**Image Compression Using Convolutional Neural Networks**

**Group members:**

Atif Iqbal(65477)

Kinza Shahzad (64750)

**Introduction:**

**What is Image Compression?**

Image compression is the process of reducing the size of an image file without significantly affecting its quality. This process is crucial in various applications, such as web usage, image storage, and transmission over networks. By reducing the amount of data required to represent the image, storage space is saved and transmission times are reduced.

**Importance**

With the increase in the use of high-resolution images, efficient image compression techniques have become essential. These techniques help in saving storage space and bandwidth, thus enabling faster transmission and efficient storage.

**Objective**

This report presents the implementation and evaluation of a Convolutional Neural Network (CNN)-based image compression model using the CIFAR-10 dataset.

**Methodology**

**Data Preparation**

The CIFAR-10 dataset, consisting of 60,000 32x32 color images in 10 different classes, was used for training and testing the model. The dataset was normalized to ensure consistent input for the model, which helps in improving the model's performance and stability.

**Model Design**

The model, named LappedTransformCNN, consists of a convolutional layer for encoding and a transpose convolutional layer for decoding. The encoding layer compresses the image by reducing its dimensions, while the decoding layer reconstructs the image back to its original dimensions. This two-step process enables effective image compression and decompression.

**Training**

The model was trained using the Mean Squared Error (MSE) loss function, which measures the average squared difference between the original and reconstructed images. The Adam optimizer, known for its efficiency and good convergence properties, was used to update the model parameters. The training process was carried out over 10 epochs, during which the model learned to minimize the reconstruction error.

**Evaluation**

After training, the model was evaluated on the test set to compute the average test loss. This evaluation helped in understanding the model's performance on unseen data and ensured that the model generalized well.

**Image Compression and Decompression**

**Compression Process**

The image compression process involves several steps:

1. **Transform the Image:** The image is transformed using the convolutional layer of the model.
2. **Quantize the Transformed Image:** The transformed image is quantized to reduce the number of levels, which helps in saving space.
3. **Convert to Byte Stream:** The quantized image is converted to a byte stream for efficient storage.

**Decompression Process**

The decompression process involves reversing the compression steps:

1. **Reconstruct Quantized Image:** The byte stream is converted back to the quantized image.
2. **Dequantize:** The quantization process is reversed to obtain the dequantized image.
3. **Use Transpose Convolution:** The dequantized image is processed using the transpose convolution layer to recover the original image.

**Example Compression and Decompression**

An example image was compressed and decompressed to visualize the results. The original and reconstructed images were compared to evaluate the quality of compression.

**Results and Evaluation**

**Visualization**

Original and reconstructed images were displayed side by side to visually assess the quality of compression. The comparison helped in understanding how well the model preserved the image quality during compression and decompression.

**Compression Ratio**

The compression ratio was calculated by comparing the sizes of the original and compressed images. This ratio indicates the effectiveness of the compression technique in reducing the image size.

**LPIPS (Learned Perceptual Image Patch Similarity)**

The LPIPS value was computed to measure the perceptual similarity between the original and reconstructed images. This metric helps in assessing the quality of the reconstructed image from a perceptual perspective, which is more aligned with human visual perception.

**Conclusion**

**Summary**

The implemented CNN-based model effectively compressed and decompressed images, achieving a significant compression ratio while maintaining a reasonable level of image quality as measured by the LPIPS metric. This demonstrates the potential of using CNNs for image compression tasks.

**Results Comparison:**

A collage of several sailboats

Description automatically generated ****

**A collage of a sailboat

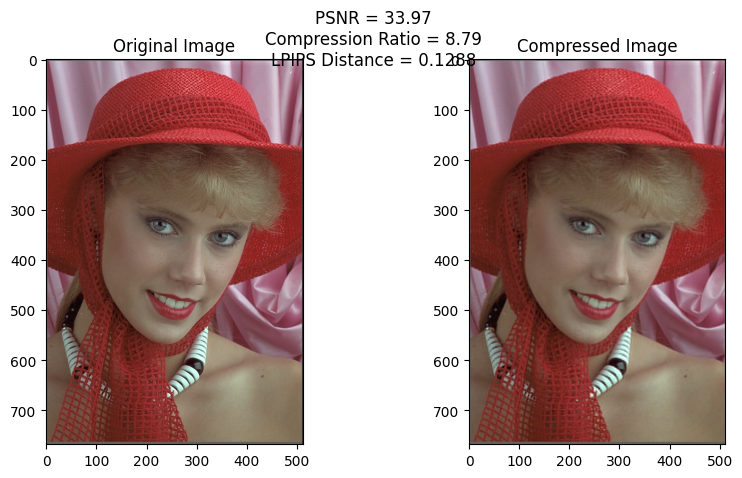
Description automatically generatedA collage of a sailboat

Description automatically generated**

A comparison of a boat in water

Description automatically generatedA boat in the water

Description automatically generated

A collage of a person wearing a red hat

Description automatically generated

A group of people on a raft

Description automatically generatedA collage of people on a raft

Description automatically generated