Exploring Model Compression Techniques for Deep Learning based Image Compression Models

Jenny Yu, Rui Zhu, Parinita Edke

Executive Summary

- Image compression is crucial as images continue to increase in number due to advancements in computing power and mobile camera capabilities.
- To save space and transfer images more quickly, it is necessary to reduce the size of digital images while maintaining their necessary information. Deep Neural Networks (DNNs) have become powerful tools and are an avenue to explore.
- However, DNNs are computationally intensive, requiring acceleration techniques to meet increasing needs. These techniques can be categorized into three categories: model compression, computational optimization, and dataflow optimization.
- **This project focuses on model compression techniques due to the high cost of transferring data and computation between different units.**

Problem

Image compression algorithms aim to reduce the amount of data required to be stored while preserving a high quality of the image. In recent years, deep learning-based approaches to image compression have shown significant improvements over traditional methods.

However, it is well-known that these deep learning models are often computationally expensive and memory-intensive, making them impractical for use in resource-constrained environments such as mobile devices and embedded systems.

Model compression techniques can be used to reduce the size of the models, making them more suitable for deployment on such devices. We aim to explore a variety of model compression techniques to deep learning-based image compression algorithms. By compressing the model, we can reduce its memory and computational requirements while still maintaining a decent image quality. This can lead to faster inference times, reduced energy consumption, and overall improved efficiency of image compression algorithms.

Related Work

- High Fidelity Compression (HiFiC) [1] achieved SOTA generative lossy compression results with conditional GANs [2] and a loss function that combines MSE with LPIPS [3], which measures the "perceptual distortion".
- Toderici et al proposed a RNN architecture consisting of a RNN-based encoder and decoder, binarizer and a neural network for entropy coding [4]. The architecture allows for the gradual improvement and refinement of the output image as more data is received from the network.
- Tantawy et al. [5] summarized three techniques for model compression: pruning, knowledge distillation, and lowering numeric precision.
- **Novelty:** We investigate the trade-off between size and latency reduction, and image quality on deep learning-based image compression approaches and extend previous research on model compression on a variety of image compression models by benchmarking the SOTA model compression approaches.

Approach

Generating the baseline

- Load the checkpoints of the HiFiC [1] and the RNN-based compression algorithm [4], with target bitrate 0.30 and 0.375, respectively.
- Use the KODAK dataset [6] to test model performance and generate the Structural Similarity Index (SSIM) [7] as performance metrics. Total inference time and average compressed image size are also recorded.

Applying model compression techniques

- *Model pruning:* prune some connections within the network to boost robustness.
- *Quantization:* use a reduced precision integer representation for the weights and/or activations.
- Generate performance metrics and compare with the baseline performance.

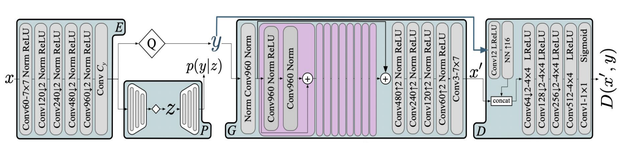

Figure 1: HiFiC architecture. The network consists of an convolution-based encoder *E,* generator *G*, a probability model *P* for modeling posterior probability, and discriminator *D* for adversarial training.

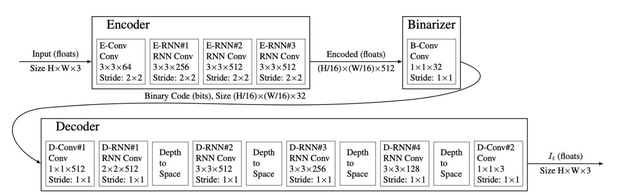

Figure 2: RNN-based architecture. The network consists of a RNN-based encoder and decoder, binarizer and entropy coding.

Results and Discussion

Pruning:

- For the RNN-based model, the pruning was done on the convolution layers. Different ratios were applied for different portions of the model (encoder, decoder). The pruning did not reduce total inference time and no difference in performance was observed.
- For the HiFiC algorithm, the pruning was done on both convolution layers and transpose convolution layers. The inference time was slightly reduced. The model's performance was worse when pruning was done on the encoder.
- The RNN-based model had a longer inference time.

Table 1: Performance of the RNN based model with pruning.

Encoder pruning ratio	Decoder pruning ratio	Total inference time	SSIM	Avg. compressed image size (B)
0	0	00:17:36	0.7886	17545.666
0.3	0	00:17:35	0.7858	17439.416
0	0.3	00:17:36	0.7876	17545.666
0.3	0.3	00:17:36	0.785	17439.416

Table 2: Performance of HiFiC model with pruning.

Encoder pruning ratio	Generator pruning ratio	Probability model pruning ratio	Total inference time	SSIM	Avg. compressed image size (B)
0	0	0	00:01:45	0.8179	18850.333
0.3	0	0	00:01:43	0.817	19048.167
0	0.3	0	00:01:40	0.7145	18850.333
0	0	0.3	00:01:40	0.8171	19192
0	0.3	0.3	00:01:40	0.7142	19048.167
0.3	0.3	0	00:01:41	0.7145	19048.167
0	0.3	0.3	00:01:40	0.7142	19192
0.3	0	0.3	00:01:42	0.8164	19429.5
0.3	0.3	0.3	00:01:37	0.7139	19429.5

Quantization:

- For the RNN-based model, as the model size decreases, the range of values is compressed resulting in loss of important information, which leads to inaccurate outputs. Higher precision for quantization might improve the accuracy but is currently not implemented in PyTorch.
- Quantization can not be applied on the HiFiC model as it is not implemented for transpose convolution layers.

References

[1] F. Mentzer, G. Toderici, M. Tschannen, and E. Agustsson, "High-fidelity generative image compression," 2020.

[2] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu et al., "Generative adversarial networks," *Communications of the ACM*, vol. 63, pp. 139–144, 6 2014.

[3] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," *Proceedings of the IEEE Computer Society CVPR*, pp. 586–595, 1 2018.

[4] G. Toderici, D. Vincent, N. Johnston, S. J. Hwang, D. Minnen, J. Shor, and M. Covell, "Full resolution image compression with recurrent neural networks," 2016.

[5] D. Tantawy, M. Zahran, and A. Wassal, "A survey on gan acceleration using memory compression technique," 2021.

[6] E. Kodak, "Kodak PhotoCD dataset."

[7] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multi-scale structural similarity for image quality assessment," *Conference Record of the Asilomar Conference on Signals, Systems and Computers*, vol. 2, pp. 1398–1402, 2003.