Post-transform: Quantization and Entropy in the JPEG Pipeline

Joshua Lee, Indy Lyness, Luca Araujo, Jiale Wan; Advisor: Layla Oesper

Introduction

We explored the effects of applying different **transform coding** techniques to the JPEG pipeline (see fig. 1.0) to explore their effects on image compression across a variety of metrics. Namely, DCT (JPEG's choice), DFT, Haar, and S+P transforms were used.

One aspect of our project was to explore different ways in which **quantization** and **entropy coding** are affected by different transform coding techniques applied upstream in the pipeline and how the combinations of these compression techniques would affect the compression of images.

Background

Data compression is the process of representing data using less information than what used initially. **Transform coding**, **quantization**, and **entropy coding** are all examples of data compression techniques.

Transform coding algorithms, based on mathematical **transforms**, alter the basic data of an image to effect a higher efficiency in compression algorithms further down the compression pipeline.

References

1. Khayam, Syed Ali. "The Discrete Cosine Transform (DCT): Theory and Application." Michigan State University, vol. 114, no. 1, 2003, p. 31
2. Said, Amir, and William A. Pearlman. "Reversible Image Compression via Multiresolution Representation and Predictive Coding." Visual Communications and Image Processing '93, vol. 2094, 1993, pp. 664–74, https://doi.org/10.1117/12.157984.
3. Sayood, Khalid. Introduction to Data Compression. 4th ed., Elsevier, 2012, https://doi.org/10.1016/B978-0-12-415796-5.00001-6.
4. Wallace, Gregory K. "THE JPEG STILL PICTURE COMPRESSION STANDARD." IEEE Transactions on Consumer Electronics, vol. 38, no. 1, Feb. 1992.

Image Compression Pipeline

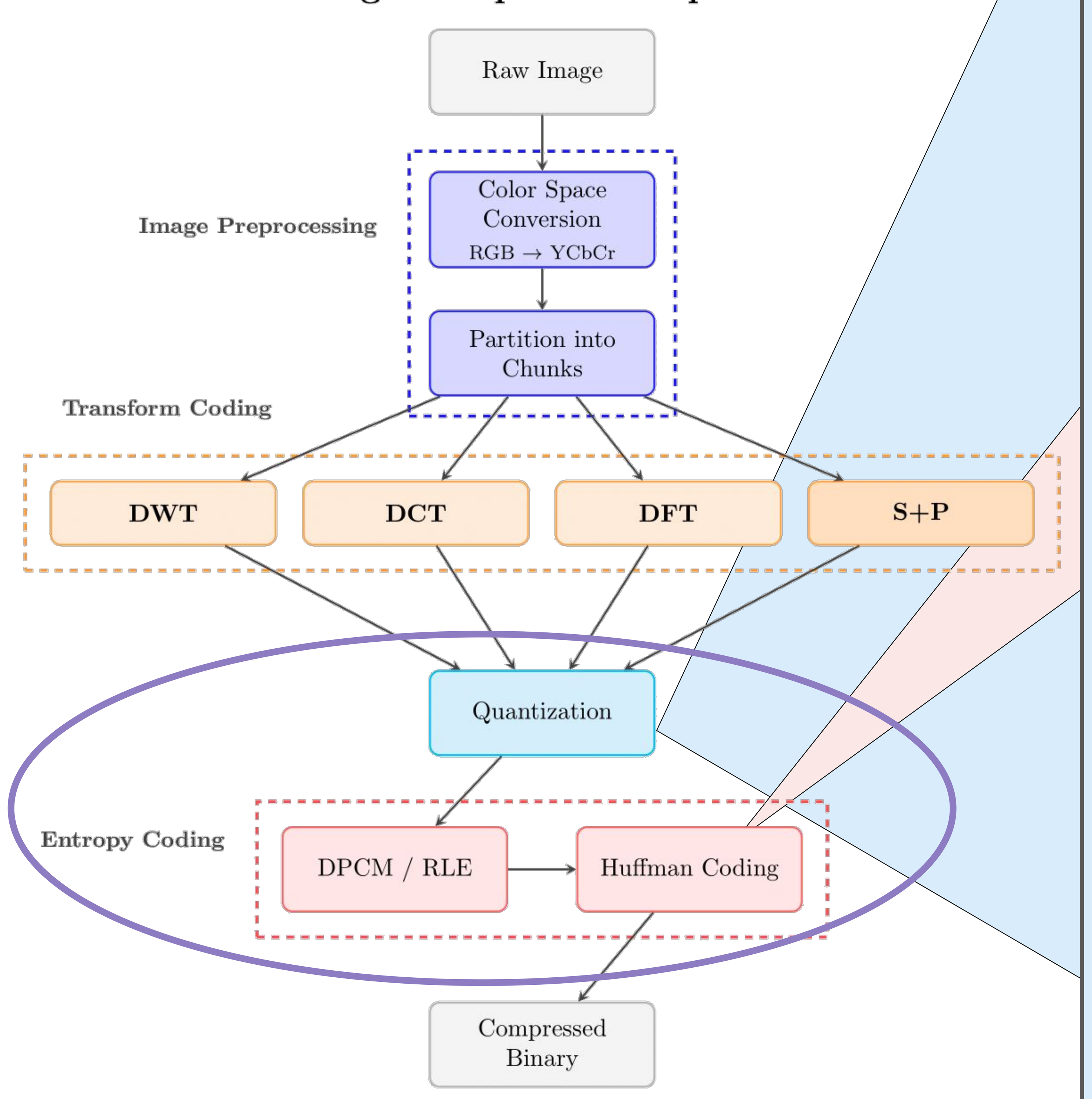

Fig. 1.0: Project pipeline, based on the JPEG Compression Std.

Quantization

Quantization is the process of "squashing" data to a smaller range of values, permanently losing data in the process. This generally involves dividing and rounding each datapoint, resulting in a less accurate version of the original, a tradeoff that allows space saving by reducing the number of possible values needed to construct the data.

Quantization in Action

These are results of different levels of quantization after applying the DCT transform, paired with "difference images" (darker is better).

Resultant Image | *Difference Image*


Fig. 2.0: Uncompressed



Fig. 2.1: Compression Ratio = 15



Fig. 2.2: CR = 32



Fig. 2.3: CR = 35



Fig. 2.4: CR = 36


Entropy Coding

Two algorithms used in this step are **Run-Length Encoding** (RLE) and **Differential Pulse Code Modulation** (DPCM). The results from these then get fed into a **Huffman Coding** algorithm to create the compressed file.

Run-Length Encoding

RLE groups digits with # of preceding 0s

DPCM & Predictive Coding

DPCM encodes the difference between adjacent values; Predictive coding uses previous values to predict the next, then encodes the difference between the predicted and actual value.

These algorithms are especially effective for the DCT transform due to its structure. However, different techniques may be better suited for other transforms!

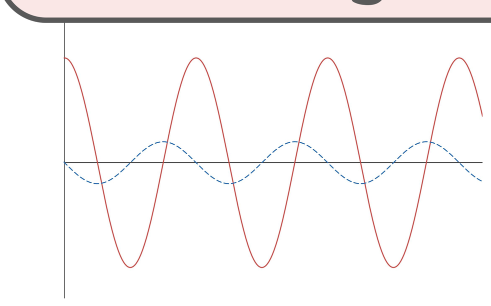

Fig 3.0: DPCM illustration. Blue wave represents difference values of red wave.

Results

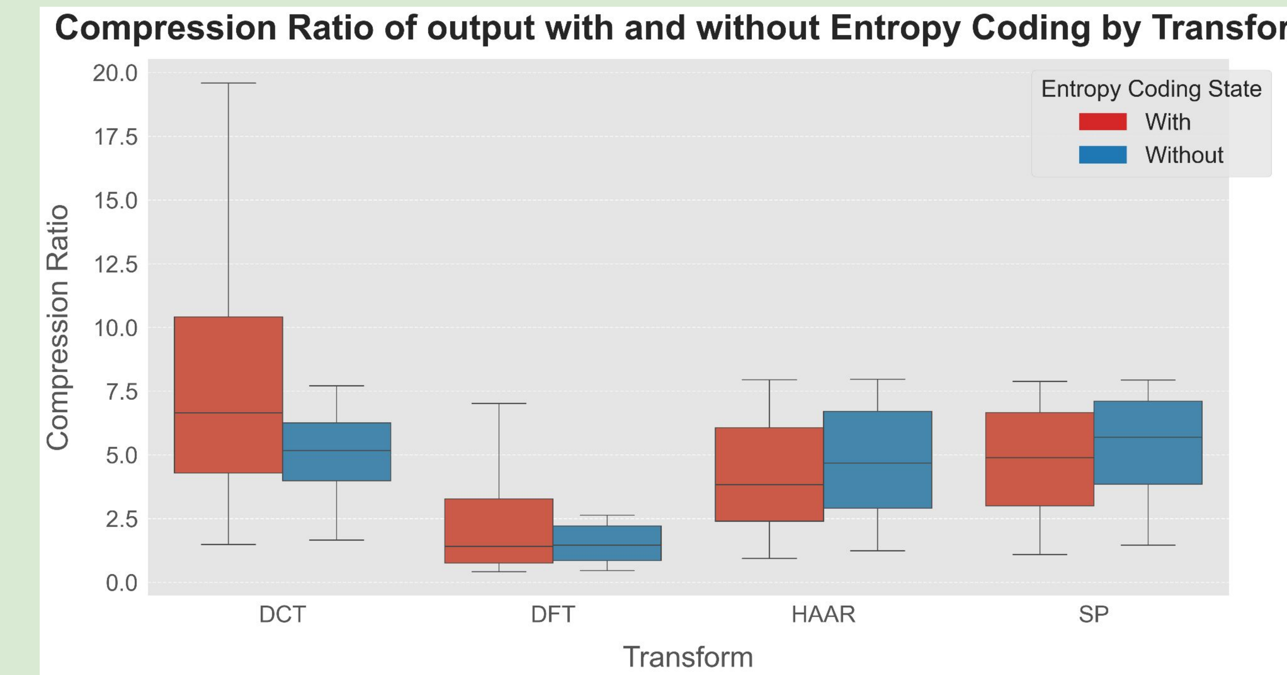

Fig. 4.0: Comparison of compression ratios with and without entropy encoding across transforms.

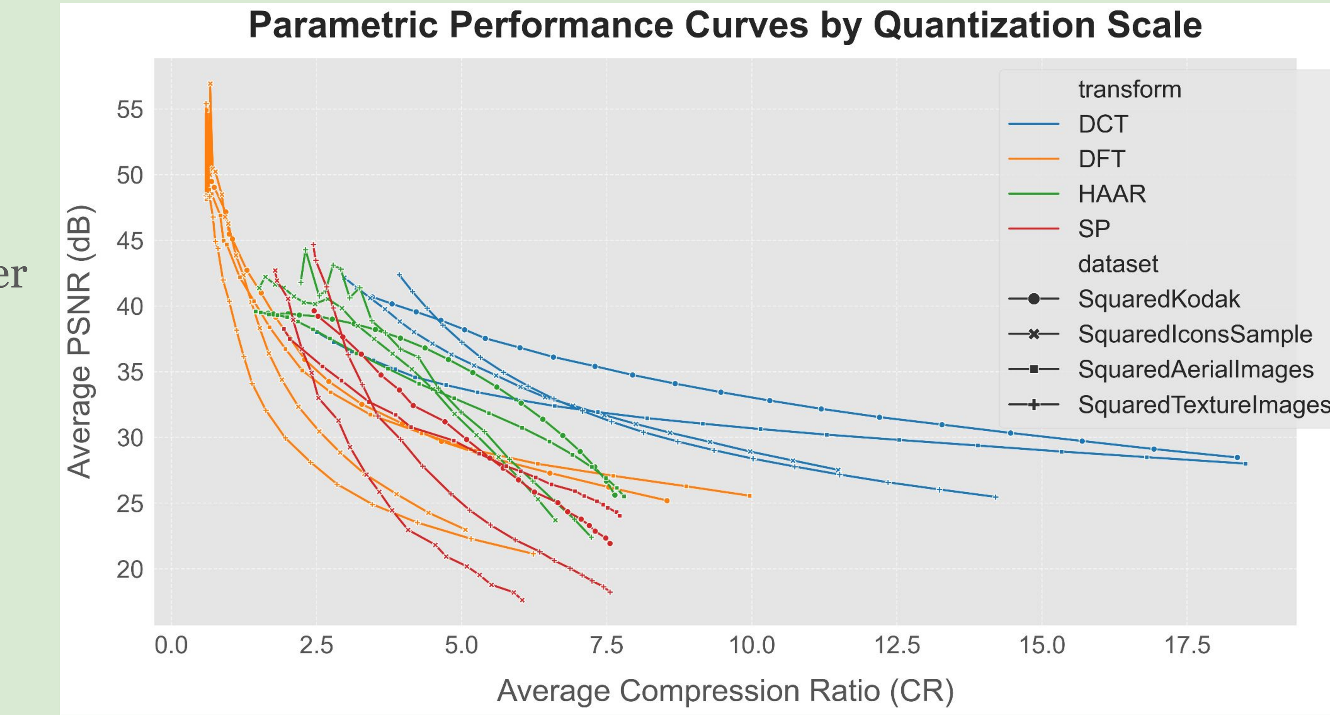

Fig. 4.1: Tradeoffs in smaller filesize (CR) vs. image distortion (PSNR).

The extensive development of the JPEG Standard is clear; quantization and entropy encoding methods for DCT are less effective on other transforms even when adapted. Haar and S+P Transforms lose on compression when prediction is applied, and no other transforms are able to match CRs achieved by DCT.