Variable Rate Image Compression with Content Adaptive Optimization

Tiansheng Guo[1], Jing Wang*[1], Ze Cui[1], Yihui Feng[1], Yunying Ge[1], Bo Bai[1]
[1]Huawei Technologies, Beijing, China
*wangjing215@huawei.com

Abstract

In this paper, we propose a variable rate image compression framework for low bit-rate image compression task. Unlike most of the variational auto-encoder (VAE) based methods, our proposal is able to achieve continuously variable rate in a single model by introducing a pair of gain units into VAE. Besides, a content adaptive optimization is applied to adapt the latent representation to the specific content while keeping the parameters of the network and the predictive model fixed. After that, due to the variable rate characteristics of our method, each image can be compressed into any quality level through a unified codec. Finally, an efficient rate control algorithm is designed to find the optimal bit allocation scheme under the constraint of the low rate challenge.

1. Introduction

Lossy image compression is one of the most fundamental and valuable problems in image processing to maintain image quality with less storage or transmission. Recently, learned image compression methods have derived significant interests and achieve much better performance than the classical image codecs, such as JPEG [1], JPEG2000 [2] and BPG [3]. Thanks to variational autoencoder (VAE) and scalar quantization assumption [4], learned image compression methods are able to trained end-to-end and achieve satisfactory results. In order to further improve the rate-distortion performance, hyperprior network [5] and autoregressive model [6] were introduced into the VAE framework to enhance the entropy estimation. Besides, Nonlocal residual block [7], attention mechanism [8] and multi-scale fusion [9] were inserted into the encoder/decoder network to improve feature extraction and reconstruction performance. On the other hand, some researchers tried to solve the obstacles of the deep image compression methods in practice. Balle *et al*. [10] proposed an integer network to avoid floating point inconsistence and enable reliable cross-platform encoding and decoding of images using variational models. Johnston *et al*. [11] applied automatic network optimization techniques and GDN without square/square root components to reduce the run time of the entire VAE architecture. Choi *et al*. [12] incorporated fully connection networks into the convolution unit and adjusted quantization bin sizes to realize rate adaption of the deep image compression methods. Cui *et al*. [13] proposed a continuously variable rate image compression framework G-VAE (Gained Variational Autoencoder), which adds a pair of gain units at the output of encoder and the input of decoder and endows the fixed-rate deep image compression frameworks continuously variable rate with negligible additional parameters and computation.

Motivated by the above latest breakthroughs, we incorporate the attention module [14], universal quantization [12] and multi-scale parallel context module [18] into the G-VAE [13] to obtain an optimal solution with high R-D performance and continuous rate adaption. Through adding a pair of gain units to VAE, the G-VAE framework could achieve continuously variable rate in a single model. By moving the attention module [14] to the higher scale feature layer, the network's performance could be effectively improved. Besides, different quantization strategies are used in the training process to improve the accuracy of entropy estimation and reconstruction quality. Rounding quantization is used as the input of decoder and hyperencoder, and universal quantization [12] is used as the input of entropy estimation module. Moreover, a rate controlling scheme is designed to select the best parameter setting for each image considering the 0.15 BPP constraint in the low bit-rate challenge. With these methods, our methods achieve 32.594 in PSNR (optimized in MSE) and 0.9781 in MS-SSIM (optimized in MS-SSIM) in the validation sets.

2. Proposed Method

Figure 1 depicts the proposed image compression framework. The encoder and decoder consist of convolution layers, GDN/IGDN units and attention modules. Convolution parameters are denoted as number of $filters \times kernel\_height \times kernel\_width / stride$, where $\uparrow$ and $\downarrow$ represent upsampling and downsampling respectively. GDN and IGDN represent generalized divisive and the inverse

counterpart respectively [4, 5]. Gain unit and inverse gain unit are introduced into our method to achieve continuously variable rate with negligible additional parameters and computation. UnivQuant represents universal quantization [12]. AE and AD represent arithmetic encoder and decoder.

2.1. Variable rate framework

Cui *et al*. [13] proposed a continuously variable rate image compression framework G-VAE, which adds a pair of gain units at the output of encoder and the input of decoder and endows the fixed-rate deep image compression frameworks continuously variable rate with negligible additional parameters and computation. The main element of the gain unit is a gain matrix, which consist of several gain vectors. Meanwhile, another gain unit is introduced at the input of the decoder to rescale the quantized gained latent representation and ensure that the decoder could reconstruct the image correctly. The inverse gain vector and the corresponding gain vector always appear in pairs, which determine the rate-distortion performance of the model. In order to enable the proposed method achieving rate adaption, pairs of gain vectors are added to the specified positions depicted as Figure 1. The loss function of the proposed method is defined as below:

$$L = \sum_{s=1}^{N} \beta_s \cdot D + R_y + R_z \tag{1}$$

where R_y and R_z represents the expected bit rate of the quantized gained latent representation and the quantized gained hyper latent representation respectively, and s represents the index of the gain vectors in the gain matrix. Distortion loss D represent mean squared error loss in our MIATLPSNR method and MS-SSIM loss in our MIATLSSIM method. By applying the interpolation between the adjacent trained gain vector pairs in the inference process, the proposed model can achieve arbitrary point in the whole continuous range of the R-D curve.

2.2. Attention mechanism

Attention can be guided to bias the allocation of available processing resources towards the most informative. In order to capture the global correlations and useful features, we utilize the residual non-local attention block [14]. Different from the previous works [9, 14], we place attention module in front of GDN and only use one attention module. This change not only reduces the network structure, but also makes attention module extract more powerful features, which effectively improves network performance.

2.3. Quantization

Quantization operation generally is indispensable to generate discrete codes. However, its gradient is zero almost everywhere except it is infinite for several threshold points. To handle this issue, several continuous proxy methods have been presented, including smoothed [19], soft-to-hard approximation [20], continuous approximation [4, 5]. Choi *et al*. [12] proposed universal quantization, which proven to achieve better R-D performance:

$$\hat{y_s} = round(\bar{y_s} + u) - u \tag{2}$$

In our framework, we adopt smoothed rounding quantization [19] as the inputs of decoder and hyperencoder, and universal quantization to model the entropy of quantized codes.

2.4. Parallel context module

As shown in Figure 1, we adopt a multi-scale parallel context module [18], which contains 3 parallel masked convolution layers. The kernel sizes in each masked convolution are 3×3, 5×5 and 7×7 respectively. In this way, only previously decoded points can be used to decode the current point. The effect of points in the parallel convolution repeating region will be magnified, and there will be no blind spots caused by the accumulation of mask convolutions.

2.5. Content adaptive optimization

Ideally, a well-trained encoder can generate an optimal latent representation of any image to be compressed by forward pass during the test phase. Unfortunately, the inevitable gap between datasets and the limited expressiveness of the network may make the latent representation generated by the encoder sub-optimal. Inspired by [18], we adopted a content adaptive operation to refine the latent representation of each image. It is worth noting that this operation only directly changes the value of the latent representation of the image during the encoder transformation stage, making the modified latent representation fit the decoder that has been deployed and fixed at the receiver as much as possible. Different from the original content adaptive method [18], we also refine the side information after the latent representation has been refined, so that it can more accurately estimate the distribution of the refined latent representation. The above process can be formulated by the following optimization problem:

$$\underset{\hat{y}}{argmin}(-\log p(\hat{y}) - \log p(\hat{z}) + \beta \cdot d(x, \hat{x})) \tag{3}$$

$$\underset{\hat{z}}{argmin}(-\log p(\hat{y}) - \log p(\hat{z})) \tag{4}$$

Following [18], we refine the latent representation through an iterative procedure. More specifically, we treat the latent representation as a set of learnable parameters

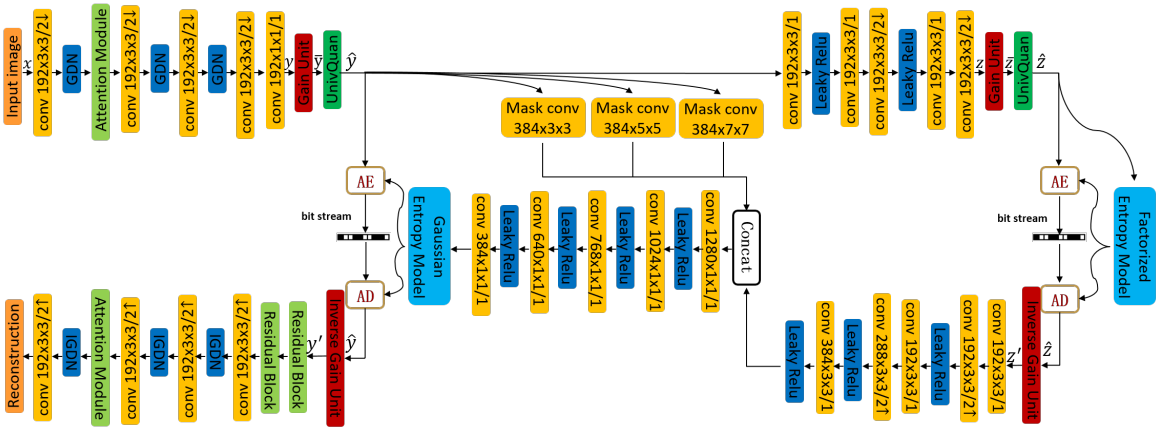

Figure 1. The image compression framework.

during the encoding phase and use gradient descent algorithm to update it iteratively. When adaptive optimization is completed, we could get a set of refined latent representations. After that, quantization and arithmetic coding will be performed to generate a transportable bit stream.

2.6. Rate controlling scheme

Rate control mechanism is one of the characteristics of traditional image compression methods. Rate control is defined to allocate models as soon as the allocated bit budget is fully utilized:

$$\begin{aligned} &\min_{y} \sum_{i=1}^{N}\sum_{j=1}^{M} D(X_j, \hat{X_{ij}}) \\ s.t. &\sum_{i=1}^{N}\sum_{j=1}^{M} y_{ij} R(X_i, \hat{X_{ij}}) < R_{max} \\ &\sum_{j=1}^{M} y_{i,j} = 1, i = 1, 2, ..., N \\ &y_{i,j} \in \{0, 1\}, i = 1, 2, ..., N, j = 1, 2, ...M \end{aligned} \tag{5}$$

Where D and R are distortions and rates between original image X_i and the reconstructed image $\hat{X_{ij}}$. M is the number of results, N is the number of image, and R_{max} is the bit budget of the image set. Optimizing Equation 5 is a NP hard problem. So we propose a two-step method to get an approximate optimal solution. Firstly, an allocation vector satisfying rate constrained conditions can be found based on Lagrange Relaxation [21]. Then, a better allocation vector will be found based on greedy algorithm.

3. Experiments

3.1. Implementation details

For training, we use more than 6000 images collected from CLIC training set and a self-building dataset. 8 patches with the size of 256×256 are randomly cropped from 8 full resolution images for training in each iteration. It takes about 3M iterations for our model to reach a stable state. We trained the model with Adam optimizer and the learning rate was initially set to 1×10^{-4} and reduced by 0.5 times when the total iterations reach 2M and 2.5M. In our experiments, n denoted the number of gain vector pairs, which was the same as the number of Lagrange multipliers. We prepared two sets of Lagrange multipliers as below:

$$\beta_{msssim} = \{0.003, 0.0015, 0.0005\} \tag{6}$$

$$\beta_{mse} = \{0.007, 0.002, 0.001\} \tag{7}$$

where β_{msssim} and β_{mse} correspond to the models trained with MS-SSIM and MSE loss respectively. In the training process, we randomly select the index s from 1 to 3 in each iteration to obtain the corresponding gain vector m_s, inverse-gain vector m'_s and Lagrange multiplier β_s from gain matrix M, inverse-gain matrix $M'$and $\beta_{msssim/mse}$ respectively. The selected gain/inverse-gain vector will be optimized with the corresponding Lagrange multiplier to adapt to different bit rates. Notably, we only need to train one model for each submission, since our variable rate model can compress the image into any desired quality level.

In the low-rate challenge, the entire test set should be compressed into 0.15 BPP or smaller. Under this constraint, we choose 32 different gain vector pairs and get 32 different

Dataset	Method	PSNR	MS-SSIM	BPP
validation	MIATLPSNR	32.594	0.9645	0.15
	MIATLSSIM	30.170	0.9781	0.15
test	MIATLPSNR	33.095	0.9693	0.15
	MIATLSSIM	30.472	0.9822	0.15

Table 1. Evaluation results on CLIC2020 datasets.

compression quality level around 0.15 bpp for rate control. In other words, any image in the test set can be compressed into any level of the 32 different quality level to reach the highest MS-SSIM/PSNR with 0.15 BPP.

3.2. Results and analysis

The evaluation results on CLIC2020 validation and test datasets are shown in Table 1. Our methods MIATLPSNR and MIATLSSIM achieve outstanding results on PSNR and MS-SSIM in the low rate compression competition. Specifically, Our MIATLPSNR can yield 32.594dB/33.095dB of PSNR on validation/test set and MIATLSSIM can reach 0.9781/0.9822 of MS-SSIM on validation/test set.

In order to prove the effectiveness of our proposal more convincingly, ablation experiments are conducted on the validation set and the results are summarized in Table 2 and Table 3. It is worth noting that the performance of our basic variable rate model outperforms most methods in competition, which proves the superiority of our network structure in attention mechanism and parallel convolution. In the validation dataset, after content adaptive operation, MS-SSIM increased from the original 0.9776 to 0.9777, while the BPP dropped from 0.149489 to 0.147262. PSNR increased from the original 32.401 to 32.482, while BPP decreased from the original 0.149995 to 0.148495. In the test dataset, it achieves similar effects, as shown in Table 2 and Table 3. This illustrates that the content adaptive operation can find a better latent representation which can represent the corresponding image more accurately with less bits. In addition, rate control strategy is adopted to control the final BPP to 0.15 with about 0.11 dB gain in PSNR and 0.0004 gain in MS-SSIM.

4. Conclusion

In this paper, we solved the problem of image compression under low bit-rate constraint by a single variable rate model. Attention mechanism and multi-scale parallel context module are adopted to improve the performance of our model. Content adaptive compression strategy is applied to generate better latent representation without architecture refinements. Besides, we designed an efficient rate control algorithm to maximize PSNR/MS-SSIM under 0.15 BPP constraint. As shown in the results of the challenges on the validation set, our approaches MIATLPSNR and MIATLSSIM yield outstanding performance on PSNR and MS-SSIM respectively.

Dataset	Method	BPP	PSNR
validation	variable rate	0.1500	32.4014
	variable rate + adaptive	0.1485	32.4820
	variable rate + adaptive + rate control	0.1500	32.5939
test	variable rate	0.1500	32.9179
	variable rate + adaptive	0.1485	33.0178
	variable rate + adaptive + rate control	0.1500	33.0954

Table 2. Ablation results of PSNR.

Dataset	Method	BPP	PSNR
validation	variable rate	0.1495	0.9776
	variable rate + adaptive	0.1473	0.9777
	variable rate + adaptive + rate control	0.1500	0.9781
test	variable rate	0.1499	0.9817
	variable rate + adaptive	0.1479	0.9818
	variable rate + adaptive + rate control	0.1500	0.9822

Table 3. Ablation results of MS-SSIM.

References

[1] Wallace G K. The JPEG still picture compression standard. IEEE transactions on consumer electronics, 38(1): xviii-xxxiv, 1992.

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

[3] Bellard F. BPG Image format. URL https://bellard.org/bpg, 2015.

[4] Ballé J, Laparra V, Simoncelli E P. End-to-end optimized image compression. arXiv preprint arXiv:1611.01704, 2016.

[5] Ballé J, Minnen D, Singh S, *et al*. Variational image compression with a scale hyperprior. arXiv preprint arXiv:1802.01436, 2018.

[6] Minnen D, Ballé J, Toderici G D. Joint autoregressive and hierarchical priors for learned image compression. Advances in Neural Information Processing Systems, 10771-10780, 2018.

[7] Liu H, Chen T, Guo P, *et al*. Non-local attention optimized deep image compression. arXiv preprint arXiv:1904.09757, 2019.

[8] Mentzer F, Agustsson E, Tschannen M, et al. Conditional probability models for deep image compression. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4394-4402, 2018.

[9] Zhou L, Sun Z, Wu X, *et al*. End-to-end optimized image compression with attention mechanism. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 0-0, 2019.

[10] Ballé J, Johnston N, Minnen D, *et al*. Integer Networks for Data Compression with Latent-Variable Models, International Conference on Learning Representations, 2019.

[11] Johnston N, Eban E, Gordon A, *et al*. Computationally efficient neural image compression. arXiv: Image and Video Processing, 2019.

[12] Choi Y, Elkhamy M, Lee J, *et al*. Variable Rate Deep image compression with a conditional autoencoder. arXiv: Image and Video Processing, 2019.

[13] Cui Z, Wang J, Bai B, *et al*. G-VAE: A continuously variable rate deep image compression framework. arXiv preprint arXiv:2003.02012, 2020.

[14] Zhang Y, Li K, Li K, *et al*. Residual non-local attention networks for image restoration. arXiv preprint arXiv:1903.10082, 2019.

[15] Ziv J. On universal quantization. IEEE Transactions on Information Theory, 1985, 31(3): 344-347.

[16] Van den Oord A, Kalchbrenner N, Espeholt L, *et al*. Conditional image generation with pixelcnn decoders. Advances in neural information processing systems, 4790-4798, 2016.

[17] Ballé J, Laparra V, Simoncelli E P. Density modeling of images using a generalized normalization transformation. arXiv preprint arXiv:1511.06281, 2015.

[18] Campos J, Simon M, Djelouah A, *et al*. Content Adaptive Optimization for Neural Image Compression. arXiv preprint arXiv:1906.01223, 2019.

[19] Theis L, Shi W, Cunningham A, *et al*. Lossy image compression with compressive autoencoders. arXiv preprint arXiv:1703.00395, 2017.

[20] Agustsson E, Mentzer F, Tschannen M, *et al*. Soft-to-hard vector quantization for end-to-end learning compressible representations. Advances in Neural Information Processing Systems, 1141-1151, 2017.

[21] Juttner.A, Szviatovszki.B, Mecs.I, Rajko.Z, Z.Lagrange Relaxation Based Method for the QoS Routing Problem. IEEE INFOCOM, 2001.