian mixture likelihoods and attention modules," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[13] G. K. Wallace, "The jpeg still picture compression standard," *IEEE Transactions on Consumer Electronics*, vol. 38, no. 1, pp. xviii–xxxiv, 1992.

[14] A Skodras, C Christopoulos, and T Ebrahimi, "The jpeg 2000 still image compression standard," *IEEE Signal Processing Magazine*, vol. 18, no. 5, pp. 36 – 58, 2001.

[15] 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, 2013.

[16] George Toderici, Damien Vincent, Nick Johnston, Sung Jin Hwang, David Minnen, Joel Shor, and Michele Covell, "Full resolution image compression with recurrent neural networks," in *CVPR*, July 2017.

[17] Chaoyi Lin, Jiabao Yao, Fangdong Chen, and Li Wang, "A spatial rnn codec for end-to-end image compression," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[18] Lirong Wu, Kejie Huang, and Haibin Shen, "A gan-based tunable image compression system," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, March 2020.

[19] Suraj Kiran Raman, Aditya Ramesh, Vijayakrishna Naganoor, Shubham Dash, Giridharan Kumaravelu, and Honglak Lee, "Compressnet: Generative compression at extremely low bitrates," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, March 2020.

[20] Mu Li, Wangmeng Zuo, Shuhang Gu, Debin Zhao, and David Zhang, "Learning convolutional networks for content-weighted image compression," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[21] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto, "Deep residual learning for image compression," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019.

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2016.

[23] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu, "Residual dense network for image super-resolution," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2472–2481.

[24] Jie Liu, Wenjie Zhang, Yuting Tang, Jie Tang, and Gangshan Wu, "Residual feature aggregation network for image super-resolution," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 2359–2368.

[25] Radu Timofte, Eirikur Agustsson, Luc Van Gool, Ming-Hsuan Yang, and Lei Zhang, "Ntire 2017 challenge on single image super-resolution: Methods and results," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, July 2017.

[26] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee, "Enhanced deep residual networks for single image super-resolution," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, July 2017.

[27] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," 2017.

[28] 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 Vol.2.

[29] Oren Rippel and Lubomir Bourdev, "Real-time adaptive image compression," in *Proceedings of the 34th International Conference on Machine Learning*, Doina Precup and Yee Whye Teh, Eds. 06–11 Aug 2017, vol. 70 of *Proceedings of Machine Learning Research*, pp. 2922–2930, PMLR.