Image Compression Project

Saksham Rathi, Kavya Gupta, Shravan Srinivasa Raghavan
(22B1003) (22B1053) (22B1054)

CS663: Digital Image Processing
Under Prof. Ajit Rajwade

Indian Institute of Technology Bombay
Autumn 2024

Contents

Problem Statement

The problem statement of this project has been taken from the following website:

CS663: Digital Image Processing

We have built an image compression engine along the lines of the JPEG algorithm. Along with this, we have implemented PCA algorithm. We have also thoroughly studied a tier-1 conference paper **Approximation and Compression With Sparse Orthonormal Transforms** and implemented the algorithm proposed in the paper.

All the algorithms were tested on a variety of image datasets. The results were compared and analyzed to understand the performance of the algorithms.

Basic Implementation

Here are the steps which were performed as part of the basic implementation:

- Computation of the 2D DCT coefficients of non-overlapping image patches
- Implementation of the quantization step
- Implementation of the Huffman tree
- Writing data to an appropriate file format (.bin) and plotting RMSE vs BPP

Here is the expression of RMSE:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M}(I_{\text{orig}}(i)(j) - I_{\text{recon}}(i)(j))^2} \tag{1}$$

where I_{orig} is the original image and I_{recon} is the reconstructed image. BPP stands for the size of the image in bits divided by the number of pixels.

Quality Comparison

Here is the comparison of the reconstucted and the original image for a quality factor of 2:

../results/Quality: 2_comparison.png

Quality Comparison

Here is the comparison of the reconstucted and the original image for a quality factor of 10:

../results/Quality: 10_comparison.png

Quality Comparison

Here is the comparison of the reconstucted and the original image for a quality factor of 50:

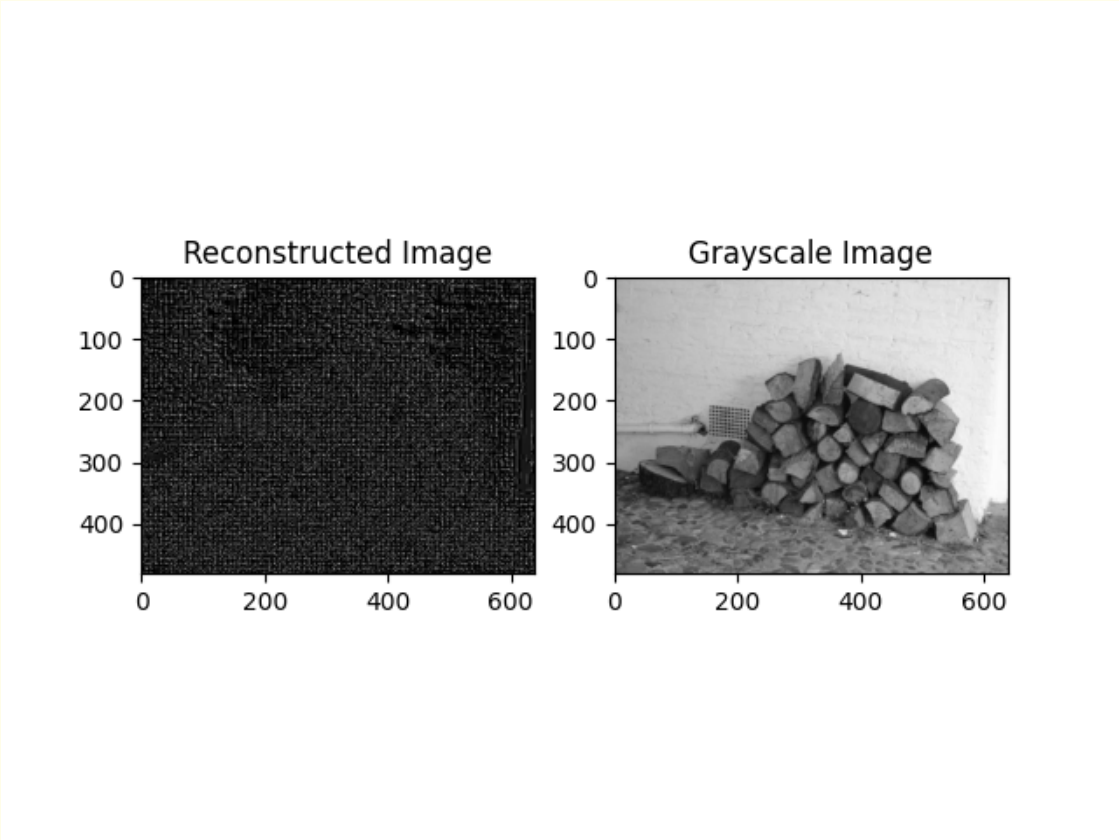

Figure: Original and Reconstructed Image

Quality Comparison

Here is the comparison of the reconstucted and the original image for a quality factor of 80:

../results/Quality: 80_comparison.png

RMSE vs BPP

For the basic implementation, we have used the dataset from the miscellaneous category of the msrcorid dataset. We picked random 20 images and used 20 quality factors (in the range of 1 to 100) to plot the RMSE vs BPP graph. Here is the graph:

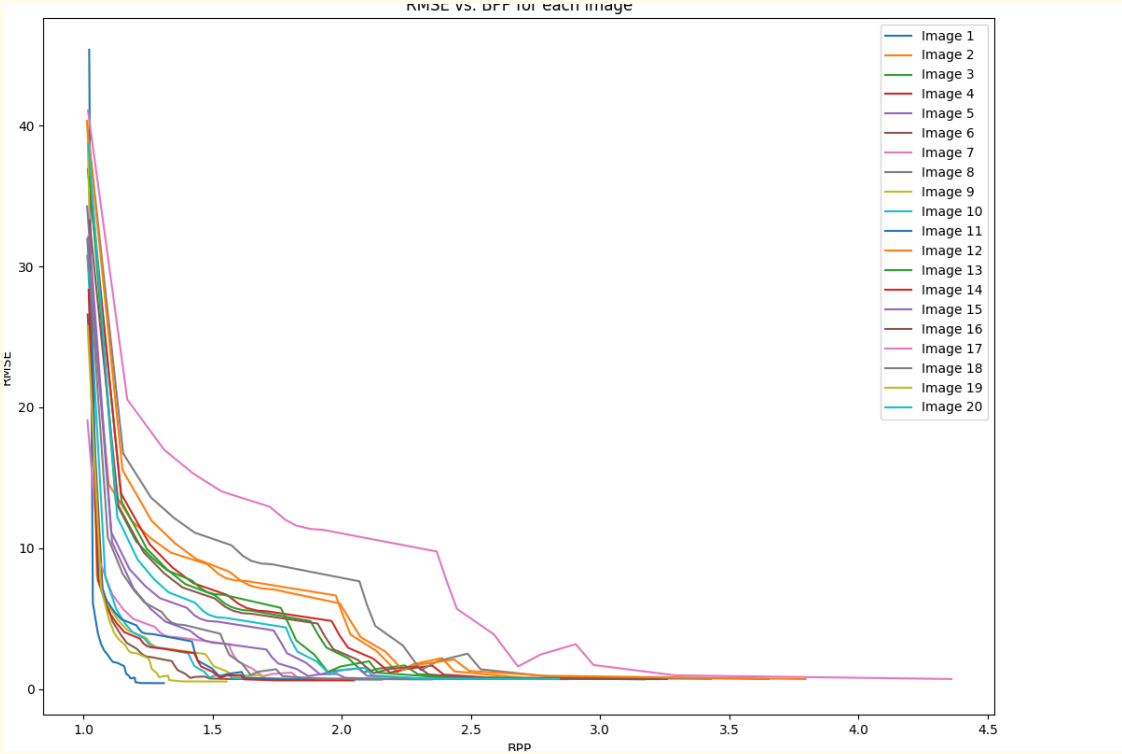

Figure: RMSE vs BPP

Run Length Encoding

The quantized DCT coefficients are arranged in a zigzag order. This pattern leaves a bunch of consecutive zeros at the end.

In runlength encoding, we replace the consecutive zeros with a pair of numbers: the number of zeros and the value of the next non-zero element. This reduces the size of the data to be stored.

RMSE vs BPP

The dataset of images from the miscellaneous category of the msrcorid dataset was used to plot the RMSE vs BPP graph. Here is the graph:

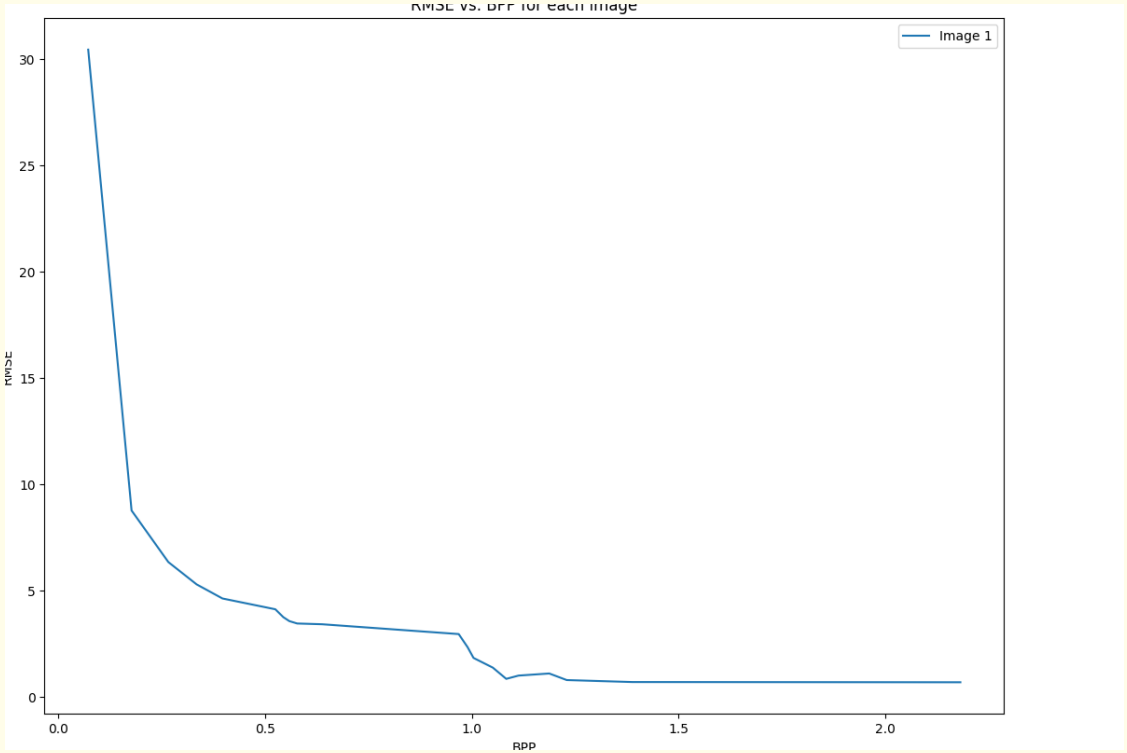

Figure: RMSE vs BPP

Comparison of Basic and Run Length Encoding

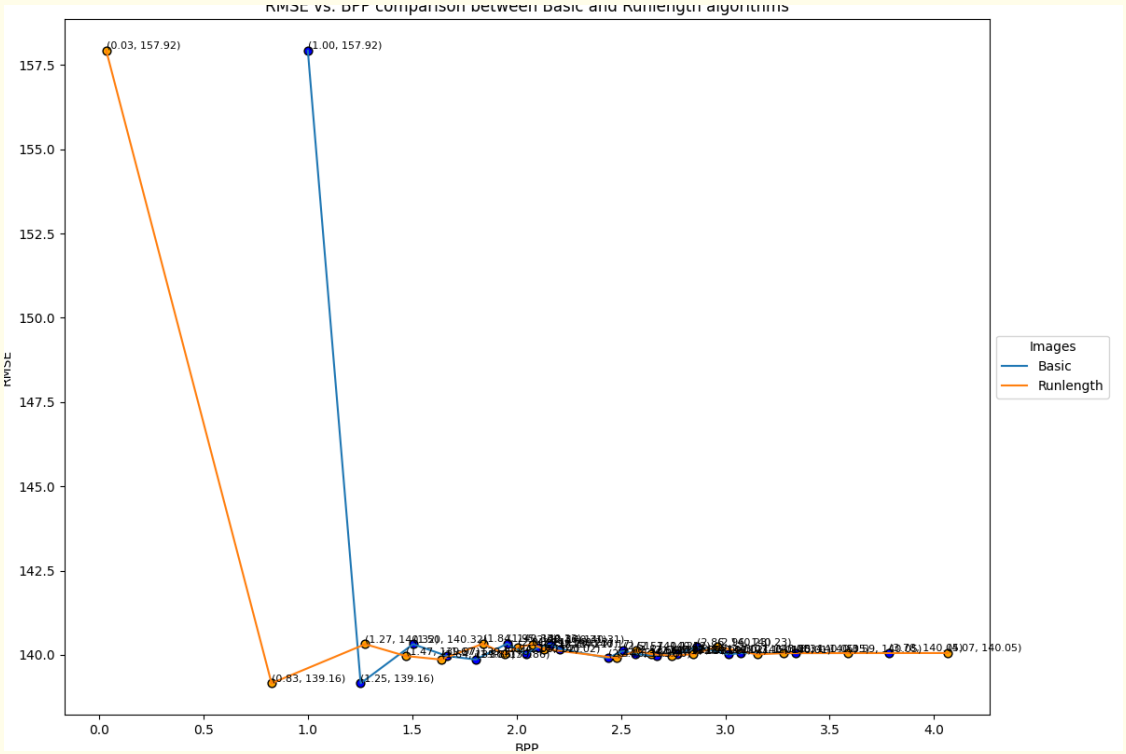

Figure: Comparison of Basic and Run Length Encoding

Individual Contributions

Saksham Rathi

Kavya Gupta

Shravan Srinivasa Raghavan

Conclusion

References

- CS663: Image Compression Slides
- Course Textbook: "Digital Image Processing" by Rafael C. Gonzalez and Richard Woods, 3rd edition
- Osman Gokhan Sezer, Onur G. Guleryuz and Yucel Altunbasak, "Approximation and Compression With Sparse Orthonormal Transforms", IEEE Transactions on Image Processing, 2015
- Sample Image Compression Code