Double/Multiple JPEG Compression Detection

Group Number-15

Hitesh Saini(14AE30008)
Kunal Jain(14AE30025)
Upamanyu Chongdar(14EC35032)

1. Introduction

Digital images have become a very important information carrier in our daily lives. However, due to the ease of use and availability of image editing tools, it brings significant concern over the integrity of digital images, especially in the field of digital forensics. Therefore, it is very important that forensics investigators possess the appropriate technology to accurately and reliably verify the integrity of digital image evidence. In this paper, we focus on the research of Joint Photographic Experts Group (JPEG) image forensic to detect the evidentiary presence of digital image forgery and tampering. JPEG is chosen due to its widespread popularity as the de facto image format used by most electronic devices. Double compression based image tampering detection is a common form of forged/tampered JPEG image detection method. The term “double compression” refers to the decompression of an already compressed JPEG image and the application of a subsequent recompression action to the image contents to store/save it back into the JPEG format. This process is a natural and mandatory part of the procedure that is carried out when JPEG images are tampered with. For example, to forge an existing JPEG image, the image needs to be decompressed, the manipulation methods such as splicing can then be applied on to the image, and the modified image is then recompressed back into a JPEG image. Such a JPEG double compression action has been observed to cause the presence of the periodic effects on the recompressed DCT coefficients. Thus, the existence of these coefficient effects for a given JPEG image can be used to verify if the image has undergone tampering.

2.Basics:JPEG Compression

Although a JPEG file can be encoded in various ways, most commonly it is done with JFIF encoding. The encoding process consists of several steps:

1. The representation of the colors in the image is converted from RGB to $Y'C_BC_R$, consisting of one luma component (Y'), representing brightness, and two chromacomponents, (C_B and C_R), representing color. This step is sometimes skipped.
2. The resolution of the chroma data is reduced, usually by a factor of 2 or 3. This reflects the fact that the eye is less sensitive to fine color details than to fine brightness details.

3. The image is split into blocks of 8×8 pixels, and for each block, each of the Y, C_B, and C_R data undergoes the discrete cosine transform (DCT). A DCT is similar to a Fourier transform in the sense that it produces a kind of spatial frequency spectrum.
4. The amplitudes of the frequency components are quantized. Human vision is much more sensitive to small variations in color or brightness over large areas than to the strength of high-frequency brightness variations. Therefore, the magnitudes of the high-frequency components are stored with a lower accuracy than the low-frequency components. The quality setting of the encoder (for example 50 or 95 on a scale of 0–100 in the Independent JPEG Group's library[20]) affects to what extent the resolution of each frequency component is reduced. If an excessively low quality setting is used, the high-frequency components are discarded altogether.
5. The resulting data for all 8×8 blocks is further compressed with a lossless algorithm, a variant of Huffman encoding.

The decoding process reverses these steps, except the *quantization* because it is irreversible. In the remainder of this section, the encoding and decoding processes are described in more detail.

3(a).DCT

Each block of each of the Y, Cb, and Cr components undergoes a discrete cosine transform (DCT). Let f(x, y) denotes a 8 × 8 block. Its DCT is:

$$F(u,v) = \frac{1}{4}C(u)C(v) \sum_{x=0}^{7}\sum_{y=0}^{7} f(x,y) cos\frac{(2x+1)u\pi}{16} cos\frac{(2y+1)v\pi}{16},$$

where

$$(u, v \in \{0 \cdots 7\});$$
$$C(u), C(v) = 1/\sqrt{2} \quad \text{for} \quad u, v = 0;$$
$$C(u), C(v) = 1 \quad \text{otherwise}.$$

The DCT temporarily increases the bit-depth of the data, since the DCT coefficients of an 8-bit/component image take up to 11 or more bits (depending on fidelity of the DCT calculation) to store. This may force the codec to temporarily use 16-bit numbers to hold these coefficients, doubling the size of the image representation at this point; these values are typically reduced back to 8-bit values by the quantization step. The temporary increase in size at this stage is not a performance concern for most JPEG implementations, since typically only a very small part of the image is stored in full DCT form at any given time during the image encoding or decoding process.

3(b).Quantization

The human eye is good at seeing small differences in brightness over a relatively large area, but not so good at distinguishing the exact strength of a high frequency brightness variation. This allows one to greatly reduce the amount of information in the high frequency components. This is done by simply dividing each component in the frequency domain by a constant for that component, and then rounding to the nearest integer. This rounding operation is the only lossy operation in the whole process (other than chroma subsampling) if the DCT computation is performed with sufficiently high precision. As a result of this, it is typically the case that many of the higher frequency components are rounded to zero, and many of the rest become small positive or negative numbers, which take many fewer bits to represent.

The elements in the quantization matrix control the compression ratio, with larger values producing greater compression. A typical quantization matrix (for a quality of 50% as specified in the original JPEG Standard), is as follows:

$$Q = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}.$$

The quantized DCT coefficients are computed with

$$B_{j,k} = \mathrm{round}\left(\frac{G_{j,k}}{Q_{j,k}}\right) \text{ for } j = 0, 1, 2, \ldots, 7; k = 0, 1, 2, \ldots, 7$$

where G is the unquantized DCT coefficients; Q is the quantization matrix above; and B is the quantized DCT coefficients.

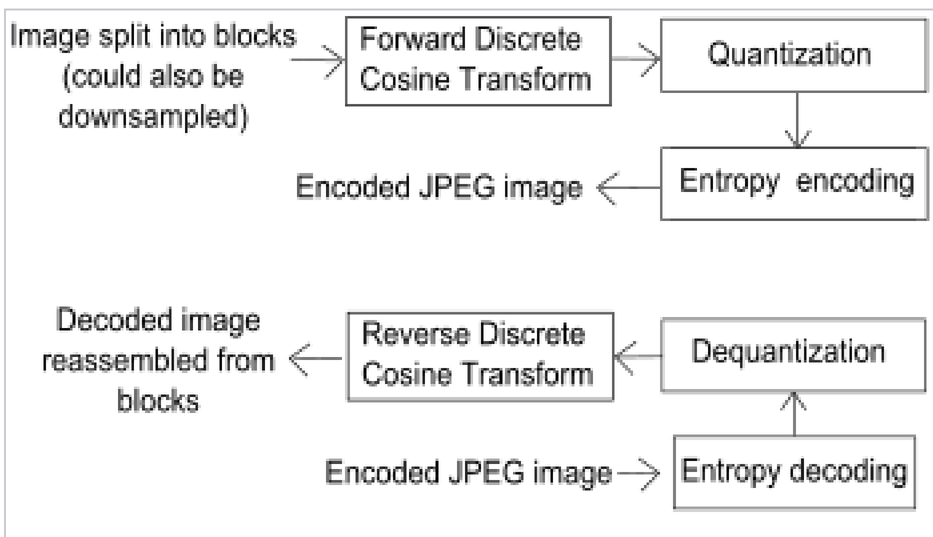
3(c)Entropy coding

Entropy coding is a special form of lossless data compression. It involves arranging the image components in a "zigzag" order employing run-length encoding (RLE) algorithm that groups similar frequencies together, inserting length coding zeros, and then using Huffman coding on what is left.

The JPEG standard also allows, but does not require, decoders to support the use of arithmetic coding, which is mathematically superior to Huffman coding. However, this feature has rarely been used, as it was historically covered by patents requiring royalty-bearing licenses, and because it is slower to encode and decode compared to Huffman coding. Arithmetic coding typically makes files about 5–7% smaller.

The previous quantized DC coefficient is used to predict the current quantized DC coefficient. The difference between the two is encoded rather than the actual value. The encoding of the 63 quantized AC coefficients does not use such prediction differencing.

Two types of encoding are sequential encoding and baseline encoding.One such example is shown below.



***Baseline sequential* JPEG encoding and decoding processes**

This encoding mode is called baseline *sequential* encoding. Baseline JPEG also supports *progressive* encoding. While sequential encoding encodes coefficients of a single block at a time (in a zigzag manner), progressive encoding encodes similar-positioned batch of coefficients of all blocks in one go (called a *scan*), followed by the next batch of coefficients of all blocks, and so on. For example, if the image is divided into N 8×8 blocks

3(d)Double Quantization

By double JPEG compression we understand the repeated compression of the image with different quantization matrices $Q\alpha$ (primary quantization matrix) and $Q\beta$ (secondary quantization matrix). The DCT coefficient F(u, v) is said to be double quantized if $Q\alpha(u, v)$ is not equal to $Q\beta(u, v)$.

Apparently, the distribution of the doubly quantized values contains periodic empty bins. This is caused by the fact that in the second quantization values of the distribution are re–distributed into more bins than in the first quantization. Generally, the double quantization process brings detectable artifacts like periodic zeros and double peaks.

4.Detection

The easily available UCID dataset comprising of uncompressed raw images in tif format was used for the project. OpenCV was used to generate singly and doubly compressed jpg images with controlled quantization values.

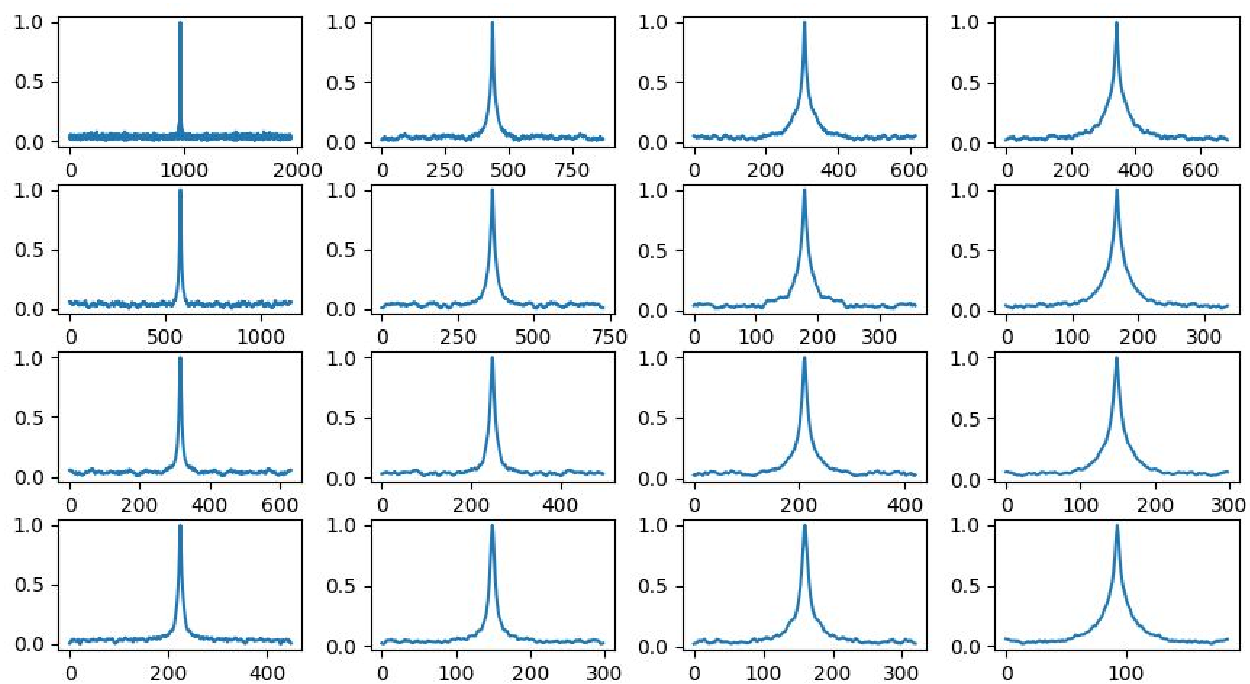
For the detection of the traces of double quantization in JPEG images , we use the fact that the histograms of DCT coefficients of a double compressed image contain specific periodic artifacts detectable in the frequency space. We have computed the zero–mean histograms of DCT coefficients corresponding to low frequencies. For each of these histograms the magnitudes of their Fourier transforms are obtained. If DCT coefficients corresponding to DCT frequency (u, v) are double quantized, the corresponding histogram and Fourier transform has a specific behavior.

Specifically, if the image is double compressed, typically the output of the method applied to the DC component contains a specific clear peak . Otherwise, there is no strong peak in the spectrum When the method is applied to a single-quantized AC component, the spectrum has a decaying trend Otherwise, in some parts, the spectrum has a local ascending trend

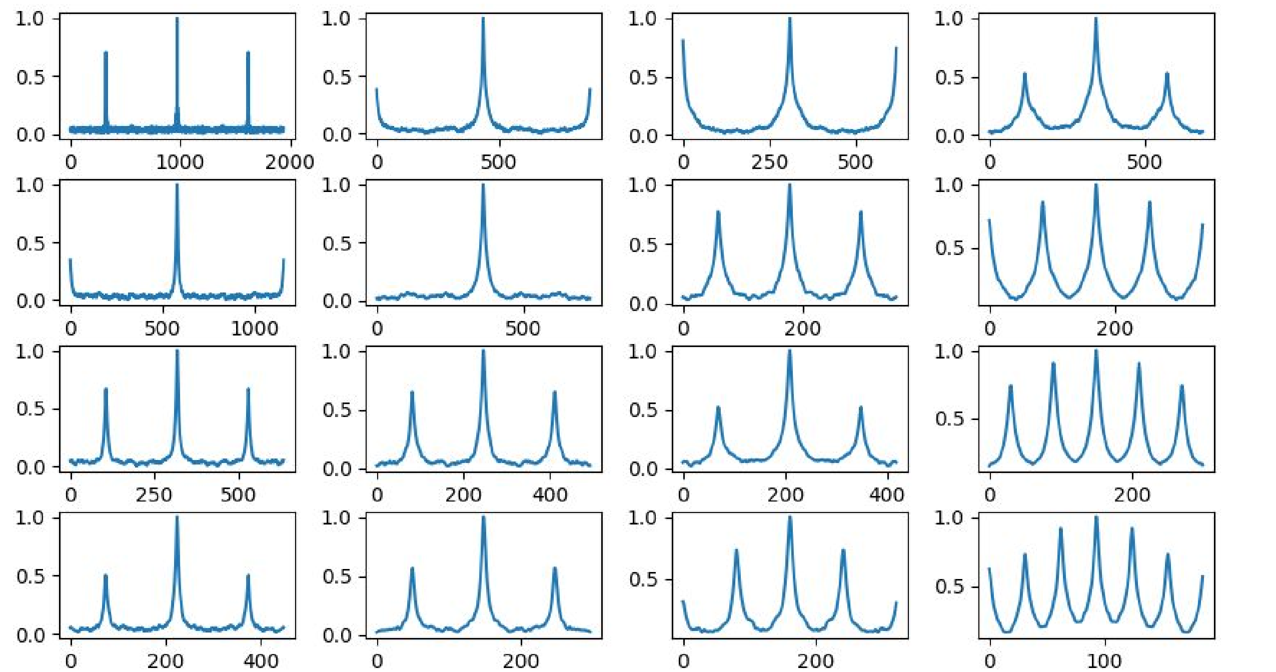
Our method computes the magnitudes of FFT of the histograms of the DCT coefficients corresponding low frequencies. Specifically, the following DCT frequencies are employed: (0, 0), (1, 0), (2, 0), (3, 0), (0, 1), (1, 1), (2, 1), (0, 2), (1, 2) and (0, 3). Because of the problem with insuffi- cient statistics for high–frequency DCT coefficients (high frequency DCT coefficients are often quantized to zeros), other frequencies are not considered. Only the first half of the spectrum is considered. We denote the result of this part by |H1| · · · |H10|, where |H1| corresponds to DC component and |Hi |, i = 2 · · · 10, correspond to AC components. Furthermore, |Hi | are normalized to have a unit length.
Double compression is detected using thresholds determined experimentally in the spreadsheet attached.

5.Plots

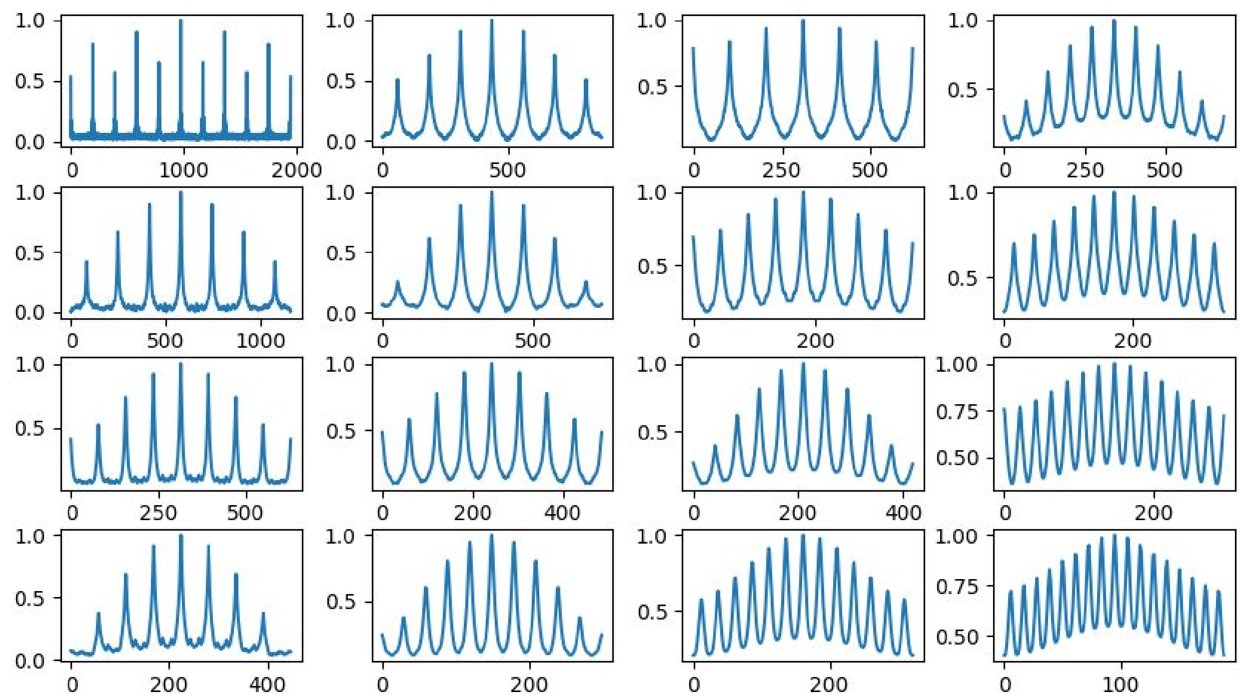
Uncompressed:-



Single compressed (Quality factor = 90):-



Double compressed (Quality factor = 70):-



Environment-OpenCV 3.3 Platform-Linux