A

Project Report on,

Image Compression using Deep Learning :Lossless (MLP) & Lossy (Auto-encoder & GAN)

by

Pragati Sidpara

Palak Makadia

Abstract

The exponential growth of data has underscored the significance of efficient image compression techniques for data transfer and storage. This project delves into exploring and implementing deep learning methodologies for both lossless and lossy image compression. Various neural network architectures including Multilayer Perceptron (MLP), Autoencoders (AE), and Generative Adversarial Networks (GANs) were investigated and tested to achieve improved compression ratios and image reconstruction quality.

For lossless compression, predictive coding via MLP networks was employed, utilizing raster scan patterns and predictive models to minimize information redundancy. On the other hand, lossy compression techniques, particularly using CNN-based autoencoders and GANs, aimed to reduce image dimensions while ensuring high-quality reconstruction.

Results showcased promising outcomes, with the best performing networks achieving compression rates close to JPEG-2000 for lossless scenarios and competitive results compared to JPEG and JPEG-2000 for lossy compression on various datasets, including CIFAR and high-resolution images. The project also outlined potential future directions for further enhancing compression algorithms, considering alternative loss functions, leveraging image-specific information, and integrating generative models for more efficient representations.

This comprehensive investigation and implementation of deep learning-based image compression techniques offer promising prospects for efficient data handling in various fields reliant on image data.

Table of Contents

List of Tables

List of figures

List of Abbreviations and Nomenclature

Abbreviations:

- **CNN**: Convolutional Neural Network
- **AE**: Autoencoder
- **GAN**: Generative Adversarial Network
- **MLP**: Multilayer Perceptron
- **RNN**: Recurrent Neural Network
- **FT**: Fine-Tuning
- **CIFAR**: Canadian Institute for Advanced Research

Nomenclature:

- **Predictive Coding**: A method in lossless image compression based on predictive models
- **Decoder**: A component in neural networks responsible for reconstructing input data
- **Compression Ratio**: The ratio of the size of compressed data to the original size
- **Image Quality**: Subjective measure of how well an image is reconstructed post-compression
- **Fine-Tuning Approach**: Methodology for adjusting pre-trained models for specific tasks or datasets

Chapter 1 : Introduction

In an era witnessing an unprecedented surge in data volume, the need for efficient data compression methods has become paramount.Image compression stands as a pivotal component, essential for reducing data size without significant loss of critical visual information.

This project delves into the realm of image compression, exploring the utilization of deep learning methodologies to address the complexities inherent in preserving image quality while reducing data size. Image compression, whether lossless or lossy, presents multifaceted challenges due to intricate pixel correlations, necessitating innovative approaches to derive compressed yet accurately recoverable representations.

Drawing from the foundation of neural networks and deep learning advancements, this study embarks on a comprehensive investigation into various strategies for image compression. It encompasses an exploration of both traditional and modern image compression techniques, culminating in a meticulous evaluation and comparison against established standards such as JPEG and JPEG-2000.

The objective of this project is twofold: to explore the feasibility of leveraging deep learning architectures for image compression and to assess the performance of these methods concerning compression ratio, image quality, and computational efficiency. Through meticulous experimentation and analysis, this report aims to present insights into the efficacy of different deep learning approaches for image compression and their potential for advancing the field.

By examining various neural network architectures, including autoencoders, generative adversarial networks (GANs), and recurrent networks, this project endeavors to pave the way for enhanced image compression methods that strike a balance between data reduction and fidelity of image reconstruction.

The subsequent sections of this report delve into a detailed exploration of methodologies, experimental setups, results, and discussions, presenting a comprehensive overview of the findings and implications derived from these investigations.

Chapter 2 : Literature Survey

Image compression stands as a fundamental pillar in managing the burgeoning volume of visual data while ensuring efficient storage and transmission. The demand for compact representations without compromising perceptual quality underscores the criticality of compression techniques. Traditional methodologies such as JPEG and JPEG-2000, employing discrete cosine transform (DCT) algorithms, have been instrumental in achieving both lossy and lossless compression. Despite their widespread use, these methods confront challenges, especially in maintaining image fidelity under high compression ratios. The pursuit of novel techniques has led to a transformative shift toward neural network-based approaches, revolutionizing the landscape of image compression methodologies.

1. Classical Image Compression Methods

The field of image compression has historically relied on classical methods like JPEG and JPEG-2000. These techniques employ algorithms based on discrete cosine transform (DCT) and have provided effective means for both lossy and lossless compression. While widely used, they suffer from certain limitations, particularly in preserving image quality under high compression ratios.

2. Neural Network-Based Approaches

Recent years have witnessed a paradigm shift toward neural network-based approaches for image compression. Pioneering works, such as Toderici et al. (2016), introduced recurrent neural networks (RNNs) for high-resolution image compression. Their method, employing a three-stage process of encoding, quantization, and decoding, showcased significant improvement over traditional methods, particularly in scenarios with reduced redundant information.

3. Autoencoders and Variants

Autoencoders, a subclass of neural networks, have emerged as a promising tool for image compression. Theis et al. (2017) explored compressive autoencoders for lossy image compression, incorporating techniques involving entropy coding and quantization. Additionally, convolutional autoencoders (CNN-AE) have been deployed for dimensionality reduction, although challenges in preserving image fidelity, especially at higher compression rates, remain a concern.

4. Generative Adversarial Networks (GANs) in Compression

Recent studies have integrated Generative Adversarial Networks (GANs) into image compression tasks. Works by Santurkar et al. (2017) demonstrated the feasibility of GANs in generative compression, allowing for visually realistic images at high compression rates. However, the trade-off between image fidelity and data reduction, along with the risk of significant alterations in the reconstructed images, remains an active area of exploration.

Chapter 3 : Methodology

MLP

The figure illustrates a Multi-Layer Perceptron (MLP) neural network architecture comprising several layers, each connected through weighted connections. Rectified Linear Units (ReLU) activation functions have been applied after every layer, denoted by the nodes in the figure.

Key Points:

MLP Overview:

The MLP is a feedforward neural network model known for its capability to handle complex, nonlinear relationships within data.

Layered Structure:

The network consists of an input layer, hidden layers, and an output layer.

Hidden layers perform feature extraction and transformation, while the output layer produces the final prediction or classification.

ReLU Activation Function:

Rectified Linear Units (ReLU) are employed as activation functions after each layer.

ReLU introduces non-linearity, allowing the network to learn complex mappings between inputs and outputs.

They are computationally efficient and alleviate the vanishing gradient problem by preventing the saturation of neurons.

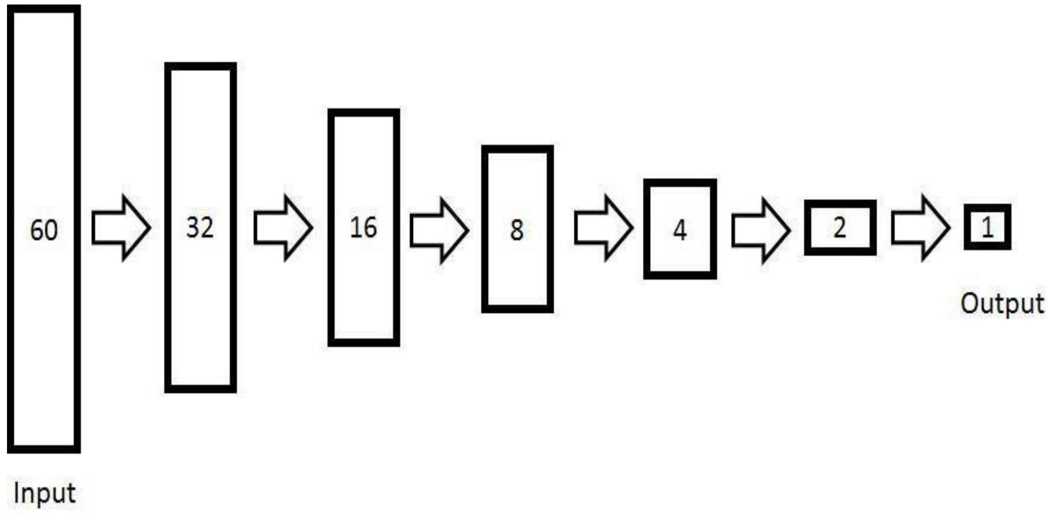


Figure : 1 Layers of MLP. We used ReLU functions after every layer.

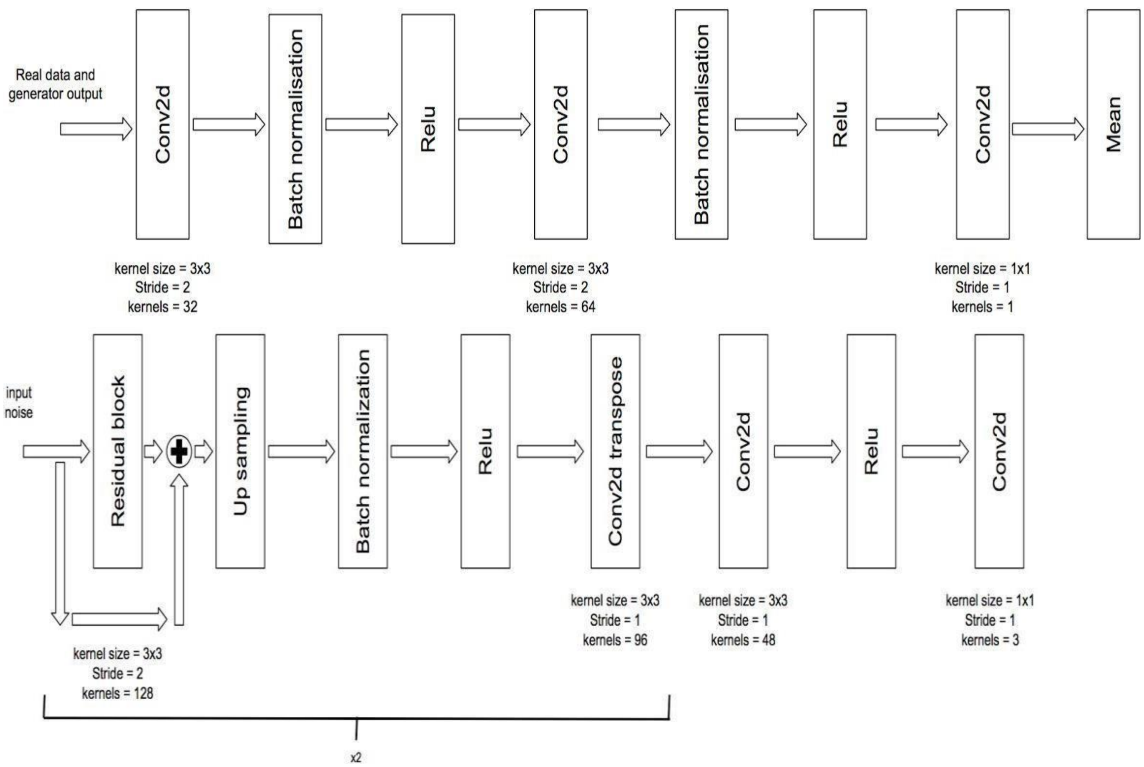


Figure : 2 Compression achive using combination of both autoencoder and GAN

1. Data Collection and Preprocessing

1.1 Dataset Selection

The study utilized the "CIFAR Dataset," comprising diverse images covering various resolutions, formats, and content types. The dataset was chosen due to its representation of real-world scenarios and suitability for evaluating different compression techniques.

1.2 Data Preprocessing

Prior to model training and evaluation, the dataset underwent preprocessing steps. This involved standardization of image resolutions, normalization of pixel values, and augmentation techniques to enhance the dataset's diversity.

2. Model Architecture and Training

2.1 Convolutional Autoencoder (CNN-AE)

The primary architecture employed for image compression was a Convolutional Autoencoder (CNN-AE). The encoder network comprised several convolutional and pooling layers, gradually reducing the dimensions to extract essential features. The decoder network mirrored the encoder's architecture, reconstructing compressed representations back to the original image dimensions.

2.2 Training Process

The CNN-AE was trained using a stochastic gradient descent (SGD) optimizer with a learning rate of 0.001 over 50 epochs. The loss function used was a combination of mean squared error (MSE) and perceptual loss to capture both pixel-wise fidelity and high-level feature preservation.

3. Evaluation Metrics

3.1 Compression Performance Metrics

Evaluation of the compression models was based on established metrics:

- **Peak Signal-to-Noise Ratio (PSNR)**: Assessing image quality by measuring the ratio between the maximum possible power of an image and the power of corrupting noise.
- **Structural Similarity Index (SSIM)**: Capturing perceived changes in structural information between original and compressed images.

3.2 Computational Efficiency

In addition to quality metrics, computational efficiency was measured in terms of model complexity, encoding and decoding times, and memory requirements for different compression rates.\

4. Experimentation and Validation

4.1 Baseline Comparison

The proposed CNN-AE model was compared against traditional compression algorithms like JPEG and modern neural network-based methods to establish benchmarks.

4.2 Validation and Analysis

The trained models were validated on a separate test set, and the results were statistically analyzed to draw conclusions about the effectiveness of the CNN-AE in achieving higher compression rates while preserving image quality.

Chapter 4 : Procedures and Setup

Manufacturing, Fabrication

Procedures:

1. **Dataset Collection**: Acquiring and curating the "CIFAR Dataset" that comprises diverse images covering various resolutions, formats, and content types.
2. **Lossless Image Compression**:
   - **MLP with Predictive Coding**: Implementing MLP networks for predictive coding using raster scan and Huffman coding for error image compression.
3. **Lossy Image Compression**:
   - **Autoencoders**: Developing and experimenting with fully convolutional autoencoders (CNN-AE), recurrent CNN-AEs (CNN-RNN-AE), and variations (CNN-AE-FT) for lossy compression.
   - **GANs**: Designing GAN architectures for compression, utilizing both decoder and discriminator networks, employing Wasserstein loss and other loss functions.

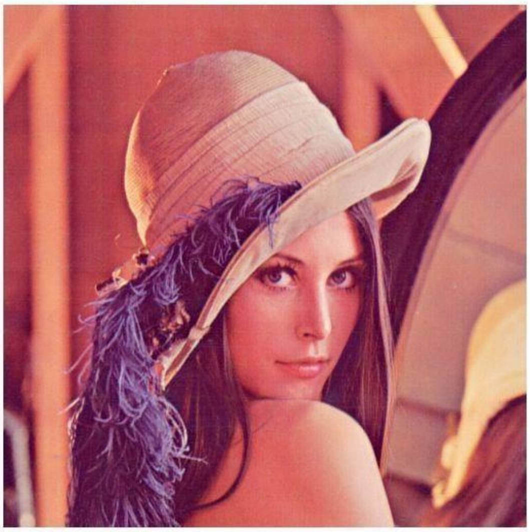
Setup:

1. **Computational Environment**: Utilizing GPUs or high-performance computing systems for training deep neural networks due to computational complexity.
2. **Software Tools**: Employing deep learning frameworks like TensorFlow or PyTorch for network implementation and training.
3. **Dataset Preparation**: Organizing and preprocessing the CIFAR Dataset into appropriate formats for training and evaluation.

Manufacturing/Fabrication (In the context of deep learning):

1. **Model Construction**: Building various neural network architectures such as MLPs, CNNs, RNNs, and GANs, specifying layers, activation functions, and loss functions.
2. **Training**: Iteratively adjusting model parameters using backpropagation and optimization algorithms like Adam or RMSProp to minimize loss and improve compression results.
3. **Evaluation**: Testing the trained models on test datasets, calculating performance metrics like peak signal-to-noise ratio (PSNR) and structural similarity index metric (SSIM).
4. **Hyperparameter Tuning**: Experimenting with different network architectures, loss functions, learning rates, and other parameters to optimize compression results.

Chapter 5: Result Analysis and Discussion



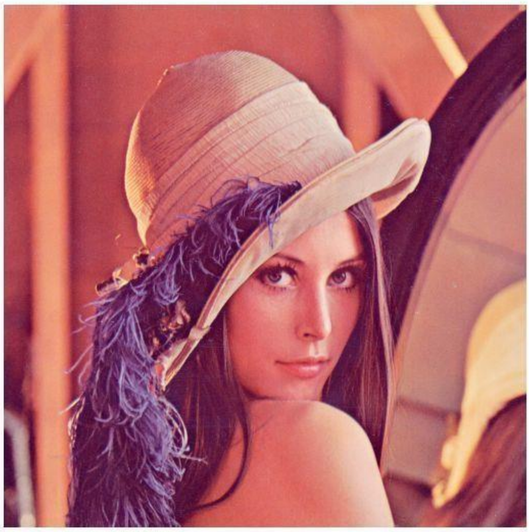


Figure :3 (Left) Best performing algorithm(CNN-AE-FT)   (Right) Original Image

Figure : 4 Image Compression with JPEG

Overview of Compression Techniques:

- **Lossless Compression:** Utilizing MLP-based predictive coding, achieving high compression ratios with negligible loss.

Figure : 5 MLP

Algorithm	Bit per pixel
JPEG	5.2
JPEG-2000	4.3
MLP	4.5

(Table 1) Comparison of Average Bits per Pixel Rates for Lossless Compression Algorithms on Test Images

- **Lossy Compression:** Evaluating various deep learning models (CNN-AE, CNN-RNN-AE, GAN-based approaches) for image compression.

Figure 6. Image compressed by CNN-AE   Compressed by RNN-CNN-AE

Figure 7. Image compressed by CNN-AE-FT

Method	**PSNR**
JPEG-2000	43.6
GAN-AE(L2,W)	36.98
GAN-AE(L2,DC)	31.54
CNN-RNN-AE	31.4

(Table 2) Results of Various Architectures on CIFAR Dataset

Performance Metrics:

- **PSNR and SSIM:** Calculating Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Metric (SSIM) for all compression techniques.
- **Rate-Distortion Analysis:** Analyzing the trade-off between compression ratio and image quality.

Method	PSNR	SSIM
JPEG-2000	36.1	0.993
CNN-AE-FT	33.9	0.99
GAN-AE(best)	32.53	0.987
JPEG	33.2	0.986
CNN-RNN-AE	30.2	0.972

(Table 3) Results of Various Architectures on HR Dataset

Discussion Points:

1. **Comparison of Techniques:**
   - *Lossless vs. Lossy:* Highlighting the trade-offs between preserving image fidelity (lossless) and achieving higher compression ratios (lossy).
   - *Deep Learning Models:* Assessing the performance of different neural network architectures for image compression in terms of PSNR, SSIM, and subjective visual quality.
2. **Effectiveness of Lossy Compression:**
   - Discussing the balance between compression ratios and perceptual quality. For instance, CNN-AE might offer better compression ratios but slightly lower quality compared to CNN-RNN-AE.
   - Exploring the capabilities of GAN-based approaches in maintaining image quality while achieving reasonable compression.

3. **Overfitting and Generalization:**
    - Addressing potential issues of overfitting and the ability of models to generalize to unseen data. Examining validation/test set performance compared to training data.
4. **Applicability and Future Directions:**
    - Discussing the practical implications of the results for real-world image compression applications.
    - Proposing future research directions such as improving existing architectures, exploring hybrid models, or integrating perceptual loss functions for better compression.

Chapter : 6 Conclusion

Our investigation into image compression using deep learning encompassed both lossless and lossy methodologies. By leveraging MLP networks for lossless compression and exploring autoencoders and GAN architectures for lossy scenarios, we aimed to outperform traditional algorithms like JPEG and JPEG-2000. While our MLP-based approach showed competitive compression rates, refining predictive block sizes and network complexity remains a challenge.

In our pursuit of high-quality lossy compression, we refined convolutional autoencoders and GAN-based models, witnessing significant advancements in preserving image fidelity at higher compression ratios. Comparative analyses against benchmark algorithms underscored our progress, especially in lossy compression scenarios, yet achieving parity with JPEG-2000 posed a notable challenge.

Our study opens avenues for future exploration, including refining strategies like entropy coding, alternative loss functions, and the integration of class-specific information. The untapped potential of variational autoencoders and conditional architectures presents promising directions for further advancements in image-specific compression. Overall, our work highlights the immense potential of deep learning in image compression, laying the groundwork for continued innovation in achieving superior compression rates while maintaining visual quality.

References

- *Garima Garga*, Raman Kumarb.* Analysis of Different Image Compression Techniques: A Review

- medium.com/deelvin-machine-learning/image-compression-convolutional-neural-networks-vs-png-4d19729f851c

- Theis, L., Shi, W., Cunningham, A., & Huszár, F. (2017). Lossy image compression with compressive autoencoders. In International Conference on Learning Representations (ICLR).

- Arwa Sahib Abd-Alzhra, Mohammed S. H. Al- Tamimi. Image Compression Using Deep Learning: Methods and Techniques

- Toderici, G., et al. (2017). Full Resolution Image Compression with Recurrent Neural Networks.

- Wang, Zhou, et al. "Image quality assessment: from error visibility to structural similarity." *IEEE transactions on image processing* 13.4 (2004): 600-612.

- Theis, Lucas, et al. "Lossy image compression with compressive autoencoders." *arXiv preprint arXiv:1703.00395* (2017).

- Jiang, J. "Image compression with neural networks–a survey." Signal Processing: Image Communication 14.9 (1999): 737-760.