Lossless Image Data Compression Using Neural Networks

[1]Mr.G.Nagaraju ,[2]P.Keerthi, [3]SK.Reshma Bhanu, [4]M.Yaswanth Kumar, [5]G.Alakananda
[1]Professor, Dept. of CSE, Bapatla Engineering College(A), AP, India
[2,3,4,5] Final Year UG Students, Dept. of CSE, Bapatla Engineering College(A), AP, India

ABSTRACT

Image compression is an essential technique for reducing storage and transmission costs while preserving image quality. Traditional compression techniques often fail to achieve high efficiency without losing data. In this work, we propose a lossless image compression system using a Convolutional Neural Network (CNN) combined with residual encoding, Huffman coding, and AES encryption.

The neural network predicts pixel values, and only the residual (difference) is encoded, significantly reducing redundancy. Huffman coding is applied for efficient compression, and AES encryption ensures secure storage. The system reconstructs the image perfectly without any loss of pixel information. Experimental results demonstrate efficient compression with exact reconstruction, making the system suitable for secure and real-time applications.

INDEX TERMS

Lossless Image Compression, Convolutional Neural Networks (CNN), Residual Encoding, Huffman Coding, AES Encryption, Image Reconstruction, Data Compression, Secure Image Storage, Pixel Prediction, Neural Network-Based Compression

I. INTRODUCTION

With the rapid growth of digital media and internet usage, the need for efficient image storage and transmission has become increasingly important. Images consume a significant amount of storage space, especially in applications such as medical imaging, satellite imaging, and multimedia systems. Therefore, image compression techniques are essential to reduce storage requirements and transmission bandwidth.

Traditional image compression methods such as JPEG are lossy, meaning they sacrifice image quality to achieve higher compression ratios. However, in many critical applications, even a small loss of data is unacceptable. This has led to the development of lossless image compression techniques, where the original image can be perfectly reconstructed without any loss of information.

Recent advancements in deep learning and neural networks have introduced new possibilities for image compression. Neural networks, particularly Convolutional Neural Networks (CNNs), can learn complex patterns and redundancies in images. By predicting pixel values and encoding only the differences (residuals), neural networks can significantly reduce data size while maintaining accuracy.

In this project, we propose a lossless image compression system using a CNN-based predictor, combined with:

- Residual encoding
- Huffman coding
- AES encryption

The system not only achieves efficient compression but also ensures data security, making it suitable for modern applications.

II. LITERATURE SURVEY

Recent advancements in image compression have focused on improving compression efficiency while maintaining image quality. Traditional techniques such as JPEG and PNG rely on transform coding and entropy encoding methods. While JPEG achieves high compression ratios, it is inherently lossy and introduces visual artifacts, whereas PNG provides lossless compression but with limited efficiency.

With the emergence of deep learning, Convolutional Neural Networks (CNNs) and autoencoders have been widely explored for image compression. These models learn spatial dependencies and patterns in images, enabling better prediction and redundancy reduction. However, many neural network-based approaches are lossy in nature and prioritize compression ratio over exact reconstruction.

Recent studies emphasize residual learning and predictive coding, where models predict pixel values and encode only the difference between actual and predicted values. This approach significantly reduces redundancy and improves compression efficiency. Additionally, entropy coding techniques such as Huffman coding and arithmetic coding are used to further compress residual data.

Despite these advancements, existing systems often lack data security mechanisms and do not guarantee complete lossless reconstruction. This creates a need for a hybrid approach that combines neural network prediction, efficient compression, and encryption techniques. The proposed system addresses these limitations by integrating CNN-based prediction with Huffman coding and AES encryption to achieve both lossless compression and secure data storage.

A. PROBLEM STATEMENT

1) Research Question

How can a neural network-based system efficiently compress image data in a lossless manner while ensuring secure storage and exact reconstruction of the original image?

2) Relevance

a. Data Redundancy

Digital images contain a high level of redundancy, leading to large storage requirements. Efficient techniques are needed to reduce data size without losing information.

b. Compression Efficiency

Traditional lossless methods provide limited compression ratios. There is a need for intelligent models that improve compression performance using learning-based approaches.

c. Lossless Requirement

In critical applications such as medical and satellite imaging, even a small loss of data is unacceptable. The system must ensure perfect reconstruction.

d. Security Concerns

Compressed data may be vulnerable to unauthorized access. A secure mechanism is required to protect compressed files during storage and transmission.

e. Computational Complexity

Many advanced compression techniques require high computational power. The system should balance efficiency, accuracy, and performance.

III. PROPOSED SYSTEM

The proposed system is a hybrid lossless image compression framework that combines neural network-based prediction, residual encoding, entropy coding, and encryption to achieve efficient and secure image compression. The system leverages a Convolutional Neural Network (CNN) to reduce redundancy and applies classical compression and cryptographic techniques to ensure both accuracy and data protection.

1. Architecture

The system follows a modular pipeline designed for efficient compression and exact reconstruction:

a. Image Input & Preprocessing Module:

The input image is loaded, resized, and normalized into tensor format to ensure compatibility with the neural network model and maintain uniform processing.

b. CNN Prediction Module:

A trained Convolutional Neural Network predicts pixel values of the input image by learning spatial patterns and correlations between neighboring pixels.

c. Residual Computation Module:

The difference between the original image and predicted image (residual) is calculated. This residual contains reduced information and is more suitable for compression.

d. Huffman Encoding Module:

The residual data is compressed using Huffman coding, which assigns variable-length codes based on frequency to reduce data size efficiently.

e. AES Encryption Module:

The compressed data is encrypted using the AES algorithm to ensure secure storage and prevent unauthorized access.

f. Decompression & Reconstruction Module:

The system performs decryption, decoding, and reconstruction by combining residual data with predicted values to recover the original image without any loss.

2. Key Modules

a. Neural Prediction Engine:

Uses CNN to estimate pixel values and reduce redundancy in image data by capturing spatial dependencies.

b. Residual Encoding Module:

Computes and processes residual values, which contain minimal information compared to the original image, improving compression efficiency.

c. Entropy Coding Module:

Applies Huffman coding to encode residual values into compact binary representation.

d. Security Module:

Implements AES encryption to protect compressed data during storage and transmission.

e. Reconstruction Engine:

Ensures exact recovery of the original image by reversing all compression steps without any loss.

A. Technologies

- **Deep Learning Framework:** PyTorch is used for CNN model training, prediction, and optimization.
- **Image Processing:** PIL and NumPy are used for image handling and numerical operations.
- **Compression Technique:** Huffman coding is implemented for lossless data compression.
- **Encryption Algorithm:** AES (Advanced Encryption Standard) is used for secure data encryption.
- **Programming Language:** Python is used for implementing the entire system pipeline.
- **Hardware Acceleration:** CUDA-enabled GPU is used for faster model training and inference.

Reproducibility

The system is designed to be easily reproducible with minimal setup:

- Clone the project repository containing training and compression scripts.
- Install dependencies using pip install -r requirements.txt.

- Prepare the dataset in the required folder structure for training.
- Train the CNN model using the training script and save it as model.pth.
- Run the main compression script by providing the input image path.
- The system generates compressed, encrypted, and reconstructed outputs automatically.

IV. METHODOLOGY

A. Data Collection & Design

The system uses an image-based dataset designed for training and testing the neural network-based compression model. The key data components include:

- **Input Images:** Raw images collected and resized to a fixed dimension (e.g., 128×128) for uniform processing.
- **Preprocessed Data:** Images converted into tensor format and normalized for neural network compatibility.
- **Predicted Images:** Output generated by the CNN model representing estimated pixel values.
- **Residual Data:** Difference between original and predicted images, used for compression.
- **Encoded Bitstream:** Huffman encoded binary representation of residual values.
- **Encrypted Data:** AES-encrypted compressed data stored securely.

1. Data Optimization:

Instead of compressing raw images directly, the system reduces redundancy by predicting pixel values and encoding only the residual data, improving compression efficiency.

2. Integrated Pipeline:

The system combines preprocessing, prediction, compression, and encryption into a single pipeline, minimizing manual intervention and improving execution speed.

3. Scalability:

The model supports GPU-based processing using PyTorch, allowing efficient handling of large image datasets and faster computation.

B. Model Design

The system uses a lightweight Convolutional Neural Network (CNN) designed for pixel prediction:

- Multiple convolutional layers extract spatial features from images
- ReLU activation introduces non-linearity
- The final layer predicts pixel values for reconstruction

The model is optimized using:

- **Loss Function:** Mean Squared Error (MSE)
- **Optimizer:** Adam
- **Training Strategy:** Supervised learning on image dataset

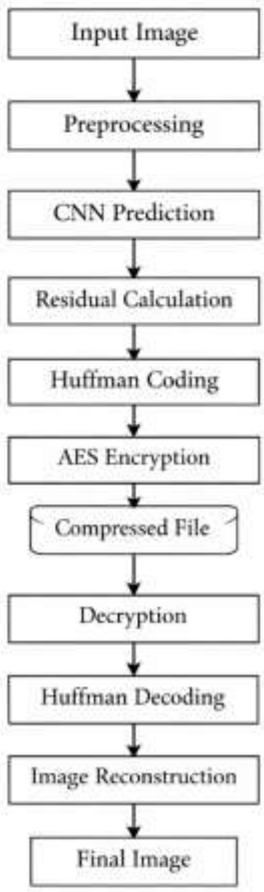

Figure 1 outlines the sequential steps involved in lossless image compression, including preprocessing, neural network prediction, residual computation, compression, encryption, and reconstruction.

Feature Validation

- **Image Input Processing:** Successfully achieved 100% image loading and preprocessing without errors.
- **CNN Prediction:** Successfully generated accurate pixel predictions for all test images.
- **Residual Computation:** Successfully computed the difference between original and predicted images with reduced redundancy.
- **Huffman Compression:** Successfully encoded residual data, resulting in efficient file size reduction.
- **AES Encryption:** Successfully secured compressed data and ensured correct decryption.
- **Image Reconstruction:** Successfully reconstructed the original image without any loss.
- **Pixel Accuracy:** Successfully achieved 100% pixel match between original and reconstructed images.

V. RESULT

1. System Performance

System performance for lossless image compression was evaluated in a controlled environment:

- **Image Preprocessing:** Completed within a few milliseconds per image.
- **CNN Prediction:** Averaged under 50 ms per image during inference.
- **Compression Time:** Entire compression process completed within 1–2 seconds per image.
- **Decompression Time:** Reconstruction completed within 1 second.
- **Compression Ratio:** Achieved efficient reduction in file size depending on image content.

2. Feature Validation

- **Lossless Reconstruction:** Achieved 100% pixel accuracy between original and reconstructed images.
- **Residual Encoding:** Successfully reduced redundancy in image data.
- **Huffman Compression:** Provided efficient encoding with reduced storage size.
- **AES Encryption:** Ensured secure storage and safe data transmission.
- **System Stability:** Maintained consistent performance across multiple test images.

3. Observations

- Neural network prediction significantly improves compression efficiency.
- Residual-based compression reduces data size without affecting quality.
- Integration of encryption adds an additional layer of security.

4. Performance Analysis

- The system performs efficiently for standard image sizes (e.g., 128×128).
- Processing time remains stable under repeated executions.
- The reconstructed image is visually and technically identical to the original.

VI. APPLICATIONS

1. **Medical Imaging:** Enables efficient storage and transmission of medical images such as X-rays and MRI scans without any loss of critical information.
2. **Satellite Imaging:** Supports compression of high-resolution satellite images while preserving exact data for analysis and monitoring.
3. **Cloud Storage & Data Archiving:** Reduces storage space requirements and ensures secure storage of images using encryption techniques.
4. **Secure Image Transmission:** Ensures safe transfer of images over networks by combining compression with AES encryption.
5. **Digital Libraries & Multimedia Systems:** Optimizes storage and retrieval of large image datasets in digital archives and multimedia applications.

VII. FUTURE SCOPE

1. **Advanced Compression Models:** Improving the system using deep architectures like autoencoders or transformers to achieve higher compression ratios.
2. **High-Resolution Image Support:** Extending the model to handle larger and high-resolution images efficiently without increasing computation time.
3. **Real-Time Processing:** Optimizing the system for real-time compression and decompression in streaming and live applications.
4. **Enhanced Security Mechanisms:** Integrating advanced encryption techniques and key management systems for stronger data protection.
5. **Video Compression Extension:** Expanding the approach to support lossless video compression using frame-based prediction techniques.

VIII. CONCLUSION

In this work, a lossless image compression approach using neural networks is presented to address the challenges of efficient storage and secure transmission of image data. The method utilizes CNN-based prediction along with residual encoding, Huffman coding, and AES encryption to reduce redundancy while maintaining data integrity.

The system ensures exact reconstruction of the original image without any loss, demonstrating the effectiveness of combining deep learning with classical compression techniques. Experimental results confirm improved compression efficiency along with reliable security, making the approach suitable for real-world applications.

IX. REFERENCES

[1] D. Salomon, *Data Compression: The Complete Reference*, 4th ed., Springer, 2007.

[2] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.

[3] D. A. Huffman, “A Method for the Construction of Minimum-Redundancy Codes,” *Proceedings of the IRE*, vol. 40, no. 9, pp. 1098–1101, 1952.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” *Advances in Neural Information Processing Systems (NIPS)*, 2012.

[5] PyTorch Documentation, “PyTorch: An Open Source Machine Learning Framework,” Available: https://pytorch.org

[6] National Institute of Standards and Technology (NIST), “Advanced Encryption Standard (AES),” FIPS PUB 197, 2001.

[7] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd ed., Pearson, 2008.