A comparative study of classical and neural based approaches for image compression

Ritu Raut(rraut2@wisc.edu), Ajay Shridhar Joshi(joshi28@wisc.edu)

6th October 2022

1 Problem Statement

Image compression is a type of data compression applied to digital images to reduce their cost for storage or transmission. The objective of image compression is to represent an image with as few bits as possible while still maintaining a high degree of fidelity to the original image. Current state-of-the-art classical algorithms take advantage of visual perception and the statistical properties of image data. Learning-based strategies, on the other hand, aim to directly learn an end-to-end mapping between the input and the output of the compression process. While the potential of learning-based image compression has been shown in recent years, the computational efficiency and memory requirements of these methods have been a limiting factor in their practicality.Recently, Latent diffusion models [6] have emerged as a promising approach for learning-based image compression. LDMs are autoencoder models that are trained to compress an input image by first mapping it to a latent space and then encoding the latent code with a entropy codec.

In this project, we aim to conduct a comprehensive survey on classical and diffusion based methods. We plan to evaluate the recent image compression methods based on diversified evaluation measures like compression ratio, the bit rate, and the peak signal-to-noise ratio, summarize their merits, and create hybrid approaches for specific design choices.

2 Related Work

Techniques for compression can be divided into 2 broad categories: traditional algorithms(eg: JPEG/JFIF, PNG) and learning-based algorithms(eg: PCA, K-means clustering, NN-based methods).

2.1 Classical Algorithms

Most widely used algorithms for lossless and lossy compression are PNG and JPEG. PNG, at a high-level is composed of 3-stages- delta encoding, LZ77 compression and Huffman coding. On the other hand, JPEG (JFIF) has a 4-stage pipeline consisting of a color transform to YCbCr, Discrete Cosine Transform followed by quantization and Huffman coding for the DCT coefficients.

2.2 Learning-based algorithms

Machine Learning-based algorithms (e.g., PCA, K-means clustering, neural networks) can be used to find a more compact representation of images, which helps compress them. There is a whole range of approaches that can be used for compression using ML. Some of the approaches we consider are [5], [6].

2.2.1 High-Fidelity Generative Image Compression

[5] propose to combine a generative adversarial method and learned compression method to achieve high quality reconstructions that are very close to the input, for high-resolution images (up to 2000×2000 pixels). In particular, they extensively explored with different configurations of normalization layers, generator and discriminator architectures, training strategies, as well as perceptual losses. They quantitatively evaluated approach with FID [3], KID[1] , LPIPS [8], and the classical distortion metrics Peak Signal to Noise Ratio and showed that results are consistent with the rate-distortion-perception theory.

2.2.2 Diffusion based models

[6] proposed latent diffusion models (LDMs) achieve new state-of-the-art scores for image inpainting and class-conditional image synthesis and highly competitive performance on various tasks, including text-to-image synthesis, unconditional image generation and super-resolution. Previous methods applied denoised optimization in the pixel space. [2]. It consumes hundreds of GPU days and inference is expensive due to sequential evaluations. Whereas [6] apply them in the latent space of powerful pretrained autoencoders using n limited computational resources while retaining their quality and flexibility. Stable Diffusion uses three trained artificial neural networks in tandem: i) Variational Auto Encoder[4] ii) U-Net [7] iii) Text Encoder The Variational Auto Encoder (VAE) encodes and decodes images from image space into some latent space representation. The latent space representation is a lower resolution (64 x 64), higher precision (4x32 bit) representation of any source image (512 x 512 at 3x8 or 4x8 bit). In this project, we are planning to utilize a variational autoencoder to compress images.

3 Implementation

3.1 Traditional algorithms

We implemented 2 traditional algorithms: PNG and JPEG(JFIF) and the following image compression techniques used in the pipeline for these algorithms:

1. Delta encoding
2. Block-based Discrete Cosine Transform (DCT- Real to real)
3. RLE Encoding
4. Color transform - RGB to YCbCr and vice versa
5. Quantization using a quantization table for DCT coefficients, to compress number of bits to represent a channel: eg: Cb or Cr channels

We faced a lot of challenges in implementing them in the form of figuring out normalization constants, clipping numbers to be particular ranges (0-255), and challenges with bit packing (reducing from 8-bit to 7/6 bits for quantization).

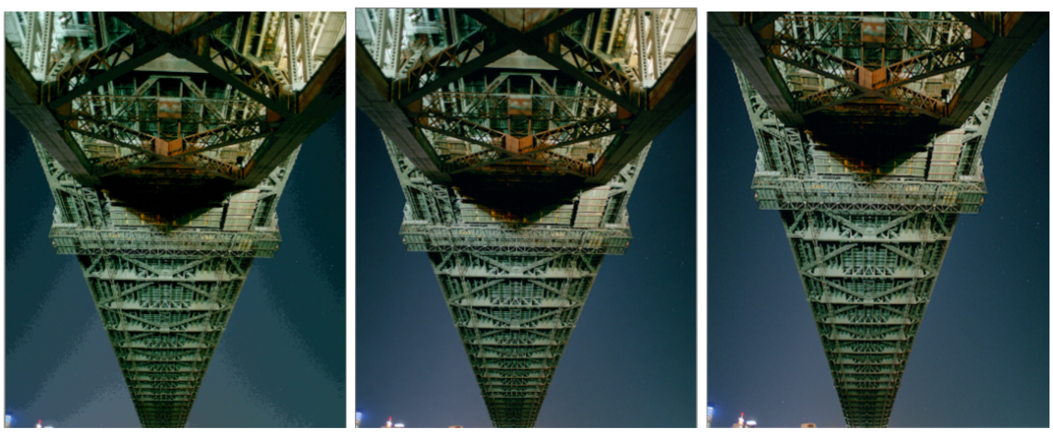
Since all techniques are implemented as independent composable layers, we also experimented by putting together multiple combinations of techniques and found out some combinations which do not work well and worsen the compression performance: eg: Using RLE and Delta encoding with ZLib and Using delta encoding with DCT + quantization gave worse compression performance than by Delta + ZLib, DCT + Quantization. Our implementation can be found here for reference.

3.2 Neural Codecs

We used trained weights for the Stable Diffusion [6] model as well as for HiFiC[5], which is a part of the Tensorflow Compression library, and ran inference on these neural codecs. The weights for the Stable Diffusion model were obtained through the HuggingFace library, and the weights for HiFiC [5] were obtained through the TF Compression library.

4 Results

We present preliminary results obtained by using our implementation for JPEG, Stable Diffusion[6], HiFiC[5] in this section. In JPEG generated image, due to quantization, there is loss of information in the image, which leads to discrete patches in the sky. Stable diffusion model [6] compressed image are hallucinating information for example, remvoing the bright light effect in the top left corner as it is seen in figure 1(c). HiFiC[5] produces a better quality image and does not encounter issue of hallucination.



(a) JPEG (b) HiFiC (c) Stable Diffusion

Figure 1: Reconstructed images of Bridge from Image Compression Benchmark using i) Our implementation of JPEG, ii) HiFiC, iii) Stable Diffusion

5 Future work

We plan to compare the image compression results between the classical and diffusion-based methods i.e. Stable-Diffusion based and JPEG. We'll study these algorithms based on the evaluation measures listed below, taking into account factors like compression speed, quality, and size. Other than this, we will apply traditional computer vision techniques like dithering [1] and paletting [2] to improve and further compressed image by diffusion models.

[1] https://www.wikiwand.com/en/Dither
[2] https://www.wikiwand.com/en/Palette$_{(}computing$)

6 Evaluation

In this section, we discuss our evaluation setting for this project. For this task, we are using the CLIC, Kodak dataset to evaluate our approaches. The CLIC dataset contains 30 images. The Kodak dataset is a set of images that have been compressed using the Kodak image compression algorithm. The dataset contains a total of 24 images, and each image has been compressed using a different level of compression. The dataset is designed to be used for image compression research, and it is hoped that it will be useful for comparing different image compression algorithms.

We will evaluate our compression algorithms using the compression ratio, the bit rate, the peak signal-to-noise ratio, and the structural similarity index. The different axes we will be using to compare different image compression algorithms are encoding speed, decoding speed, and reduction in image quality for the compressed image, which can be measured using multiple metrics.

7 Timeline

Time	Task	Status
Week 1-2	JPEG + PNG implementation	Completed
Week 3-4	ML-based compression methods (1-2 methods implementation)	Completed
Week 5 - 6	Experiments for introducing novelty (eg: improving codec runtime/compression ratio, fusing classical + ML-based approaches)	In Progress
Week 7	Evaluation on datasets + Interpretation of results/inferences	To be done
Week 8	Demo + Webpage + Final PPT + Report	To be done

References

[1] Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans. *arXiv preprint arXiv:1801.01401*, 2018.

[2] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

[3] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[4] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[5] Fabian Mentzer, George D Toderici, Michael Tschannen, and Eirikur Agustsson. High-fidelity generative image compression. *Advances in Neural Information Processing Systems*, 33, 2020.

[6] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

[7] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[8] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.