CommLab Final Project Report – Image Compression

石旻翰
National Taiwan University,
Electrical Engineering

邱啓翰
National Taiwan University,
Electrical Engineering

1 Introduction of Image Compression

As image data size grows rapidly in recent years, image compression plays an important role in data transfer and storage. However, it is a challenging task since there are some trade-off between the data size and its quality, which leads to two methods of image compression – Lossy Compression and Lossless Compression. Also, the other challenge is that there are highly complex unknown correlations between pixels. Thus, it is difficult to find and recover them. In our work, we are eager to seek a well-compressed representation for images, try to implement it in lossy and lossless way, and compare the resulted images generated by classical methods, like JPEG, and those done by machine-learning methods, like AutoEncoder.

2 Related Work

2.1 Classical Image Compression

2.1.1 JPEG

JPEG is proposed in 1992. With a deep insight to human eyes weakness, research shows that human eyes are not sensitive the high-frequency signals in pictures, but sensitive to the brightness of colors. YCbCr use a normalization formula shown in Fig.1 of RGB to adjust the brightness of images, and this improves the efficiency of the encoder for image compression. JPEG use Discrete Cosine Transform (DCT) and Quantization to filter the high-frequency signals, leading to the loss of information for the images. However, this loss can not be detected by human eyes.

$$
\begin{aligned}
Y &= (77R + 150G + 29B)/256 \\
Cb &= (-44R - 87G + 131B)/256 + 128 \\
Cr &= (131R - 110G - 21B)/256 + 128
\end{aligned}
$$

Figure 1: YCbCr formulas

2.1.2 JPEG2000

From the name of JPEG 2000, it is clear that it is the Joint Photographic Experts Group that developed JPEG 2000. It is an improved version of JPEG that provides better compression with lossless information during conversion. It is mainly based on wavelet technology, which is more difficult than JPEG. Fig. 2 shows the detailed data flow of JPEG 2000. Due to its algorithmic complexity in implementation and patent issues, JPEG 2000 has never become a standard in many software applications. However, its advantage of non-distortion is still useful in areas such as biomedical image analysis.

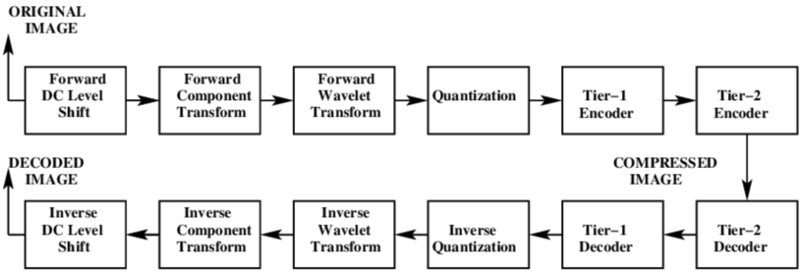

Figure 2: JPEG 2020 dataflow [1]

2.2 Machine-Learning Based Image Compression

2.2.1 CNN-AutoEncoder

Autoencoder is a lossy image compression method, the model structure is shown in Fig.3. From the structure, there are two parts, encoder and decoder. They are constructed by CNN. Encoder encodes the input image to a latent space, and decoder decodes the compressed representation in the latent space. Through this model, we can adjust the dimension of latent space to achieve image compression with different compression ratios, and the quality of output images depends on the training process.

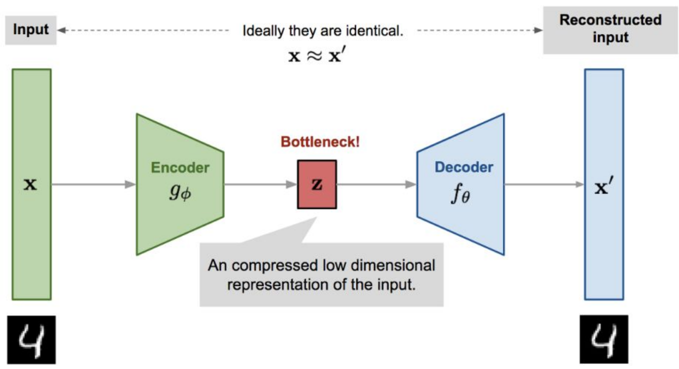

Figure 3: Model Structure of AutoEncoder

2.2.2 MLPs

MLPs applied for image compression is a lossless compression method, for we adopt Huffman coding to code the error images. The model structure of MLPs [2] is shown in Fig.4. MLPs use the hidden layers to achieve image compression, and the best setting of the number of the hidden layer can be found through experiments.

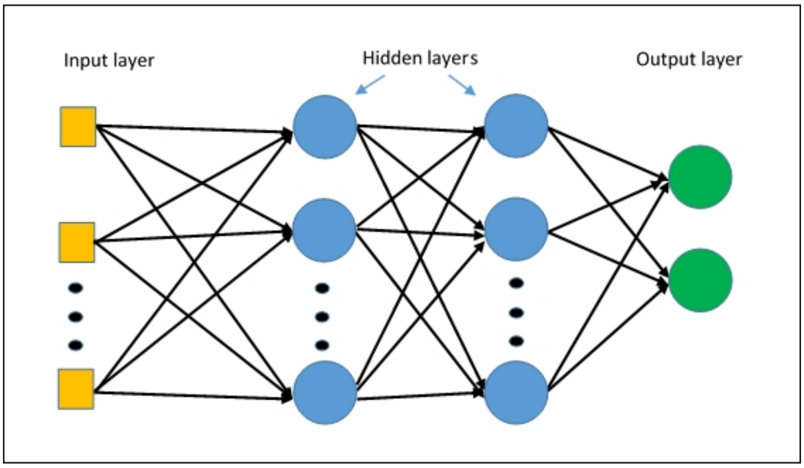

Figure 4: Model Structure of MLPs

3 Experiment Setup

3.1 Dataset

We use a dataset with 5000 pictures in the form of BMP to train our model, whose size is 512 x 512 x 3, so the required storage of one original picture is equal to nearly 786kB.

3.2 Evaluation Metric

Bits per pixel(bpp)
We adopt Bits per pixel(bpp) as one of our evaluation metrics for the comparison between lossless image compression methods, like JPEG2000 and MLPs.
Peak Signal to Noise Ratio(PSNR) and Structural Similaritu Index(SSIM)
We use two different metrics for comparisons. They are Peak Signal Noise Ratio(PSNR) and Structural Similarity Index(SSIM), respectively. PSNR can be formulated as:

$$\mathrm{PSNR} = 20log_{10}\left(\frac{255}{\sqrt{\mathrm{MSE}}}\right)$$

where,

$$\mathrm{MSE} = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left(I(i,j) - I'(i,j)\right)^2$$

I is the input image and I' is the reconstructed image after decoding. Note that, the larger value of PSNR shows the image compression algorithm performs better. However, there would be biased if we only adopt one evaluation metric, so we find the other one, Signal Similarity Index(SSIM), which can be formulated as:

$$\mathrm{SSIM}(I,I') = \frac{(2\mu_I\mu_{I'} + c_1)(2\sigma_{II'} + c_2)}{(\mu_I^2 + \mu_{I'}^2 + c_1)(\sigma_I^2 + \sigma_{I'}^2 + c_2)}$$

μ_I and $\mu_{I'}$ is the mean value of input images I and reconstructed images I', respectively, and σ_i and $\sigma_{I'}$ is the standard deviation of I and I', respectively. c_1 and c_2 are constants, and in our work, we set $c_1 = 6.5$, and $c_2 = 58.5$. Note that the range of SSIM is between -1 and 1, and SSIM=1 means that the images are exactly the same.

4 Results Analysis

4.1 Lossy Image Compression

Lossy images include JPEG, CNN AutoEncoder, and RNN AutoEncoder. We tried different compression rates for the three to find out the distortion-compression curve.

4.1.1 JPEG

We tried 100 different compression rates by changing the quantization table, and finally get Fig. 5. We can find that the SSIM reaches 1.0 and the MSE reaches 0, indicating that the image is almost lossless when bpp reaches 11 or $11/24$ in compression ratio, namely. On the other hand, when bpp is less than 2, obvious image distortions are visible to the naked eye.

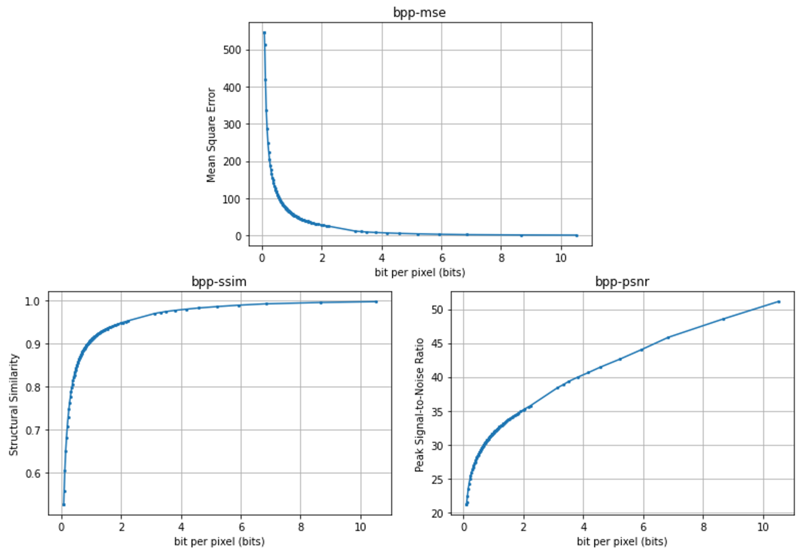

Figure 5: distortion-compression diagram of JPEG

4.1.2 CNN-AutoEncoder

Our CNN-AutoEncoder encoder model consists of two layers of the Conv-ReLU- BN structure and a binarizer with sign function and a Conv layer, and our decoder model contains two layers of the Conv-ReLU- BN structure whose size is inverse to that of the encoder model.
For our experiments, we use the binarizer, which is the last layer of the encoder, in different sizes and train our model from scratch for over 500 epochs each. Fig. 6 shows our result for this task. From the graph below, it can be seen that our performance is slightly worse than JPEG. There are a few possible reasons for this phenomenon, including small datasets, a lack of pre-trained models, shallow model layers... etc. Fortunately, we can still see the potential of CNN-AE in image compression, as it offers acceptable performance at a low compression ratio.

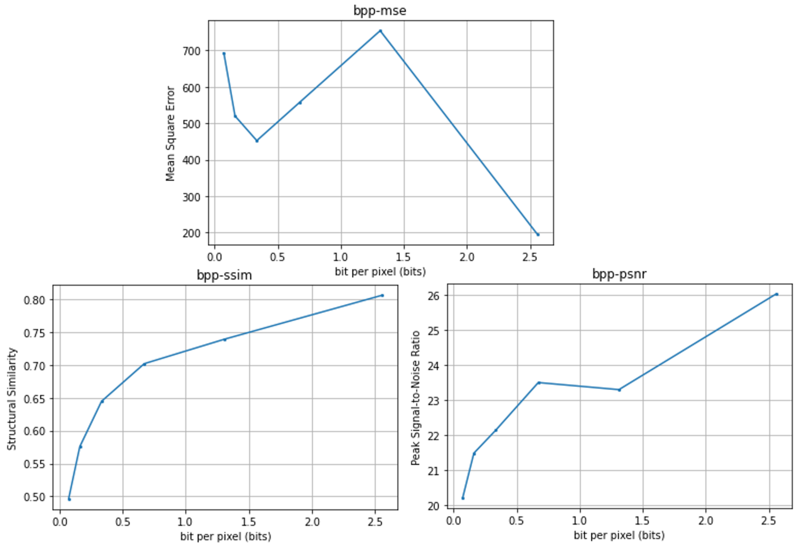

Figure 6: distortion-compression diagram of CNN-AutoEncoder

4.1.3 Further Experiments

We also conducted some experiments with different model structures, including LSTM AutoEncoder and modified CNN AutoEncoder In the first experiment, we use a similar structure to the original CNN-AE, but with deeper Conv layers and LSTM cells. The following table shows our results, and as expected, we were able to improve the performance. All the evaluation metrics are much better than the original CNN-AE design, especially the MSE performance. Therefore, we can expect to get even better performance when we construct the model deeper and get a larger data set.

model	SSIM	MSE	PSNR	BPP
original	0.72	615.22	23.45	0.91
new trial	0.78	171.08	27.11	0.91

In the last experiment, we try to train models without binarization function, i.e., without sign function and get a terrible result where SSIM falls below 0.25 while bpp reaches almost 1.0. Thus, we can conclude that binarization is a crucial element for image compression in CNN-based encoders.

4.2 Lossless Image Compression

In this subsection, we compare the performance of two lossless compression methods, JPEG 2000 and MLPs, to see which is better. Since both methods are lossless, their PSNR is infinite, their SSIM is 1.0, and their MSE is 0. Therefore, we focus on comparing their bits per pixel.

4.2.1 JPEG 2000

Due to the complexity of the algorithm, we only encode BMP files into J2K files using an online converter, so we do not know the quantization table and other details. The bpp value of JPEG

2000 is **11.14** and the compression ratio is **0.464**. Interestingly, the bpp value is almost the same as JPEG and the SSIM is almost equal to 1.0. Perhaps this is an indication of the minimum ratio of such an algorithm to obtain a lossless image.

4.3 MLPs

Our result is largely based on the open-source [3] method, as we tried it ourselves several times but could not get a better performance than this. In the end, the bpp of MLP is **8.6** and the compression ratio is **0.36**, which is much better than JPEG 2000. Although we get quite satisfactory results, we believe we can still improve the performance for deeper MLP layers since the source is designed for images with smaller sizes.

5 Conclusion

- For the size of pictures in our dataset is quite large(512 x 512 x 3), we think the machine learning base method, like CNN-AutoEncoder and MLPs, should be cascaded deeper to achieve a better result.
- Apart from the special compression requirements, a bpp value of two may be a threshold above which one can see significant differences with the naked eye.

References

[1] S. Attluri, Vikram Jayaram, and Bryan Usevitch. "Optimized Rate Allocation of Hyperspectral Images in Compressed Domain Under JPEG 2000 Part 2." In: Dec. 2022, pp. 228–232. ISBN: 1-4244-0069-4. DOI: 10.1109/SSIAI.2006.1633756.

[2] *Implicit image compression*. URL: https://varun19299.github.io/implicit-image-compression/.

[3] Scelesticsiva. *Scelesticsiva/neural-networks-for-image-compression: Image compression using neural networks*. URL: https://github.com/scelesticsiva/Neural-Networks-for-Image-Compression.

Work Distribution

Member	Work and contribution	Cowork
邱啓翰	CNN-AE, JPEG, JPEG2000	Report, Slides
石旻翰	MLPs, CNN-AE	Report, Slides