Image Compression Base on Trit-Plane and Attention

Mingjun Fang[1] 23020231154138 (school of Information)
Heshan Wang[1] 23020231154227 (school of Information)
Kun Yang[1] 23020231154244 (school of Information)
CanYi Liu[1] 23020231154149 (school of Information)
JiaWei Lai[1] 23020231154198 (school of Information)
[1]XiaMen University

Abstract

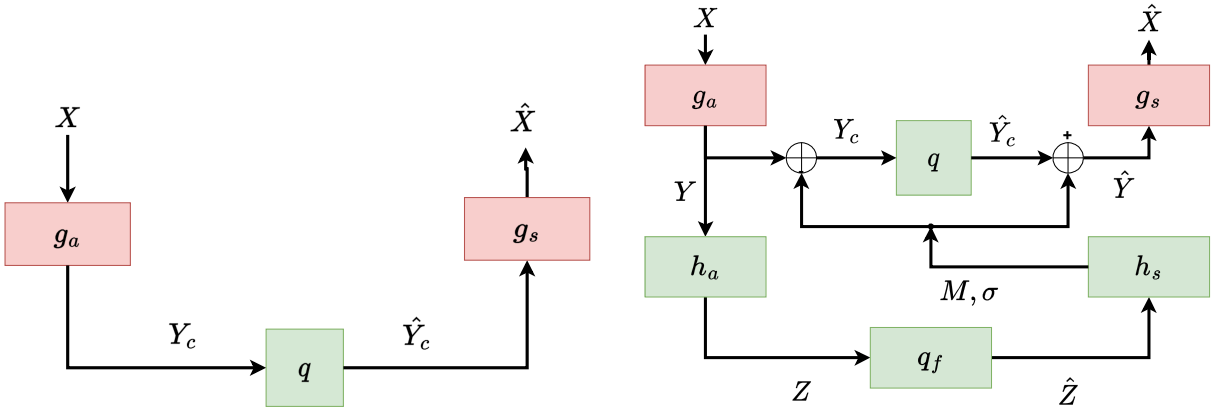
Traditional image compression algorithms have been relatively mature, such as JPEG(Wallace 1991), JPEG2000. With the development of deep learning, convolutional neural networks have been leveraged in image compression. Trit-plane coding is one of the best practices of CNN in image compression. It enables deep progressive image compression, but it cannot use autoregressive context models. In this paper, we propose the attention-based trit-plane coding (ATC) algorithm to achieve progressive compression more compactly. First, we intend to develop the attention-based rate reduction module to accurately estimate the trit probabilities of latent elements and thus encode the trit-planes compactly. Second, we plan to develop the attention-based distortion reduction module to refine partial latent tensors from the trit-planes and improve the reconstructed image quality. Third, we aspire to propose a retraining scheme for the decoder to attain better rate-distortion tradeoffs. Extensive experiments show that ATC outperforms the baseline trit-plane code significantly. We plan to evaluate our model using three datasets, which will be introduced in more detail in the plan section.

Introduction

With the development of the digital information age, the amount of data faced by people has increased dramatically, and more and more attention has been paid to the research of data compression technology. Image compression has been one of the important topics. Traditional image compression is based on the theory of information theory and digital signal processing technology, through the elimination of redundancy between digital image pixels to achieve image compression processing. Traditional image compression algorithms have been relatively mature, such as JPEG (Wallace 1991), JPEG2000 (Skodras, Christopoulos, and Ebrahimi 2001), and so on. However, with the in-depth study and application of these traditional image compression methods, many drawbacks of these methods have been found, such as the serious square effect of recovered images at high compression ratios, and the characteristics of the human eye's visual system are not easy to be introduced into the traditional methods.

Deep learning has grown rapidly in recent years, people began to explore deep learning and successfully applied it to Image Compression. CNNs have been developing rapidly in the image field, especially in the field of computer vision, and the sparse connection and parameter-sharing characteristics of CNN convolutional operation make CNNs advantageous in image compression. These two features of convolutional neural networks better reduce the computational complexity and allow training towards deeper and better network structures. Using these modules, they provide decent rate-distortion (RD) results. Several innovations have been made to improve the RD performance, including differentiable quantization approximation (Agustsson et al. 2017; Ballé, Laparra, and Simoncelli 2016), hyperprior (Ballé et al. 2018), contextual models (He et al. 2021; Mentzer et al. 2018; Minnen, Ballé, and Toderici 2018) and a priori models (Cheng et al. 2020; Cui et al. 2021), as well as bit Rate-Distortion Optimization (RDO) (Wang et al. 2022), frequency-aware transform block (FAT) (Li et al. 2023), and QPressFormer (Luka, Negrel, and Picard 2023). Therefore, deep image codecs are better and more competitive than traditional codecs. better and more competitive.

Most image compression methods are non-progressive. Therefore, these methods expect to have the complete compressed image available for decoding. However, in many cases, making the entire file available is a challenge. As a result, the user or system experiences some delay before reconstructing the image for viewing or further processing. Progressive compression (Liu et al. 2021) solves this problem by allowing the decoder to obtain an initial preview with even a small portion of the data. Afterward, the decoder can reconstruct a better-quality image by receiving the rest of the bits. However, relatively few deep codecs support such progressive compression or scalable coding (Ohm 2005). Many codecs require multiple pieces of training of their networks to achieve compression at as many bit rates as possible (Ballé et al. 2018; Cheng et al. 2020). Some codecs support variable rate coding (Cui et al. 2021; Yang et al. 2021), but they should generate multiple bitstreams for different bitrates, and it is more efficient to truncate a single bit stream for different bit rates (Lee et al. 2022; Lu et al. 2021). Besides, these models cannot use the context model for encoding and decoding (He et al. 2021; Lee, Cho, and Beack 2018). They assume that the codec contexts are



(a) Framework of Traditional Base Model

(b) Framework of Our Method's Foundational Model

Figure 1: Image Compression Framework

synchronized, but in reality, these potential elements are in different states. Therefore new context models need to be proposed in order to solve this problem. Meanwhile, we noticed that the attention mechanism helps to obtain context information. So, we intend to implement a context image compression algorithm based on the attention mechanism to present better compression results.

In this paper, we propose the attention trit-plane coding (ATC) algorithm for progressive image compression, which introduces novel context models to address the limitations of existing models. Based on our observations, the current models face limitations in effectively utilizing the context model for encoding and decoding. These limitations arise from the assumption that the codec contexts are synchronized, which does not align with the reality where these potential elements often exist in different states. This discrepancy highlights the need for the development of new context models to address this gap.

To overcome this problem, we leverage the attention mechanism, which has proven effective in obtaining context information, to enhance our compression algorithm. By incorporating the attention mechanism into the ATC algorithm, we aim to achieve better compression results and improve the overall performance of the image compression process.

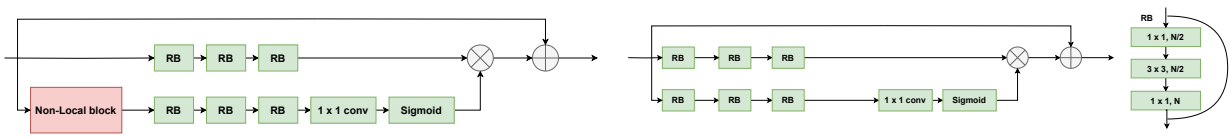
Our proposed ATC algorithm takes advantage of the attention mechanism to selectively focus on informative regions of the image during the encoding and decoding process. This allows us to allocate more bits to important regions while reducing the bit allocation for less significant regions, resulting in improved compression efficiency. By considering the contextual information in a more dynamic and adaptive manner, our algorithm can effectively handle the variations and complexities present in real-world image data.

Experiments and evaluations are conducted to validate the effectiveness of the proposed ATC algorithm. The results demonstrate that our approach outperforms existing methods in terms of compression ratio, and we use RD point (Rate-Distortion point) to evaluate the trade-off between compression rate and distortion

Related Work

Early image codecs mainly used lossless compression to preserve the original image information but required a large amount of storage space and transmission bandwidth. At the same time, image compression also mainly relied on fixed bit rates for compression, which meant that the same bit rate was used for compression regardless of the image content. To achieve variable bit rate compression, multiple training sessions were conducted at the cost of time and memory.

With the rapid development of deep learning technology, new breakthroughs have been made in coding and decoding, and variable bit rate compression technology has also been further improved. Some research works use deep neural networks for coding and decoding, achieving significant progress. Some research works use convolutional neural networks (CNN) to encode and decode video frames to achieve more efficient compression. For example, an end-to-end image compression framework based on CNNs consists of a non-linear analysis transform (encoder)(Ballé, Laparra, and Simoncelli 2016)., a uniform quantizer (multi-binary rounding), and a non-linear synthesis transform (decoder). Some research works also explore the use of recurrent neural networks and long short-term memory networks for encoding and decoding to achieve smoother playback effectscite(Gregor et al. 2016). Meanwhile, some research works use deep neural networks to predict the content of images and dynamically adjust the bit rate based on the characteristics of the content. This method can achieve more precise bit rate control, thereby further reducing the bit rate while ensuring quality as a representative method using RNN for image compression(Toderici et al. 2015), which first used convolutional LSTM to achieve variable bit rate end-to-end learning image compression. Based on scalable neural networks, models (Yang et al. 2021) use subsets of network parameters to control the bit rate. With the introduction of transforms, vision transforms(Dosovitskiy et al. 2020) has also been applied to the field of image compres-



(a) Original attention modules

(b) Framework of Our Method's Foundational Model

Figure 2: Different attention modules

sion. For example, Swin transform is applied to their encoder, decoder, super-encoder, and super-decoder in(Zhu, Yang, and Cohen 2021), achieving good results. Some research also uses wavelet transform for video encoding and decoding to achieve better video quality and compression ratio. For example, IWAVE++(Ma et al. 2020)is a new end-to-end optimized image compression scheme.

In learning-based decoders, context models are often adopted. These decoders typically use deep learning techniques to learn and predict the content of images or videos. Context models(He et al. 2021)are used here to help predict the next content of images or videos, therefore achieving more efficient and higher-quality compression and reconstruction. Channel-based conditional context and residual prediction modules based on potential representations(Minnen and Singh 2020) have been designed by some works to optimize network architecture.

Overall, the development process of compression technology is an ongoing process of optimization and improvement. With the continuous development of technology, we can expect to see more innovations and improvements to achieve higher quality and more efficient image compression.

Method

Formulation of Learned Compression Models

In the traditional encoding-decoding compression approach (Goyal 2001), image compression can be formulated by (as Fig. 1(a))

$$\begin{aligned} \boldsymbol{Y} &= g_a(\boldsymbol{X}) \\ \hat{\boldsymbol{Y}} &= q(\boldsymbol{Y}) \\ \hat{\boldsymbol{X}} &= g_s(\hat{\boldsymbol{Y}}) \end{aligned} \tag{1}$$

where X, $\hat{X}$, Y, and $\hat{Y}$ are raw images, reconstructed images, a latent presentation before quantization, and compressed codes, respectively. q represents the quantization and entropy coding. An image X is transformed into a latent representation Y. Y is then quantized and entropy coded to obtain the compressed bit stream $\hat{Y}$. When needed, $\hat{Y}$ is decoded to obtain the recovered image $\hat{X}$.

The latest and most widely used image compression framework in Fig. 1(b) (Ballé et al. 2018; Cheng et al. 2020; Cui et al. 2021; Lee, Cho, and Beack 2018; Minnen, Ballé, and Toderici 2018; Yang et al. 2021), which is based on the traditional framework, consisting of an encoder g_a, a decoder g_s, a hyper encoder h_a, and a hyper decoder h_s.

Based on the traditional framework, an image X is transformed into a latent representation Y and a hyper latent representation Z sequentially by g_a and h_a.

$$\begin{aligned} \boldsymbol{Y} &= g_a(\boldsymbol{X}) \\ \boldsymbol{Z} &= h_a(\boldsymbol{Y}) \end{aligned} \tag{2}$$

Using the factorized prior model, denoted by $q_f(\cdot)$, Z is digitized to $\hat{Z}$. From $\hat{Z}$, h_s yields M and σ, which contain the means and standard deviations of the elements in Y, respectively. These elements are assumed to be independent Gaussian random variables. Then, the mean-removed (or centered) $\boldsymbol{Y_c} = \boldsymbol{Y} - M$ is quantized to

$$\hat{\boldsymbol{Y}}_{\boldsymbol{c}} = q(\boldsymbol{Y_c}) \tag{3}$$

where rounding is used for the quantizer $q(\cdot)$. Finally, the decoder g_s adds M back to $\hat{Y}_c$ to yield

$$\hat{\boldsymbol{Y}} = \hat{\boldsymbol{Y}}_{\boldsymbol{c}} + M \tag{4}$$

and uses $\hat{Y}$ to reconstruct $\hat{X}$.

Network Architecture

Our network architecture has a similar structure as(Cheng et al. 2019). We use residual blocks to increase the large receptive field and improve the rate-distortion performance. The Decoder side uses subpixel convolution instead of transposed convolution as upsampling units to keep more details. N denotes the number of channels and represents the model capacity. We use the Gaussian mixture model, thus requiring $3 \times N \times K$ channels for the output of the auxiliary autoencoder.

Since the existing framework is already well-established and incorporates a significant amount of information theory knowledge, making modifications to it may be challenging. Therefore, we carefully analyze the existing encoder-decoder architecture and identify areas where enhancements can be made. In this paper, we focus on making modifications to the encoder g_a and decoder g_s components of the framework. This approach allows us to build upon the existing framework while introducing novel contributions to the specific components that we are modifying.

Furthermore, recent works use attention modules to improve the performance of image restoration and compression. The proposed attention module is illustrated in (as Fig. 2(a)), but very time-consuming for training. In Image Compression, non-local blocks are often used to model global contextual relationships to capture long-range dependencies in images. Such a block introduces more computation overhead as it requires global operations on all locations in the image. In terms of context modeling, residual blocks already provide a considerable perceptual domain in

the network architecture, so for some tasks, the non-local block may appear redundant, and removing it can reduce the computational burden. This is especially important in compression tasks that require efficient training and inference, as compression models need to remain efficient when processing large amounts of image data. So, we simplify this module by removing non-local blocks, since deep residual blocks can already capture very large perceptual domains in our network architecture.

A simplified attention module is shown in(as Fig. 2(b)) and can also reduce the loss with moderate complexity. Attention modules can help the networks to pay more attention to challenging parts and reduce the bits of simple parts. Then we insert a simplified attention module into the encoder network (as Fig. 3).

Loss

Following the original framework, the loss function (Liu et al. 2022) of our model is

$$
\begin{aligned}
&L = R + \lambda * 255^2 * d_{\mathrm{MSE}}(\hat{\boldsymbol{x}}, \boldsymbol{x}) \\
&\text{with} \quad d_{\mathrm{MSE}}\left(\hat{\boldsymbol{x}}, \boldsymbol{x}\right) = \frac{1}{HW} \sum_{i=1}^{H} \sum_{j=1}^{W} \left(\hat{\boldsymbol{x}}_{i,j} - \boldsymbol{x}_{i,j}\right)^2
\end{aligned} \tag{5}
$$

where MSE means employing MSE as distortion metric,H and W represent the height and width of the image, respectively. $R = R_{\hat{\boldsymbol{y}}} + R_{\hat{\boldsymbol{z}}}$ represents the total bitrate of $\hat{\boldsymbol{y}}$ and $\hat{\boldsymbol{z}}$.The regularization parameter λ is used to control the trade-off between rate and distortion. In this work, the bitrate is estimated by calculating entropy of $\hat{\boldsymbol{y}}$ and $\hat{\boldsymbol{z}}$, which can be formulated as:

$$
\begin{aligned}
R_{\hat{y}} &= -\sum_{i} \log_2 \left(p_{\hat{y}_i|\hat{z}_i}\left(\hat{y}_i \mid \hat{z}_i\right)\right) \\
R_{\hat{z}} &= -\sum_{i} \log_2 \left(p_{\hat{z}_i|\psi_i}(\hat{z}_i \mid \psi_i)\right)
\end{aligned} \tag{6}
$$

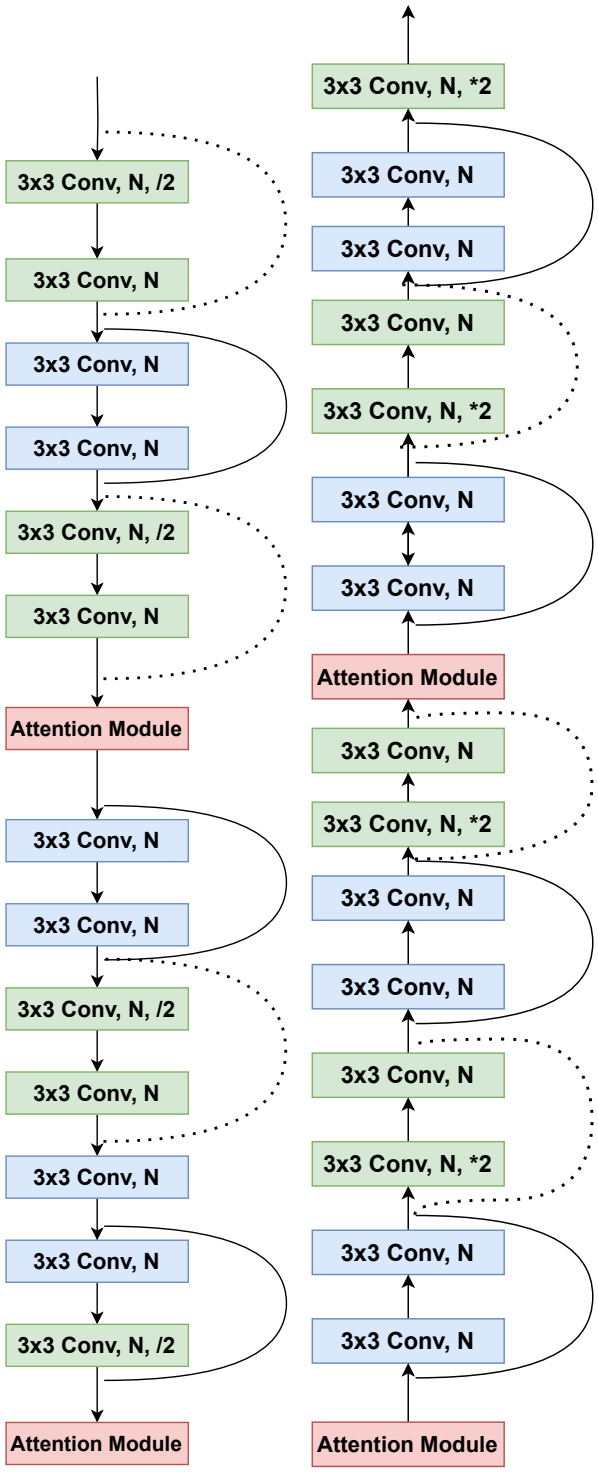
By minimizing this loss function, our model aims to achieve a balance between compression efficiency (rate) and reconstruction quality (distortion).

Experiment

Dataset

The dataset used to train the model in this experiment comes from the validation set of ImageNet 2010(Deng et al. 2009). Since the training data for the input model requires the same dimensions, 10,000 images with dimensions larger than 256x256 were selected from this set. These images were then divided into training and testing sets in an 8:2 ratio for model training.

For the evaluation dataset, we primarily utilized the Kodak dataset, which includes 24 images with dimensions of 768 x 512. Additionally, for comparison with our proposed method, we employed datasets from CLIC and JPEG-AI, containing 41 and 16 images, respectively, with resolutions of up to 2K.



(a) Improved g_a architecture (b) Improved g_s architecture

Figure 3: Encoder Network architecture

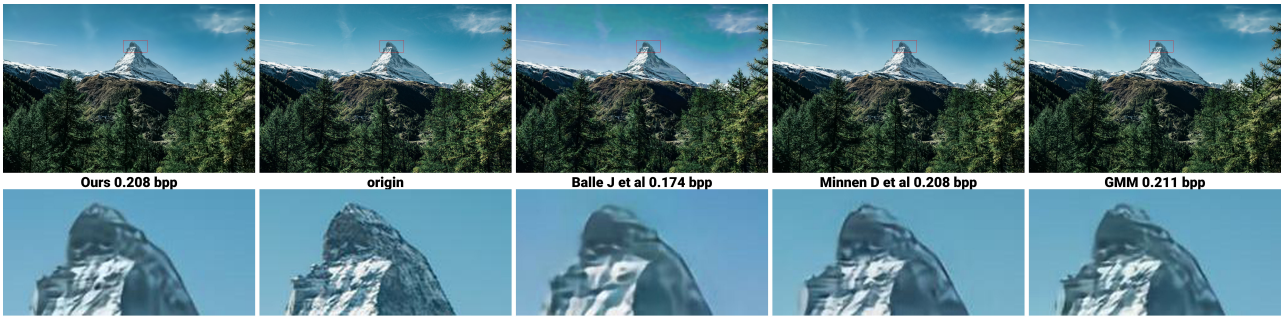


Figure 4: Visualization of reconstructed image

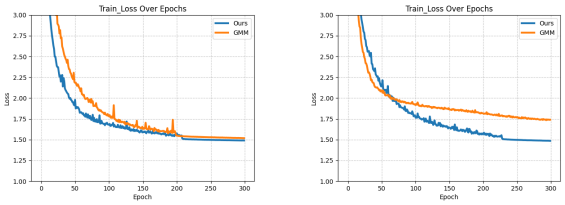
Training Details

In our training process, we utilize the Adam optimizer with a batch size of 32 and employ a strategy for dynamically adjusting the learning rate. Specifically, when the model's loss on the test set does not decrease for five consecutive rounds, we reduce the learning rate by a factor of 0.1, aiming to carefully search for optimal model parameters in the later stages of training. The initial learning rate is set to 0.0001. For data augmentation during training, we apply random cropping to the training set, while central cropping is used for the test set. To prevent gradient explosions, we implement gradient clipping during the training process for gradient scaling. To expedite the training speed, we adopt parallel training, conducting model training on four NVIDIA GeForce RTX 3090 GPUs. Throughout the training process, we compare the losses after each epoch and select the model weights from the epoch with the minimal test set loss as our final model.

Results of Experiment

We compared the performance of our model with previous end-to-end approaches, including the Gaussian mixture model(Cheng et al. 2020), Ballé J et al.(Ballé, Laparra, and Simoncelli 2016) utilized non-linear analysis transforms, uniform quantizers and non-linear synthesis transforms to create their model, achieving excellent results in image compression. Additionally, Minnen et al. (Minnen, Ballé, and Toderici 2018)proposed a novel algorithm composed of analysis transform, uniform quantizer, and synthesis transform.

We trained both the GMM and our proposed model on the same dataset, using the same equipment and hyperparameter settings. The results are depicted in (as Fig. 5. It can be observed from the graph that our proposed model outperforms the GMM in terms of train set loss and test set loss. As the model continues to train, our proposed model demonstrates a better reduction in test set loss. This improvement is attributed to the attention mechanism we introduced, which better captures compression details, and the residual connections, which mitigate overfitting. The corresponding rate-distortion points are depicted in(as Table. 1, to show the coding gain of proposed approaches. It can be observed our proposed approaches can improve the rate-distortion performance regardless of the model capacity.



(a) Train Loss (b) Test Loss

Figure 5: Model Comparison

Table 1: RD point

Algorithm	bpp	MS-SSIM	PSNR
ATC	0.560	0.983	30.26
Baseline	0.756	0.981	29.88

For the other two models, we directly utilized their pre-trained parameters to compare them with our model in the context of image compression evaluation. To demonstrate the superiority of our approach, we visualized the reconstructed images and presented a comparative analysis. (as Fig. 4 illustrates the reconstructed image for the first picture from jpeg-ai, with a close approximation of 0.17 bpp and an approximate compression ratio of 200:1. Details in the image indicate that our proposed model performs better in preserving fine details compared to the original image.

Future Work

While our current network architecture is based on a well-established framework, there is room for improving Network Architecture. Investigating different residual block structures or introducing skip connections can enhance the model's capability to capture complex image features efficiently.

We can also find a way to enhance Loss Function. Future work can focus on incorporating perceptual loss functions or utilizing perceptual metrics such as SSIM or PSNR can enhance the preservation of important visual features and improve the overall perceptual quality of reconstructed images.

While our current framework primarily focuses on image compression, extending the model to different data domains

is an intriguing avenue for future research.

Conclusion

We use a learned image compression approach using a discretized Gaussian mixture of likelihoods and attention modules. Our contribution is to utilize a simplified attention module with moderate complexity in our network architecture to achieve high coding efficiency.

Our model exhibits a more substantial reduction in test set loss, indicative of its superior ability to capture compression details. The results affirm that our approach consistently improves rate-distortion performance across various model capacities, showcasing its versatility and effectiveness. Visualization of the reconstructed images with a close approximation of 0.17 bpp and an approximate compression ratio of 200:1, further supports the superiority of our proposed model. Moreover, from the experimental results, our experiments surpass some previous ones. The superiority of our proposed model is clearly evident from the visualization of compressed images. Additionally, from the RD point table, our experiments also show good performance. However, compared to the latest work in the field of image compression, our designed model still has some shortcomings. We believe our approach contributes valuable insights to the field of image compression and opens avenues for further exploration and refinement of end-to-end approaches.

References

Agustsson, E.; Mentzer, F.; Tschannen, M.; Cavigelli, L.; Timofte, R.; Benini, L.; and Gool, L. V. 2017. Soft-to-hard vector quantization for end-to-end learning compressible representations. *Advances in neural information processing systems*, 30.

Ballé, J.; Laparra, V.; and Simoncelli, E. P. 2016. End-to-end optimized image compression. *arXiv preprint arXiv:1611.01704*.

Ballé, J.; Minnen, D.; Singh, S.; Hwang, S. J.; and Johnston, N. 2018. Variational image compression with a scale hyperprior. *arXiv preprint arXiv:1802.01436*.

Cheng, Z.; Sun, H.; Takeuchi, M.; and Katto, J. 2019. Deep Residual Learning for Image Compression. In *CVPR Workshops*, 0.

Cheng, Z.; Sun, H.; Takeuchi, M.; and Katto, J. 2020. Learned image compression with discretized gaussian mixture likelihoods and attention modules. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 7939–7948.

Cui, Z.; Wang, J.; Gao, S.; Guo, T.; Feng, Y.; and Bai, B. 2021. Asymmetric gained deep image compression with continuous rate adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10532–10541.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Goyal, V. K. 2001. Theoretical foundations of transform coding. *IEEE Signal Processing Magazine*, 18(5): 9–21.

Gregor, K.; Besse, F.; Jimenez Rezende, D.; Danihelka, I.; and Wierstra, D. 2016. Towards conceptual compression. *Advances In Neural Information Processing Systems*, 29.

He, D.; Zheng, Y.; Sun, B.; Wang, Y.; and Qin, H. 2021. Checkerboard context model for efficient learned image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14771–14780.

Lee, J.; Cho, S.; and Beack, S.-K. 2018. Context-adaptive entropy model for end-to-end optimized image compression. *arXiv preprint arXiv:1809.10452*.

Lee, J.-H.; Jeon, S.; Choi, K. P.; Park, Y.; and Kim, C.-S. 2022. DPICT: Deep progressive image compression using trit-planes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16113–16122.

Li, H.; Li, S.; Dai, W.; Li, C.; Zou, J.; and Xiong, H. 2023. Frequency-Aware Transformer for Learned Image Compression. *arXiv preprint arXiv:2310.16387*.

Liu, S.; Huang, Y.; Yang, H.; Liang, Y.; and Liu, W. 2022. End-to-end image compression method based on perception metric. *Signal, Image and Video Processing*, 16(7): 1803–1810.

Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; and Guo, B. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, 10012–10022.

Lu, Y.; Zhu, Y.; Yang, Y.; Said, A.; and Cohen, T. S. 2021. Progressive neural image compression with nested quantization and latent ordering. In *2021 IEEE International Conference on Image Processing (ICIP)*, 539–543. IEEE.

Luka, N.; Negrel, R.; and Picard, D. 2023. Image Compression using only Attention based Neural Networks. *arXiv preprint arXiv:2310.11265*.

Ma, H.; Liu, D.; Yan, N.; Li, H.; and Wu, F. 2020. End-to-end optimized versatile image compression with wavelet-like transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(3): 1247–1263.

Mentzer, F.; Agustsson, E.; Tschannen, M.; Timofte, R.; and Van Gool, L. 2018. Conditional probability models for deep image compression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4394–4402.

Minnen, D.; Ballé, J.; and Toderici, G. D. 2018. Joint autoregressive and hierarchical priors for learned image compression. *Advances in neural information processing systems*, 31.

Minnen, D.; and Singh, S. 2020. Channel-wise autoregressive entropy models for learned image compression. In *2020 IEEE International Conference on Image Processing (ICIP)*, 3339–3343. IEEE.

Ohm, J.-R. 2005. Advances in scalable video coding. *Proceedings of the IEEE*, 93(1): 42–56.

Skodras, A.; Christopoulos, C.; and Ebrahimi, T. 2001. The JPEG 2000 still image compression standard. *IEEE Signal processing magazine*, 18(5): 36–58.

Toderici, G.; O'Malley, S. M.; Hwang, S. J.; Vincent, D.; Minnen, D.; Baluja, S.; Covell, M.; and Sukthankar, R. 2015. Variable rate image compression with recurrent neural networks. *arXiv preprint arXiv:1511.06085*.

Wallace, G. K. 1991. The JPEG still picture compression standard. *Communications of the ACM*, 34(4): 30–44.

Wang, D.; Yang, W.; Hu, Y.; and Liu, J. 2022. Neural data-dependent transform for learned image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 17379–17388.

Yang, F.; Herranz, L.; Cheng, Y.; and Mozerov, M. G. 2021. Slimmable compressive autoencoders for practical neural image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4998–5007.

Zhu, Y.; Yang, Y.; and Cohen, T. 2021. Transformer-based transform coding. In *International Conference on Learning Representations*.