Research on Anti-noise of Image Compression Based on Variational Autoencoder

Li Gengzeng *
School of information and communication engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China
Corresponding author’s e-mail: ligengzeng@bupt.edu.cn

Abstract. We introduced the Variational Compression Autoencoder (VCAE) that introduces noise training, and uses KL divergence to balance compression and anti-noise. Variational autoencoders improve the anti-noise performance of the compression framework through noise training and learning decoupled latent representations. Experiments on the Kodak datasets show that our framework is superior to other deep learning compression frameworks such as compressed autoencoders in the MS-SSIM indicator under the same bit error rate.

1. Introduction
With the commercialization and deployment of 5G networks, the rapid development of various applications has led to a rapid increase in data and various application interactions. The big data white paper issued by the China Academy of Information and Communications Technology (CAICT) shows that the amount of data generated in 2020 will reach 47ZB, and the global data volume is about to usher in a larger-scale outbreak. In this huge amount of data, image and video data occupies a large proportion. A large amount of image and video data causes an increase in hardware storage costs and a waste of bandwidth. In this context, image compression technology is particularly important. In recent years, the deep learning image compression framework has achieved great success, but the current compression framework focuses on improving the compression effect, and the research on the anti-error ability of the deep learning framework has not been in-depth. This paper deeply studies the anti-noise performance of image compression framework based on deep learning, and is dedicated to providing ideas for solving the anti-interference compression framework of deep learning framework. We combine the variational autoencoder[12] and the compression framework to improve the anti-noise performance of the deep learning compression framework.

2. Related Works
Generally speaking, traditional image compression technologies are divided into lossy image compression technologies and lossless image technologies. Lossless image compression usually uses techniques such as Huffman algorithm, arithmetic coding, run-length coding, etc. to eliminate spatial and temporal redundancy in the image to achieve the purpose of compression. Since the lossless compression technology retains all the information of the image during the compression process, the compression ratio is generally small. Compared with the lossless image compression technology, the lossy image compression standard represented by JPEG[1] is more widely used. The JPEG image compression standard uses discrete cosine algorithm to convert images into frequency domain space, and uses non-uniform quantization to reduce image high-frequency components to compress images.
JPEG image compression technology relies on manual design of complex non-linear transformations, and the encoder and decoder need to be optimized independently, so the flexibility and performance are becoming less and less suitable for current image compression requirements.

In recent years, the image compression framework based on deep learning has been continuously improved. Balle et al. [2] first used convolutional autoencoders in image compression. They added a uniform quantizer to the autoencoder structure to compress the input image, and solved the problem that the model cannot be trained introduced by quantization. Not only that, they added GDN structure to the network to optimize the compression effect. Experiments have proved that their frame compression performance is better than traditional compression techniques such as JPEG.

Thesi et al.[3] added an entropy rate estimation module to the framework based on Balle’s work, so that the overall framework can be optimized for entropy coding and compression, and the convolutional autoencoder based on the residual network solves the problem of deep neural network gradient calculation, so that the frame has a better compression effect.

The work of Li et al.[4] went one step further, adding importance map branches to the convolutional autoencoder. Learning the spatial characteristics of the image through the convolutional autoencoder realizes the adaptive bit rate allocation according to importance map.

Since the compression method based on the convolutional autoencoder has problems in rate control, different networks need to be trained for different compressed image bit rates, which will lead to a great waste of training resources. In order to solve the above problems, Toderici et al.[5-6] proposed a compression framework based on LSTM for the first time. The article uses the LSTM network to input memory and parameter sharing properties, and uses the LSTM network for progressive encoding. Baig et al. [7] combined RNN and image restoration technology to improve the compression effect.

Tschannen et al. [8] first used GAN for image compression. On this basis, Agustsson et al. [9] not only used adversarial training to improve the compression effect, but also used the generation ability of GAN to generate part of the image.

Rippel et al. [10] combined the multi-scale model and adversarial training in the field of image compression for the first time, and they surpassed all traditional image compression techniques. The same applies the multi-scale model Nakanishi [11] on the other hand to enhance the compression effect. They combine the multi-scale model and the entropy coding module to improve the performance of the framework. Although the above-mentioned research has been very effective in image compression, their framework suffers a greater loss of accuracy in reconstructed images when code errors occur. This paper proposes a compression framework based on a variational autoencoder, which improves the error resistance performance of the image compression framework.

3. Variational autoencoder compression framework

3.1. Deep learning compression framework

A typical deep learning image compression framework is shown in Figure 1. The frame is composed of encoder, decoder and quantizer:

$$E(x): X^n \rightarrow Z^n$$

$$Q(z): Z^n \rightarrow \tilde{Z}^n$$

$$D(\tilde{z}): \tilde{Z}^n \rightarrow \hat{X}^n$$

As in the above formula, the encoder $E(x)$ down-samples the input data, the quantizer $Q(z)$ converts the float number into an integer, and the decoder reconstructs the image. At this stage, almost all autoencoder structure compression frameworks are based on this structure and its variants, achieving excellent compression performance. However, this structure has a very significant problem: the features that are down-sampled by the deep convolutional network are highly compressed. From the perspective of information theory, $E(x)$ removes a lot of redundant information in the process of
mapping the $X$ from n-dimensional to m-dimensional. $z$ contains highly concentrated information related to image reconstruction, which is also the purpose of framework optimization. However, once $z$ changes due to noise interference during transmission, the information related to the image reconstruction task will be contaminated, and the image reconstruction quality will also decline rapidly. Shannon's information theory believes that source coding needs to compress images as much as possible to remove redundancy, while channel coding needs to reintroduce redundancy to improve noise immunity. Information theory shows that today’s deep learning compression framework compresses information as much as possible is not conducive to anti-noise. Inspired by this, we tried to reintroduce noise to the compression framework to improve the anti-interference ability of the coding against noise, and improve the learning goal of the framework, so that the framework achieves the mutual trade-off and joint optimization of noise resistance and compression.

![Fig. 1 Typical deep learning compression framework](image1)

3.2. Variational autoencoder compression framework

As mentioned above, our goal is to introduce noise training in the deep learning compression framework. Naturally let us think of a variational autoencoder. Unlike the typical compression framework, the variational autoencoder does not directly learn the latent representation of the data. The variational autoencoder uses two encoders to learn the mean and variance of the latent representation, and introduces Gaussian noise to express it as a distribution. As shown in Figure 2, the input image $X$ is down-sampled by a deep convolutional neural network and then quantized into a binary code by a quantizer. The binary code is sent to the two CNN networks after channel transmission to learn the mean and variance of the data respectively.

![Fig. 2 Variational autoencoder image compression framework](image2)

Variational autoencoder adds standard normal distribution noise to the hidden layer of the model and adds penalty term KL divergence to the loss function so that the output of the estimating network
is the statistical distribution of the training data learned by the model. The loss function of the variational autoencoder is expressed as:

\[
\text{Loss} = \mathbb{E}_{x \sim p(x)} \left[ \mathbb{E}_{z \sim p(z \mid x)} \left[ -\log q(x \mid z) \right] \right] + KL(p(z \mid x) \parallel q(z))
\]  

(1)

The first term of the formula is equivalent to the reconstruction loss of the ordinary autoencoder, and the second term is the KL divergence of the posterior distribution and the prior distribution. Optimize the first term so that the reconstruction loss of the input image and the reconstructed image is as small as possible, that is, it is hoped that the latent representation retains as much information about the task as possible, that is, the information of the reconstructed image. The second item of the optimization goal is to make the statistical distribution represented by the middle layer as close to the standard normal distribution as possible. There are two advantages here. One is to introduce noise for training to make the decoder robust to noise; the other is to decouple features. The function of KL divergence is to let the latent express the alignment to the standard normal distribution. The standard normal distribution has a very significant feature, that is, each component of the standard normal distribution is independent of each other. This is a very good property in feature learning, which means that changing one of the components will not affect the other components. Higgins et al. [13] prove that variational autoencoders use penalty terms to decouple the middle-level representation as much as possible, while ordinary autoencoder structures do not have such properties. As a result, when a bit error occurs, the change of one component may cause different features to change at the same time when reconstructing the image, which may further aggravate the loss of image reconstruction. In addition, considering the particularity of the compression task, we not only added Gaussian noise in the middle layer. In fact, the quantization noise caused by our quantization will also affect the task. Considering that quantization itself is a kind of damage to the middle layer representation, we believe that the variational autoencoder should learn the data characteristics after quantization, instead of sampling the middle layer representation before quantization and then quantizing. From the perspective of feature decoupling, quantization after sampling will destroy and train sufficiently decoupled features. However, after sampling after quantization, even if the encoding changes, the features are still independent of each other, which can still prevent noise. We proved this in subsequent experiments.

4. Experiment and Analysis

4.1. Model architecture

Our framework is shown in Figure 2. The encoder includes three down-sampled convolutional neural networks and 15 residual blocks, followed by two down-sampled convolutional layers. The quantizer is a binary quantizer:

\[
Q(z_q) = l, \text{if} \quad \frac{l}{2} < z_q < \frac{l + 1}{2}, l = 0, 1
\]  

(2)

The mean variance estimation network is a convolutional neural network. The decoder is the mirror structure of the encoder.

4.2. Experiment Setup

We use the Pytorch framework to build the model, the initial learning rate is 4e-3, the optimization function is Adam, we train the model 300 epochs, the training set is BSDS500, and the test set is Kodak24. We trained a typical convolution compression autoencoder to compare with our framework, which architecture is basically the same as ours, except for the distribution estimation module. At the same time, for the variational autoencoder compression framework, we tried two schemes, placing the quantizer after sampling and before sampling, to verify our conjecture. We use multi-scale structural similarity (MS-SSIM) as an evaluation indicator of image reconstruction quality. We trained two compression models with different bits per pixel (bpp), which bpp are 0.25 and 0.5 respectively, to
observe the anti-noise performance of the framework under different compression ratios. We add noise to the binary code to simulate the channel, and use the final bit error rate as a measure.

4.3. Loss Function
We define the loss function as a trade-off between image reconstruction accuracy and KL divergence. The first item is defined as the MS-SSMI value between the original image and the reconstructed image, KL divergence is calculated using reparameterization trick:

\[
Loss = C \ast (1 - MSSSIM(X, \hat{X})) + \frac{1}{2} \beta(-\log \sigma^2 + \mu^2 + \sigma^2 - 1)
\]

Where \( X \) is the input image, \( \hat{X} \) is the reconstructed image, \( C \) and \( \beta \) are the trade-off factors, \( \sigma \) is the standard deviation, and \( \mu \) is the mean.

4.4. Experiment results
We trained two models with different bpp and tested them on the kodak test set to simulate channel errors by adding noise to the binary code. In order to reduce the error caused by randomness, we conducted 10 experiments for each bit error rate and took the average value to record. Our experimental results are shown in Figure 3 and Figure 4.

![Fig. 3 Under different bit error rate scenarios, the accuracy of reconstructed images of VCAE and CAE, the model’s bpp is 0.5](image.png)
From the experimental results in Fig. 3 and Fig. 4, it can be seen that our variational compression autoencoder (VCAE) is better than ordinary compression autoencoders in anti-noise performance. The models of different bpp show that the performance is relatively close when the bit error rate is small. When the bit error rate gradually increases, the image accuracy of our VCAE model decreases less, and the accuracy of the CAE model decreases more. Especially on the 0.5bpp model, when the bit error rate reaches 0.21, the difference between the two MSSSIMs is nearly 20%. It is worth noting that we are concerned about the anti-noise performance of the compression frame. Therefore, our compression model does not add an entropy coding module. How to improve the anti-noise performance with the entropy coding module will be our future work.

![Graph showing MSSSIM vs BER for VCAE and CAE with bpp=0.25](image)

**Table 1. VCAE frame quantizer in different positions**

| BER  | VCAE Quantizer before sampling | VCAE Quantizer after sampling |
|------|-------------------------------|-----------------------------|
| 0%   | MS-SSIM:0.9                   | MS-SSIM:0.89                |
| 6%   | MS-SSIM:0.892                 | MS-SSIM:0.882               |
| 12%  | MS-SSIM:0.867                 | MS-SSIM:0.821               |

As shown in the experimental data in Table 1, when the quantizer is located after sampling, the anti-noise performance of the VCAE framework is basically not improved, but the introduction of noise training and quantization loss is slightly lower than the CAE performance.

### 5. Conclusion

This paper studies variational autoencoders, and designs an image compression framework (VCAE) based on variational autoencoders for possible noise interference and error problems in the compression framework. Variational autoencoders introduce noise to train a robust decoder, and learn the decoupling representation to improve the anti-noise performance of the framework. We conclude that the VCAE frame has better compression image quality than the ordinary CAE frame under different bit error rate scenarios, and has better anti-noise performance.
References

[1] Wallace G K. (1992) The JPEG still picture compression standard[J]. Communications of the Acm, 38(1):xviii-xxxiv.

[2] J. Ball’e, V. Laparra, and E. P. Simoncelli. (2016) End-to-end optimized image compression[J]. arXiv preprint arXiv:161101704.

[3] Theis L, Shi W, Cunningham, Andrew, Huszár, Ferenc. (2017) Lossy Image Compression with Compressive Autoencoders[J]. arXiv preprint arXiv:1703.00395.

[4] Li M, Zuo W, Gu S, et al.(2018) Learning convolutional networks for content-weighted image compression[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018:3214-3223.

[5] Toderici G, OMalley S M, Hwang S J, et al.(2015) Variable rate image compression with recurrent neural networks[J]. arXiv preprint arXiv:1511.06085.

[6] Toderici G, Vincent D, Johnston N et al. (2017) Full resolution image compression with recurrent neural networks[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017:5304-5314.

[7] Baig M H, Koltun V, Torresani L. (2017) Learning to inpaint for image compression[C]. Advances in Neural Information Processing Systems. 2017:1246-1255.

[8] Tschannen M, Agustsson E, Lucic M. (2018) Deep generative models for distribution-preserving lossy compression[C]. Advances in Neural Information Processing Systems. 2018:5929-5940.

[9] Agustsson E, Tschannen M, Mentzer F, et al. (2019) Generative Adversarial Networks for Extreme Learned Image Compression[C]. IEEE/CVF international Conference on Computer Vision (ICCV).

[10] Rippel O, Bourdev L. (2017) Real-time adaptive image compression[C]. Proceedings of the 34th International Conference on Machine Learning-Volume 70. JMLR. org, 2017:2922-2930.

[11] Nakanishi K M, Maeda S, Miyato T, et al. (2018) Neural multi-scale image compression[C]. Asian Conference on Computer Vision. Springer, Cham, 2018:718-732.

[12] D.P. Kingma and M. Welling, (2013) Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114.

[13] Higgins I, Matthey L, Pal A. Burgess, C Glorot X., Botvinick M, Mohamed S, Lerchner A. (2017). beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. ICLR.