Adaptive Importance Channel Selection for Perceptual Image Compression

Yifan He, Feng Li, Huihui Bai and Yao Zhao*
Institute of Information Science, Beijing Jiaotong University
Beijing, 100044 - China
[e-mail :yifanhe@bjtu.edu.cn, l1feng@bjtu.edu.cn, hhbai@bjtu.edu.cn, yzhao@bjtu.edu.cn]
*Corresponding author: Yao Zhao

Received January 2, 2020; revised July 13, 2020; accepted August 27, 2020; published September 30, 2020

Abstract

Recently, auto-encoder has emerged as the most popular method in convolutional neural network (CNN) based image compression and has achieved impressive performance. In the traditional auto-encoder based image compression model, the encoder simply sends the features of last layer to the decoder, which cannot allocate bits over different spatial regions in an efficient way. Besides, these methods do not fully exploit the contextual information under different receptive fields for better reconstruction performance. In this paper, to solve these issues, a novel auto-encoder model is designed for image compression, which can effectively transmit the hierarchical features of the encoder to the decoder. Specifically, we first propose an adaptive bit-allocation strategy, which can adaptively select an importance channel. Then, we conduct the multiply operation on the generated importance mask and the features of the last layer in our proposed encoder to achieve efficient bit allocation. Moreover, we present an additional novel perceptual loss function for more accurate image details. Extensive experiments demonstrated that the proposed model can achieve significant superiority compared with JPEG and JPEG2000 both in both subjective and objective quality. Besides, our model shows better performance than the state-of-the-art convolutional neural network (CNN)-based image compression methods in terms of PSNR.

Keywords: Image compression, Auto-encoder, Perceptual loss, Bit-allocate strategy, Importance map

This work was supported in part by National Natural Science Foundation of China (No. 61972023) and Fundamental Research Funds for the Central Universities (2019JBZ102, 2019YJS031, 2018JBZ001).

http://doi.org/10.3837/tiis.2020.09.014 **ISSN : 1976-7277**

1. Introduction

In the past 20 years, the digital media technology has achieved great progress. Considering that everyone can share their daily life by taking pictures and videos with their friends on Internet, and the resolution of pictures and videos taken by mobile phones are increasing year by year, the storage of data is increasing with enormous rate. Due to the limitation of data transmission technology, image compression has become a key technique for various multimedia transmission services. Image compression typically starts from obtaining a description of an image, then quantifying the description, and recovering the image from the obtained description. A general image compression system mainly includes three components, *i.e.* encoder, quantizer, and decoder, to form a codec. The image compression codec in typical encoding standards, such as JPEG [1], JPEG2000 [2], and BPG [3] using the intra-coded HEVC [4], rely on the hand-crafted image transformation and separated optimization on codecs, which is not optimal for compression performance.

An image compression system needs to deal with quantization, and to control the trade-off between reconstruction error d and the bitrates R. To minimizing $d + \lambda R$, there are two directions. On the one hand, the rate term, which is determined by the entropy H of the latent image representation, can be optimized by an exact entropy rate estimator. On the other hand, the distortion term, which measures difference between the input image of encoder and the reconstructed image by decoder, can be minimized by designing better encoder and decoder.

Recently, inspired by the powerful learning ability of deep convolutional neural networks(CNNs) in image restoration tasks [5,6], many methods [7-11] adopt CNNs to form different frameworks for lossless image compression [7] or lossy image compression [8-11], which have achieved significant improvement than many traditional image compression codecs. Mentzer *et al.* [7] propose a practical lossless image compression framework by learning-based method, named L3C, which introduces a fully parallelizable hierarchical probabilistic model for entropy coding, which can be optimized by an end-to-end way. This pioneer method shows significant superiority compared with many popular engineered codecs, such as PNG, WebP and JPEG2000.

Besides the lossless image compressiom, there are some CNN-based methods [8-10] that have been proposed to work on learned lossy compression. In [8], Ballé *et al.* propose an end-to-end image lossy compression network, which consists of a nonlinear analysis transformation, a uniform quantizer, and a nonlinear synthesis transformation, which produce nearly better compression performance than the standard JPEG and JPEG2000 method. However, such method treats the transmitted features equally and directly feeds the output of encoder to the corresponding decoder, which can not focus on the important information across spatial locations under limited bits for effective image compression. In [9], Mentzer *et al.* focus on the rate-distortion (R-D) of the latent image representation and presents a conditional probability model to optimize the R-D trade-off. The authors formulate a spatial-aware network, which can use an importance map to help the network spatially attend to the most important regions of the image with different numbers of bits. However, this approach chooses the fixed first feature map of the last layer in encoder as importance map, which can not adaptively emphasize informative spatial regions for various input. In [10], motivated by that the information is highly variant in different areas of an image, Li *et al.*

develop a CNN-based end-to-end system for content weighted image compression, which can allocate the content-aware bits under the guidance of a content-weighted importance map. The importance map can be produced by an convolutional neural network and the sum of the importance map can serve as a continuous alternative of discrate entropy estimation to control compression rate.

However, such auto-encoder based image compression methods regard the image compression of different bit rates as indenpent tasks, which will face the challenge of large storages. To address this issue, in [11], Toderici *et al* proposes a LSTM recurrent network for variable-rate image compression, which can provide variable compression rates during deployment without requiring retraining of the network. Although this method could provide variable compression rates in an model, it has not any bit-allocation strategy and leads to less accurate reconstruction. In addition, the framework in [8,9,10] can not make fully use of the hierarchical features extracted from the encoder, which cannot produce a reconstructed image with better quality and detial.

In this paper, to solve the problems mentioned above, we implement a novel auto-encoder framework for learned lossy image. Instead of directly sending the features of last layers in encoder to the correponsding decoder, to explore the features under different receptive fields, we aggregate the features from each downsampling layer of our encoder to obtain more accurate feature representation. Besides, different from choosing the first channel as the importance map in [9], we put forward an adaptive important channel selection stragety by comparing the sum of each channel. And then the multiply operation is conducted on the importance mask from the selective channel and the last layer features of the encoder to achieve efficient bit allocation. Furthermore, previous deep learning-based image compression models simply minimize the mean square error (MSE) between the reconstructed image and original input, which will generate overly-smooth compressed results. Therefore, we present an additional novel perceptual loss function and combine it with reconstruction loss to optimize our network, which can produce the compressed image with visual pleasant details.

Our main contributions are summarized as:

- To utilize the hierarchical features extracted by encoder, a novel auto-encoder framework is proposed, which transmits the hierarchical features from encoder to decoder to reconstruct images with better quality.
- We proposed an adaptive important channel selection stragety to achieve efficient bit allocation.
- The preceptual loss is generated by the proposed encoder rather than extra pre-train network to improve visual details.
- The extensive experiments on KodakCD image datasets demonstrate that the proposed method performs favorably against the state-of-the-art compression approachs in terms of PSNR.

The remainder of this paper is organized as follows. In Section 2, the related work is introduced. The proposed method is presented in Section 3, including hierarchical auto-encoder framework, adaptive importance map and encoder preceptual loss. The experimental results and comparisons with other method are demonstrated in Section 4. The conclusion of this paper is presented in Senction 5.

2. Related Work

In early years, the auto-encoder model was first proposed by Hinton to address unsupervised learning problems [12]. Traditionally, it is composed of two or three layers of a neural network and applied BP (back propagation) [13] technique to learn nonlinear transformation for compressing and reconstructing the input data. It aims at learning an identity equation:

$$\mathrm{D}\big(E(x)\big) = \tilde{x} \tag{1}$$

which makes the output approximately equals to the input.

Therefore, the auto-encoder is perfectly suitable for the data compression task, and there are many works [8,14,15] that use traditional auto-encoder model to compress images and have made great progress. On the other hand, other works [9,10,16] modify the traditional auto-encoder structure or propose new network structures for the better compression quality.

2.1 Bit allocation strategy

Since the conventional encoder assigns the same number of bit symbols for each spatial areas of the original image. However, in practice, image information in different spatial locations are highly variable. The importance map tries to allocate automatically lower symbols for the smooth information regions (e.g., the cloud) and higher symbols for the complicate regions (e.g., the house with exquisite and complicated pattern). In [9] an importance mask is added in the latter of encoder last layer for spatial bit allocation, which is produced from the first feature map of the encoder last layer. In [10] the content-aware convolutional neural network is used to learn an importance map to achieve different bit rates.

2.2 Multi-scale structure for image compression

In [16] a new auto-encoder structure is presented that exploits the multi-scale features of input images. The proposed model consists of two components: a multi-scale lossy auto-encoder and a multi-scale lossless coder for entropy coding. The lossy auto-encoder model directly connects the encoder and decoder at different depths to encode multi-scale image features. Then, the encoder sends the part of each layer to the corresponding layer in decoder. The lossless coder simultaneously encodes the quantized multi-scale features to produce transmitted symbols for decrease the time of encoding.

2.3 Variable compression rate

In deep learning-based image compression methods, there is a problem that is how to compress an image at different bit rates. Several options have been explored including training multiple modes [8], learning quantization-scaling parameters [14], and transmitting a subset of the encoded representation with a recurrent structure [11,17].

The architecture of [11] consists of encoder and decoder based on recurrent neural network (RNN), a binarizer, and a neural network to model the distribution of latent variables. It solves the problem of variable compression rate from two aspects, that is, designing a residual encoder with powerful ability of feature extraction, and designing a probability estimation model for capture long-term dependencies between the patches of input image. In [17] three improvements over previous research are introduced. First, a new recurrent architecture is proposed, which makes the image compression network models and propagates spatial information more effectively between the network's hidden layers. Second, besides lossless entropy coding, a bit allocation algorithm is adopted to adequately exploit the limited number

of bits in complex image regions. Finally, the results demonstrate that training with the combination of pixel-wise loss and structural similarity (SSIM) can improve the compression performance according to multiple metrics. These RNN-based methods provide a way to solve the problem of variable compression rate. However, the RNN-based methods are less accurate reconstruction in each compression rates.

2.4 Generative compression

In [18] the concept of generative compression is described as the compression of data using generative models. For model generative image, they use the variational auto-encoders [19] to alternate Generative Adversarial Networks (GANs). Their results show that the method of generative compression is more resilient to bit error rates than traditional image compression methods at very low bitrates. However, their model has merely proved the effectiveness of generative compression in small images below 64×64, and has limited effects on larger images.

In [20] a new GAN-based network for extreme learned image compression is proposed, which aims at full-resolution images, targeting bitrates below 0.1 bpp and obtaining visually pleasing images at significantly lower bitrates than previous methods. The proposed method consists of unconditional and conditional GANS. The unconditional GANs can generate the overall image content with lower image quality, and the conditional GANs can utilize the corresponding semantic label map to reconstruct the parts of the image with better detail. Their results show that for extreme low bitrates, the proposed method can reconstruct the original image with better visual quality.

3. Proposed Method

In this section, the proposed network architecture is firstly introduced. And then we describe each of three main techniques used in our model: adaptive importance map, multi-scale auto-encoder, and encoder perceptual loss.

3.1 Overview

Given an original input image $x \in R^{H \times W \times C}$, we wish to design an image compression system to compress the image as small as possible and make the restored image same as the original image. In an image compression system, the procedure of obtaining the compressed bitstream of input can be described as follows.

$$s = H_e\{Q[E(x;\varphi)]\} \tag{2}$$

where $E(x;\varphi) : R^d \rightarrow R^m$ represents the encoder, which maps the input to a latent representation $z = E(x;\varphi)$. The quantizer $Q: R \rightarrow B$ discretizes the coordinates of z to $N = |B|$ centers, obtaining $\hat{z}$ with $\hat{z}_i = Q(z_i) \in B$, which have limited value numbers and can be losslessly encoded into a bitstream s by an entropy encoder $H_e(\cdot) \in (0,1)$. When the decoder receives the bitstreams, the process of the decoder restoring the final image can be formulated as

$$\tilde{x} = D[H_d(s);\theta] \tag{3}$$

Here, $\tilde{x}$ is the corresponding reconstructed image from compressed binary symbols. The decoder $D(x;\theta)$ forms the reconstructed image $\tilde{x}$ from the quantized latent representation $\hat{z}$, which is in turn losslessly decoded from the bitstream by entropy decoder $H_d(\cdot) \in B$.

3.2 Hierarchical Auto-encoder Structure

In this subsection, we describe our proposed hierarchical structure of the encoder-decoder. As shown in **Fig. 1**, the proposed network is composed of four parts: an encoder-decoder for hierarchical features extraction, a bit allocation module, a quantizer and an entropy encoder-decoder. The encoder takes an image as input to produce four outputs $f_k^{H_k \times W_k \times N} (k = 1,2,3,4)$ with different scales. Next, at $k = 1,2,3$, three 1-channel convolutional layers are employed to make these outputs to 1 channel. These outputs are downsampled to the same size as $f_4^{H_4 \times W_4 \times N}$. This features are concatenated together $z = [\tilde{f}_1^{H_4 \times W_4 \times 1}, \tilde{f}_2^{H_4 \times W_4 \times 1}, \tilde{f}_3^{H_4 \times W_4 \times 1}, f_4^{H_4 \times W_4 \times N}]$. After that, the concatenated hierarchical features z are sent to the bit allocation model for an importance mask m. We conduct the multiply operation on m and z to achieve efficient bit allocation.

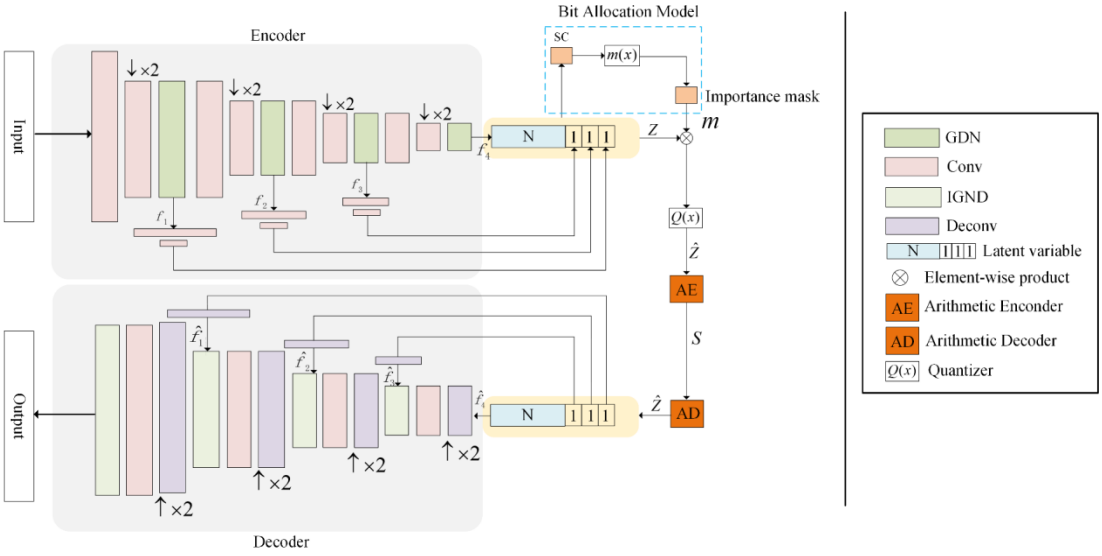


Fig. 1. The proposed multi-scale auto-encoder architecture for image compression.

$$\tilde{z} = z \otimes m \tag{4}$$

where $z, m \in R^{H_4 \times W_4 \times (N+3)}$, $m \in [0,1]$. Then, the generated feature $\tilde{z}$ is quantized and arithmetic encoded (AE) to get s.

When binary symbols are transmitted to the decoder, the arithmetic decoder (AD) firstly decodes it. Furthermore, the first, second and third channel of the decoded features are separately upsampled to the corresponding decoder layers size to get the decoder inputs with different scales $(\hat{f}_1^{H_1 \times W_1 \times 1}, \hat{f}_2^{H_2 \times W_2 \times 1}, \hat{f}_3^{H_3 \times W_3 \times 1}, \hat{f}_4^{H_4 \times W_4 \times N})$. To reconstruct the original image, the rest channels of $\hat{z}$ are directly sent to the last layer of our decoder. At the same time, the upsampled features are concatenated to the corresponding decoder layers respectively to provide multi-scale information.

$$l_D^3 = l_D^4(\hat{f}_4^{H_4 \times W_4 \times N}) \tag{5}$$

$$\tilde{x} = l_D^s(concat(l_D^{s+1}, \hat{f}_s^{H_s \times W_s \times 1})) \quad s = (0,1,2) \tag{6}$$

where l_D^s represents the *s-th* layer output of decoder. $\tilde{x}$ is the final reconstructed image. $concat(\cdot)$ denotes the concatenation operation.

3.3 Adaptive Importance Map

In previous method [8], the importance mask is added at the end of the encoder for spatial bit-allocation. The authors choose the first feature map from the final output features generated by the encoder as importance map, which can not adaptively emphasize informative spatial regions for various input.

For the features produced by the encoder, we consider that the large number of high values can cause more bits allocation during transmission. Since that the feature maps produced by the convolutional kernels have different values, the feature map containing a lot of large values will have big gaps with others including the small values. In the entropy coding stage, the feature map with a lot of high values can consume more bits, which shows this feature map contains more information than others. At the same time, it is crucial to choose a feature map which contains aboundant information as importance map for more effective bits alloction. As a result, in our network, we select the feature map with the largest sum of all values within itself as our importance map.

The process of choosing importance map can be describe as follows. Given an input image *x*, which have $H \times W \times 3$ scales, the encoder E has three strides-2 convolution layers and bottleneck z has C channels. The dimension of z and $\hat{z}$ will be $\frac{H}{8} \times \frac{W}{8} \times C$.

We choose adaptively the importance map, which is the *n-th* channel of encoder last layer with largest values after summing.

$$n = argmax_{\{k\}} \sum_{i,j} f_{i,j,k} \tag{7}$$

Actually, the range of the importance map values f_n cannot be used directly in produce importance mask. We need to make a transformation for the range of f_n values, that is:

$$\tilde{f_n} = C \times sigmoid(f_n) \;\; \tilde{f_n} \in (0, C) \tag{8}$$

Finally, the importance map size is $\frac{H}{8} \times \frac{W}{8} \times 1$ and should be expanded into mask m which have same size with z by a simple function:

$$\mathrm{m}_{i,j,k} = \begin{cases} 1 \;\; , & if \; k < \tilde{f}_{i,j,n} \\ \left(\tilde{f}_{i,j,n} - k\right), & if \; k \le \tilde{f}_{i,j,n} \le k+1 \\ 0 \;\; , & if \; k+1 > \tilde{f}_{i,j,n} \end{cases} \tag{9}$$

where $\tilde{f}_{i,j,n}$ means the values of importance map at spatial location (i, j), and k denotes the index of mask m channel.

3.4 Encoding Perceptual Loss

Nowadays many computation perception algorithm [21,22,23] are proposed. The perceptual loss shows that the visually high-quality images can be generated by defining and optimizing perceptual loss function based on high-level features.

The traditional loss function is to calculate the pixel-level distance between ground-truth image and generated image, which makes each pixel of the generated image as similar as possible to original image. If the generated image has few pixel offsets from the original image, the pixel-level loss function will show highly discrepancy, whereas the generated image visually is very similar to original image.

Considering that people can't find the slight pixel offsets between two different images, the perceptual loss based on high-level features can better evaluate image visual similarity between two images. The perceptual loss typically needs a pre-trained network to extract high-level features, and the pre-trained network often chooses the VGG-Net trained in ImageNet dataset. That is:

$$d_{feat}^{l}(x,\tilde{x}) = \frac{1}{C_l W_l H_l}\|VGG_l(x) - VGG_l(\tilde{x})\|_2^2 \tag{10}$$

where l is the *l-th* layer of VGG-Net, the *l-th* output is a feature map of shape $C_l \times W_l \times H_l$. However, in image or video compression task, the auto-encoder structure has been restricted the depth of encoder and decoder, because the structure of encoder and decoder should be generally mirror symmetrical. If the number of encoder layers is increased by *N* layers, the total numbers of auto-encoder layers will increase by *2N* layers. Therefore, in the case of memory limitation, the addition of the pre-trained model will make the depth of encoder and decoder become shallower, which lead to the degradation on the reconstructed image quality.

Instead of using pre-trained model, we use the encoder to get perceptual loss. In our proposed auto-encoder framework, the encoder can extract the features from input, which can be used to compute perceptual loss, that is:

$$d_{feat}(x,\tilde{x}) = \frac{1}{CWH}\|E(x) - E(\tilde{x})\|_2^2 \tag{11}$$

where the output of encoder is feature maps of shape $H \times W \times C$, and $E(x)$ represent the encoder final outputs when input an image x. $\tilde{x}$ is the reconstruction of x in decoder.

3.5 Loss Function

In this section, we describe the loss function used in our model in training step. Optimizing the trade-off between image reconstruction distortion and the bit rates in image compression is the permanent theme. We adopt it as a part of our loss function to learn compression and reconstruction of an image.

However, the section 3.4 has analyzed the disadvantage of pixel-level loss function, that is, the traditional distortion MSE will make the decoder reconstruct over smooth image. Therefore, we propose a new perceptual loss as another part of distortion term to enhance the detail of the reconstructed image. Suppose that mini-batch input image is $x = \{x^{(1)}, x^{(2)}, \dots, x^{(B)}\}$ and the masked outputs of encoder are $\tilde{z} = \{\tilde{z}^{(1)}, \tilde{z}^{(2)}, \dots, \tilde{z}^{(B)}\}$, our object function can be described as follows.

$$\mathrm{L} = \frac{1}{B}\sum_b \left(\lambda_2 \left(d\left(x^{(b)}, \tilde{x}^{(b)}\right) + \lambda_1 d_{feat}\left(x^{(b)}, \tilde{x}^{(b)}\right)\right) + L_R\left(Q\left(\tilde{z}^{(b)}\right)\right)\right) \tag{12}$$

where L_R is the rate loss, which describes the entropy of compressed image. In our model, we adopt MSE (Mean Squared Error) as the $d(\cdot,\cdot)$, and the $d_{feat}(\cdot,\cdot)$ is the encoder perceptual loss described in section 3.4.

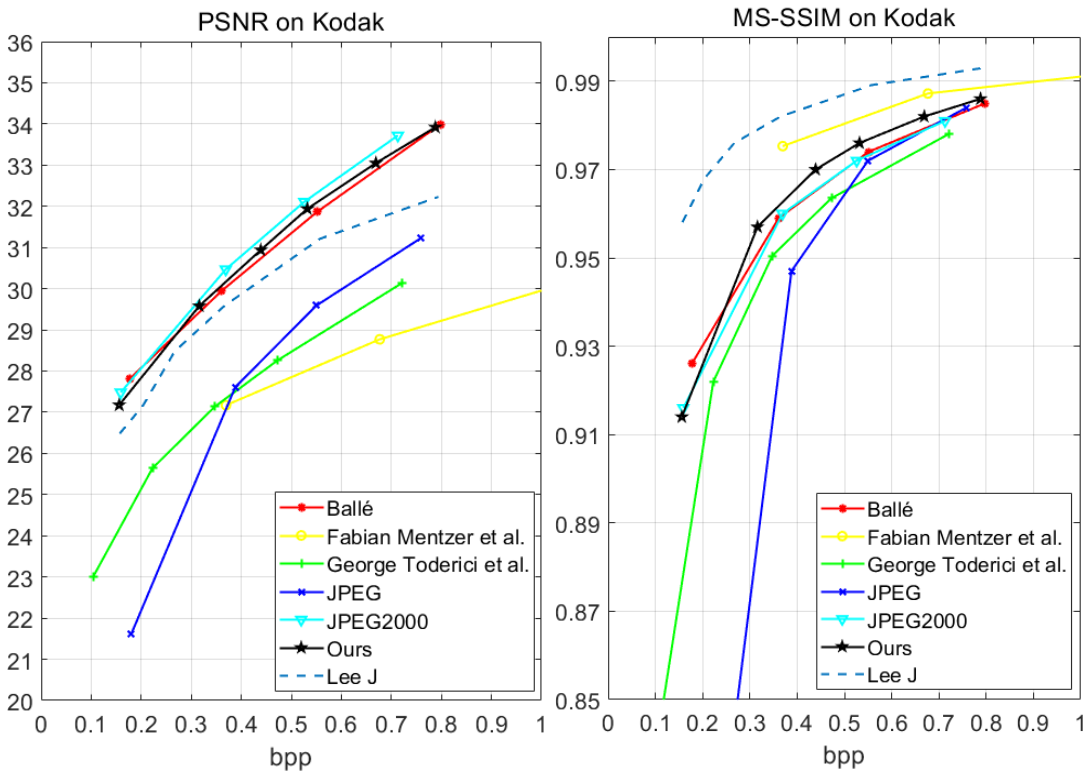


Fig. 2. Comparison of the rate-distortion curves by different methods: (left) PSNR, (right) MS-SSIM.

4. Experiments

Our hierarchical auto-encoder image compression models are trained on the subset of ImageNet [24], which includes 33,600 images with a size larger than 128×128. During training, these images are cropped into 128×128 patches and feed these patches to our network as original inputs. After training, we conduct experiments to evaluate the performance of our network for image compression task on the Kodak PhotoCD [25] image dataset, which consists of 24 natural images with size 512×768 or 768×512.

4.1 Parameter Setting

In our experiments, we set the number of convolutional kernel output channels n according to the bitrates, i.e. 128, when the bitrate is lower than 0.5 bpp and 192 otherwise. Then, different values of the trade-off parameter λ_2 in the range [0.001, 0.02] are chosen to get different bitrates. The encoder perceptual loss term λ_1 is set to 10 and other network parameters have shown in **Fig. 1**.

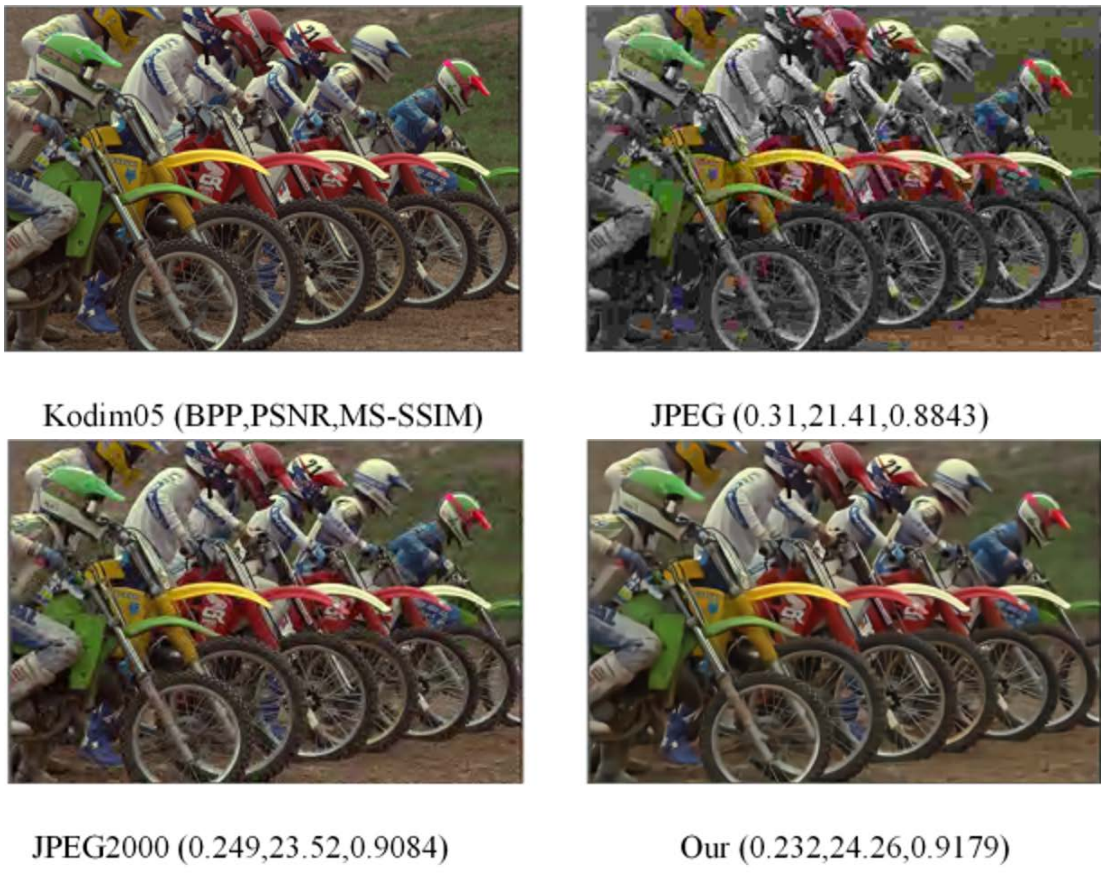
The generalized divisive normalization (GDN) is chosen as our activation function in encoder and the inverse generalized divisive normalization (IGDN) used in decoder, which are proposed in [26]. In the stage of entropy coding, the method of model the probability distribution of latent variable representation is the same as that proposed in [27]. During training, in order to backpropagate gradient through the non-differentiable quantizer, we add a uniform noise to latent representation for replacing the quantizer, as in [8].

4.2 Performance Evaluation

We compared the performance of our proposed methods with existing image compression standard formats, JPEG, JPEG2000, and state-of-the-art CNNs-based image compression methods. In this paper, image distortion is evaluated by Multi-Scale Structure Similarity (MS-SSIM) [28] and Peak Signal-to-Noise Ratio (PSNR), while compression ratio is evaluated by bits per pixel (bpp), which calculates as the number of bits used to code the original image divided by the number of pixels.

Fig. 2 shows the R-D curves with different compression methods on Kodak dataset. In terms of MS-SSIM, our proposed method has achieved superior performance to the existing image compression standard formats (JPEG; JPEG2000) and deep learning-based methods ([8,11]). Moreover, when PSNR is used to evaluation, these deep learning-based methods ([9,11,29]) have poor performance, but our model still keep the performance in a good level.

Finally, we provide subjective comparisons between our compression results and other results of popular codecs in **Fig. 3**. Because each of the codecs can only compress an image to a coarse-level output bit rate, when compared with other codecs, we choose the bitrates of other codecs that is same or larger than the bitrates produced by our model, which purpose is to give other image compression methods an advantage in term of bitrates. In **Fig. 3**, These results indicate that the images compressed by standard compression methods usually perform well when evaluated with PSNR, but perform poorly when evaluated with MS-SSIM. Our model enables the compressed images to perform better when evaluated with PSNR and MS-SSIM.



Kodim07 (BPP,PSNR,MS-SSIM)

JPEG (0.17,21.87,0.8269)

JPEG2000 (0.146,28.23,0.9494)

Our (0.135,28.43,0.9529)

Kodim08 (BPP,PSNR,MS-SSIM)

JPEG (0.30,20.63,0.8849)

JPEG2000 (0.279,23.33,0.9224)

Our (0.26,23.70,0.9332)

Kodim14 (BPP,PSNR,MS-SSIM)

JPEG (0.175,20.74,0.7360)

JPEG2000 (0.182,25.36,0.8859)

Our (0.166,25.16,0.8911)

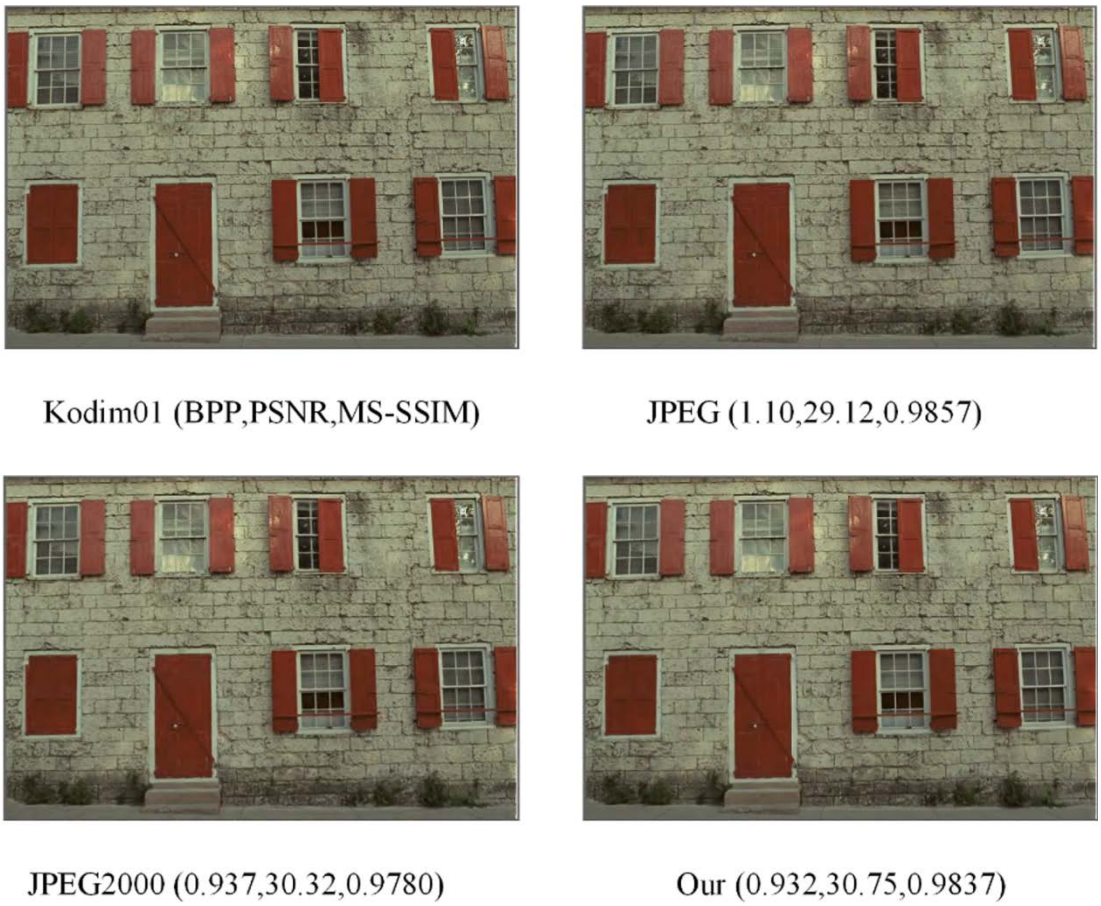


Fig. 3. Image producted by different compression systems at different compression rate. From the left to right: groundtruth, JPEG, JPEG2000 and Ours.

4.3 Ablation Study

4.3.1 Adaptive importance map

As described in detail in Section 3.3, we adaptively choose an important map to dynamically adjust the bit allocation of different channels features used for encoding spatial locations of an image effectively. To prove the advantage of the adaptive selection model, we trained three auto-encoder M, $M_I^1$and M_I^*, where M_I^* choose the largest feature map of last layer in encoder as an importance map, M_I^1 uses the first feature map as importance map, and M has not bit-allocation model. During training, M, M_I^1 and M_I^* have set $N = 128$ and trained with the same iteration. In **Table 1**, it shows that the MS-SSIM and PSNR results evaluated in Kodak PhotoCD image dataset.

These results mean that, no matter which channel is chosen as the importance map, the addition of importance map can improve the compression model performance. At the same time, our strategy of adaptive choosing importance map can best boost the MS-SSIM from 0.9593 to 0.9608 and PSNR from 29.96 to 30.18 in model M.

Table 1. Importance Channel Selection Experiments

Metric	M	M_I^1	M_I^*
PSNR	29.96	29.99	30.18
MS-SSIM	0.9593	0.9605	0.9608
BPP	0.361	0.361	0.361

Furthermore, **Fig. 4** shows the visualization of all channels of the latent representation for *M*, which displays that the information discrepancy between different importance maps. In these feature maps, the 26th channel is the largest feature map and has been upsampled for better observation, which obviously has more semantic information than others.

Fig. 4 shows the visualization of all channels of the latent representation for M, which displays that the information discrepancy between different importance maps. In these feature maps, the 26th channel obviously has more semantic information than others, which will be selected as the importance channel. As the result, the proposed channel selection strategy has advantage to represent semantic information. Furthermore, the selective important channel comes directly from the features of encoder last layer, and does not need additional convolutional operator. If extra semantic segmentation network is introduced, it may lead to more semantic information, but it will increase computation complexity and more network parameters.

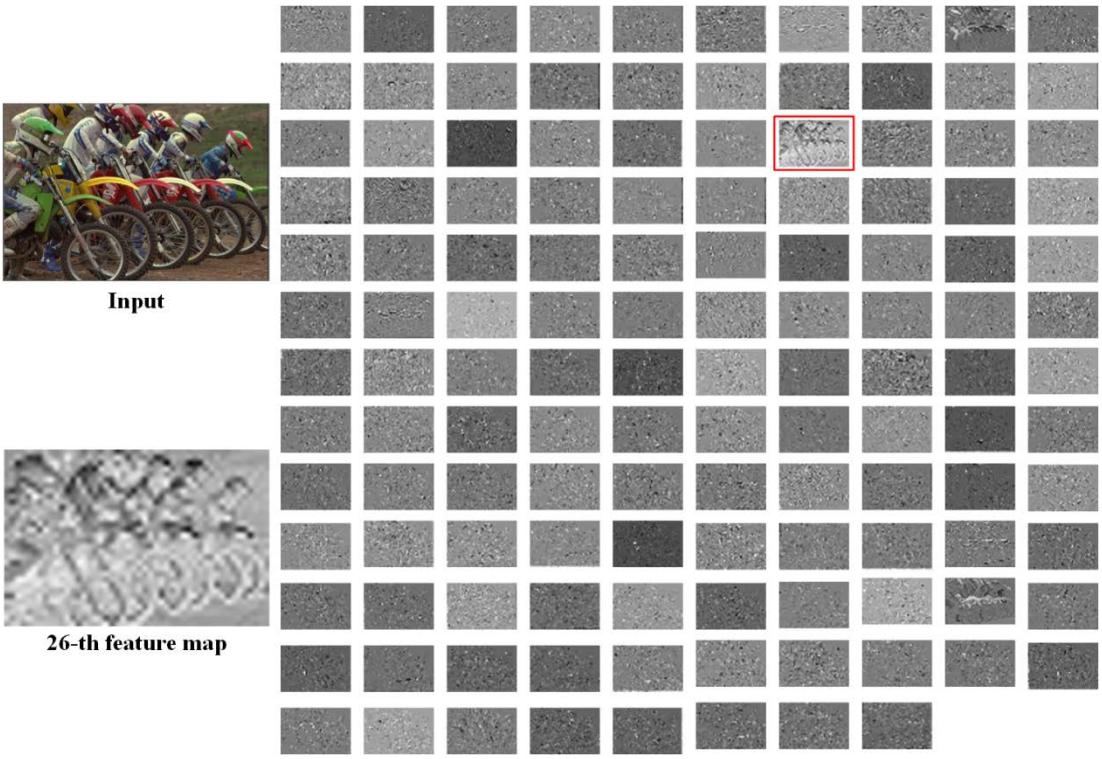


Fig. 4. Visualization of the latent representation in model M at a median-bpp operating point

4.3.2 Encoder perceptual loss

The detail of encoder perceptual loss is described in Section 3.4. The purpose of choosing encoder perceptual loss as a feature-level constrain term in our loss function is that, it can make our model pay attention to the detail of reconstructed image and does not require extra memory beside auto-encoder structure during training. We trained two auto-encoder M and M_p, where M only chooses MSE as the distortion term and M_p selects the combination of MSE and encoder perceptual loss. The entropy rate terms of objective function are same in M and M_p.

Fig. 5 shows this combination can guide our image compression model to reconstruct an image with better quality. It is noted that the proposed encoder perceptual loss does not require additional pre-training network to obtain high-level features, which reduces the load of GPU during the training stage. Considering that the perceptual loss needs encoder to extract high-level features, it cannot be used as a network loss alone. Therefore, we adopt joint loss function of perceptual loss and MSE loss as our reconstruction loss term.

Fig. 5. PSNR and MS-SSIM comparison between the model M and the model M_p

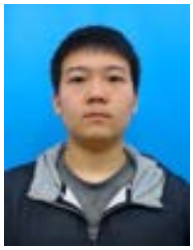
5. Conclusion

In this paper, we introduced three techniques: adaptive importance channel, multi-scale auto-encoder network, and encoder perceptual loss. Our experiments show that these techniques boost our performance. The proposed method of adaptive importance channel enables our model with the ability to allocate bits and improves our model's performance on MS-SSIM and PSNR. Training with encoder perceptual loss and multi-scale auto-encoder structure provide further improvements to reconstruct perceptual structures, such as sharp edges and details textures. Additionally, our methods are a worthy choice for other auto-encoder compression networks to boost their performance.

References

[1] G. K. Wallace, "The jpeg still picture compression standard," *IEEE transactions on consumer electronics*, 38(1), xviii- xxxiv, 1992. Article (CrossRef Link)
[2] A. Skodras, C. Christopoulos, and T. Ebrahimi, "The jpeg 2000 still image compression standard," *IEEE Signal processing magazine*, 18(5), 36–58, 2001. Article (CrossRef Link)
[3] F. Bellard, "BPG Image Format," 2014. Accessed: 2017-01-30. Article (CrossRef Link)
[4] J. Lainema, F. Bossen, W.-J. Han, J. Min, and K. Ugur, "Intra coding of the HEVC standard," *IEEE Trans. Circuits Syst. Video Technol*., vol. 22, no. 12, pp. 1792–1801, Dec. 2012. Article (CrossRef Link)
[5] F. Li, H. Bai, Y. Zhao, "FilterNet: Adaptive Information Filtering Network for Accurate and Fast Image Super-Resolution," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 6, pp. 1511-1523, 2019. Article (CrossRef Link)
[6] J. Kim, J. Kwon Lee, K. Mu Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proc. of the IEEE conference on computer vision and pattern recognition*, 1646-1654, 2016. Article (CrossRef Link)
[7] F. Mentzer, E. Agustsson, M. Tschannen, et al., "Practical full resolution learned lossless image compression," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 10629-10638, 2019. Article (CrossRef Link)
[8] J. Ballé, V. Laparra, and E. P. Simoncelli, "End-to-end optimized image compression," in *Proc. of International Conference on Learning Representations*, 2017. Article (CrossRef Link)
[9] F. Mentzer, E. Agustsson, M. Tschannen, et al., "Conditional probability models for deep image compression," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 4394-4402, 2018. Article (CrossRef Link)
[10] M. Li, W. Zuo, S. Gu, et al., "Learning convolutional networks for content-weighted image compression," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 3214-3223, 2018. Article (CrossRef Link)
[11] G. Toderici, D. Vincent, N. Johnston, et al., "Full resolution image compression with recurrent neural networks," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 5306-5314, 2017. Article (CrossRef Link)
[12] D. E. Rumelhart, G. E. Hinton, R. J. Williams. "Learning internal representations by error propagation," *California Univ San Diego La Jolla Inst for Cognitive Science*, 1985. Article (CrossRef Link)
[13] Y. LeCun, B. E. Boser, J. S. Denker, et al., "Handwritten digit recognition with a back-propagation network," in *Proc. of Advances in neural information processing systems*, 396-404, 1990. Article (CrossRef Link)
[14] L Theis, W Shi, A Cunningham, F Huszár, "Lossy image compression with compressive autoencoders," in *Proc. of International Conference on Learning Representations*, 2017. Article (CrossRef Link)

[15] T. Dumas, A. Roumy, C. Guillemot. "Autoencoder based image compression: can the learning be quantization independent?," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 1188-1192, 2018. Article (CrossRef Link)
[16] K. M. Nakanishi, S. Maeda, T. Miyato, et al., "Neural multi-scale image compression," in *Proc. of Asian Conference on Computer Vision. Springer, Cham*, 718-732, 2018. Article (CrossRef Link)
[17] N. Johnston, D. Vincent, D. Minnen, et al., "Improved lossy image compression with priming and spatially adaptive bit rates for recurrent networks," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 4385-4393, 2018. Article (CrossRef Link)
[18] S. Santurkar, D. Budden, N. Shavit. "Generative compression," in *Proc. of Picture Coding Symposium. IEEE*, 258-262, 2018. Article (CrossRef Link)
[19] D. P. Kingma, M. Welling. "Auto-encoding variational bayes," in *Proc. of International Conference on Learning Representations*, 2014. Article (CrossRef Link)
[20] E. Agustsson, M. Tschannen, F. Mentzer, et al., "Generative adversarial networks for extreme learned image compression," in *Proc. of the IEEE International Conference on Computer Vision*, 221-231, 2019. Article (CrossRef Link)
[21] J. Johnson, A. Alahi, L. Fei-Fei. "Perceptual losses for real-time style transfer and super-resolution," in *Proc. of European conference on computer vision. Springer, Cham*, 694-711, 2016. Article (CrossRef Link)
[22] M Jian, KM Lam, J Dong, et al., "Visual-patch-attention-aware Saliency Detection," *IEEE Transactions on Cybernetics*, Vol. 45, No. 8, pp. 1575-1586, 2015. Article (CrossRef Link)
[23] M Jian, W zhang, Y Hui, et al., "Saliency detection based on directional patches extraction and principal local color contrast," *Journal of Visual Communication and Image Representation*, Vol. 57, pp. 1-11, 2018. Article (CrossRef Link)
[24] J. Deng, W. Dong, R. Socher, et al., "Imagenet: A large-scale hierarchical image database," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 248-255, 2019. Article (CrossRef Link)
[25] Kodak PhotoCD dataset. Article (CrossRef Link)
[26] J. Ballé. "Efficient nonlinear transforms for lossy image compression," in *Proc. of Picture Coding Symposium. IEEE*, 248-252, 2018. Article (CrossRef Link)
[27] J. Ballé, D Minnen, S Singh, et al., "Variational image compression with a scale hyperprior," in *Proc. of International Conference on Learning Representations*, 2018. Article (CrossRef Link)
[28] Z. Wang, E. P. Simoncelli, A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *Proc. of The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers*, 2, 1398-1402, 2003. Article (CrossRef Link)
[29] J. Lee, S. Cho, S. K. Beack. "Context-adaptive entropy model for end-to-end optimized image compression," in *Proc. of International Conference on Learning Representations*, 2019. Article (CrossRef Link)



Yifan He received the B.S. degree in Communication Engineering in 2018 from the School of Automation and Information Engineering, Xi'an University of Technology. He is currently pursuing the M.S. degree in the Institute of Information Science, Beijing Jiaotong University. He works in image lossy compression and deep learning.

Feng Li received his B.S. degree in Anhui Normal University, China, in 2012. Now, he is pursuing his Ph. D degree in Institute of Information Science, Beijing Jiaotong University, Beijing, China. His research interests are image and video compression, image and video super resolution, computer vision and deep learning.

Huihui Bai received her B.S. degree from Beijing Jiaotong University, China, in 2001, and her Ph.D. degree from Beijing Jiaotong University, China, in 2008. She is currently a professor in Beijing Jiaotong University. She has been engaged in R&D work in video coding technologies and standards, such as HEVC, 3D video compression, multiple description video coding (MDC), and distributed video coding (DVC).

Yao Zhao received the B.S. degree from Fuzhou University, China, in 1989, and the ME degree from Southeast University, Nanjing, China, in 1992, both from the Radio Engineering Department, and the PhD degree from the Institute of Information Science, Beijing Jiaotong University (BJTU), China, in 1996. He became an associate professor at BJTU in 1998 and became a professor in 2001. From 2001 to 2002, he was a senior research fellow with the Information and Communication Theory Group, Faculty of Information Technology and Systems, Delft University of Technology, Delft, The Netherlands. He is currently the director of the Institute of Information Science, BJTU. His current research interests include image/video coding, digital watermarking and forensics, and video analysis and understanding. He serves on the editorial boards of several international journals, including as associate editors of IEEE Transactions on Cybernetics, IEEE Signal Processing Letters, and an area editor of Signal Processing: Image Communication (Elsevier), etc. He was named a distinguished young scholar by the National Science Foundation of China in 2010, and was elected as a Chang Jiang Scholar of Ministry of Education of China in 2013. He is a senior member of the IEEE.