

# ECEN 4532 - Lab 5: JPEG Image Processing

Andrew Teta

April 15, 2019



# Contents

<b>Introduction</b>	<b>3</b>
<b>1 Introduction</b>	<b>3</b>
<b>2 Background</b>	<b>3</b>
2.1 Why Compress Images? . . . . .	3
2.2 Compression . . . . .	3
2.3 Encoder . . . . .	3
2.4 Decoder . . . . .	4
2.5 Important Compression Parameters . . . . .	4
<b>3 Transform Coding</b>	<b>4</b>
3.1 Discrete Cosine Based Coders . . . . .	5
<b>4 JPEG Compression Algorithm</b>	<b>5</b>
4.1 Forward and Inverse DCT . . . . .	5

# 1 Introduction

JPEG (Joint Photographic Experts Group) is a common compression scheme and file type used for image data compression and storage. JPEG is very common as it is efficient and easily implemented. This lab is an exercise in implementing the framework of the JPEG algorithm.

## 2 Background

### 2.1 Why Compress Images?

Image data is collected from many different sources in many different resolutions. Some examples would be camera pictures, medical (X-Ray, MRI, etc.), and video. Some of these are originally captured in analog format on tape or film media and some are inherently digital, such as a digital camera or video recorder. There are many reasons why a digital media format may be advantageous. For example:

- Storage and transmission
- Further processing (color correction, enhancement, crop, machine vision, etc)
- Database storage

Storing images and videos in digital form at full resolution, especially for analog formats, introduces large hurdles for storage capacity and efficiency. Full resolution can also be called uncompressed and uncompressed data can make it more difficult to store, transmit, and search the data. Thus, there is motivation for some way to encode a lot of data in a small number of bits, neglect unnecessary information, or combine these both into a single 'compression scheme'.

### 2.2 Compression

At a high level, compression can be decomposed into the three components of (1) an encoder, (2) a decoder, and (3) a channel. A channel can be considered the media transmission platform. Examples include: wireless, satellite, ethernet, CD, optical fiber, computer memory, etc.

There is potential for transmission errors to occur in the channel, such as bit corruption over a wireless communication or something similar, however we will consider for this lab, only a perfect channel.

### 2.3 Encoder

JPEG is a transform-based encoder so we will focus on that, although there are others. A transform encoder can be decomposed into

1. Pre-processing. These would be things that occur before the actual compression algorithm with the purpose of preparing the data to minimize coding complexity or increase compression efficiency. Some examples would include:
  - filtering
  - zero-padding
  - symmetric extension on the boundaries
  - tiling
2. Transform. The transform is intended to reduce the correlation among image pixels and describe most of the information in an image with a small set of significant coefficients. Then, most of the energy is condensed into a few coefficients.

3. Quantization. The transform output is a set of real-valued coefficients. The purpose of quantization is to represent these values with a much smaller set of values, often more easily represented within the encoding scheme. Quantization is non-reversible and distortion inducing. Image quality is balanced against the number of bits necessary to represent the data here.
4. Entropy coding. This is a creatively efficient way to reduce bit quantity. More common symbols (or information) is represented with shorter code-words and vice-versa.

## 2.4 Decoder

The decoder reconstructs the image by inverting the encoder's Entropy coding, Quantization, and Transformation. Often, a final post-processing step is performed on the reconstructed image to conceal coding artifacts and improve the visual appearance.

## 2.5 Important Compression Parameters

Some fundamental parameters include:

- Compression efficiency. This measures the compression ratio (often in bits per pixel (bpp))
- Fidelity. This is algorithmic distortion and can often be measured using the PSNR (Peak Signal to Noise Ratio). This is only quantitative, however, and image data often still requires visual inspection to evaluate the compression method.
- Complexity. In situations where real-time compression is required, the algorithm complexity is a key parameter.
- Memory. Sometimes memory can be a limiting factor and the compression method needs to work with that.
- Robustness. In certain applications such as wireless communication, channel coding errors can be catastrophic to reconstruction and therefore the compression algorithm must be robust.

## 3 Transform Coding

Transform coding is a linear operation which converts data from image space to some transformed space. Often the transformed space is in the frequency domain but this is a little more confusing in spatial data such as images. The transformed values are referred to as coefficients and when the proper transform is applied, the coefficients will be less correlated than the original samples. The idea is that, similar to 1D transform coding, a majority of the data will fall into a smaller range and the rest can be considered unnecessary and be quantized to zero. Most compression transforms operate in the frequency domain (i.e. Fourier Transform). Furthermore, many fast algorithms exist for such a transform (often based on the FFT) and in many cases for images, a frequency domain transform can be considered nearly as optimal as the theoretically best transform.

In practice, many aspects of the human visual system are well understood in the frequency domain, so working in this domain is easier than an optimal domain. In addition, known facts about human visual system sensitivity to different spatial frequencies can be incorporated.

The theoretically best transform is known as the Karhunen-Loève transform (KLT), and is a function of image statistics. Essentially, the KLT is the best case scenario for condensing the most energy into the fewest number of transform coefficients. The KLT depends on knowledge of the correlation matrix of the input and so is rarely used in practice. However, it does provide justification for the performance of transform coding. For certain inputs, the Discrete Cosine Transform (DCT) provides an approximation to the KLT. One more alternative to the KLT is a wavelet-based image compression algorithm which researchers have worked on. This is an orthonormal transform that provides very efficient decorrelation of images and can under certain circumstances provide an approximation to the KLT.

### 3.1 Discrete Cosine Based Coders

The DCT can be considered a good choice of transform. For highly correlated data, the DCT provides excellent energy compaction and can be computed using an FFT. Unlike an FFT, however, the DCT coefficients are real. The DCT is now used for the JPEG compression standard.

## 4 JPEG Compression Algorithm

There are three main steps to the JPEG algorithm. These steps operate in a forward and reverse direction, corresponding to an encoding and decoding. In the forward direction, the algorithm implements a forward DCT (FDCT) and quantizes the data. This is followed by entropy encoding. The decoding process implements the inverse of each step.

### 4.1 Forward and Inverse DCT

An image is divided into non-overlapping blocks of size 8 x 8 and scanned left to right and top to bottom. Each block is transformed using a 2-dimensional DCT, which is given by:

$$F_{u,v} = \frac{1}{4} C_u C_v \sum_{x=0}^7 \sum_{y=0}^7 f(x,y) \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16}, \quad u, v = 0, \dots, 7 \quad (1)$$

where  $C_v$  follows the same rule as  $C_u$  and

$$C_u = \begin{cases} 1 & \text{if } u \neq 0, \\ \frac{1}{\sqrt{2}} & \text{if } u = 0. \end{cases}$$

The coefficients  $F_{u,v}$  are coarsely approximated over a small set of possible values (quantization) and replaced by  $\tilde{F}_{u,v}$ . The quantization process introduces distortion and contributes to image quality loss.

With  $\tilde{F}_{u,v}$  as the input (decoding), the inverse DCT can be applied to reconstruct an estimate of the block by:

$$\tilde{f}(x,y) = \frac{1}{4} \sum_{u=0}^7 \sum_{v=0}^7 C_u C_v \tilde{F}_{u,v} \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16}, \quad x, y = 0, \dots, 7 \quad (2)$$

where  $C_v$  follows the same rule as  $C_u$  and

$$C_u = \begin{cases} 1 & \text{if } u \neq 0, \\ \frac{1}{\sqrt{2}} & \text{if } u = 0. \end{cases}$$

Once the DCT coefficients are calculated (in the forward encoding direction), the block is ordered in a zigzag pattern to improve spatial correlation and avoid edge cases.

### 4.2 Quantization and Inverse Quantization