Estimation of Primary Quantization Matrix in Double Compressed JPEG Images

Jan Lukáš and Jessica Fridrich
Department of Electrical and Computer Engineering
SUNY Binghamton, Binghamton, NY 13902-6000
{jan.lukas, fridrich}@binghamton.edu

Abstract

In this report, we present a method for estimation of primary quantization matrix from a double compressed JPEG image. We first identify characteristic features that occur in DCT histograms of individual coefficients due to double compression. Then, we present 3 different approaches that estimate the original quantization matrix from double compressed images. Finally, most successful of them - Neural Network classifier is discussed and its performance and reliability is evaluated in a series of experiments on various databases of double compressed images. It is also explained in this paper, how double compression detection techniques and primary quantization matrix estimators can be used in steganalysis of JPEG files and in digital forensic analysis for detection of digital forgeries.

1. Introduction

A double compressed JPEG file is created when a JPEG image is decompressed and then resaved with a different quantization matrix. There are at least two reasons why forensic experts should be interested in double compressed images and the estimation of the primary (first) quantization table. First, double compressed JPEG images often result from digital manipulation (forgeries) when a portion of the manipulated image is replaced with another portion from another image and resaved. In this case, the pasted portion will likely exhibit traces of only a single compression while the rest of the image will exhibit signs of double compression. This observation could in principle be used to identify manipulated areas in digital images. Second, double compressed images are often produced by steganographic programs (e.g., by F5 [5], OutGuess [4], or J-Steg). For some steganalytic methods [4],[5], it is very important to estimate the primary quantization matrix to facilitate accurate and reliable steganalysis.

Previously, successful attempts to determine whether a bitmap image was originally JPEG compressed have been made in [2] and [3]. A simple idea how to detect JPEG double compression and estimate the primary quality factor was suggested and partially realized in [4] and [5]. On this idea was built our first detection approach presented in section 4.1.

Due to simplicity of explanation, we focus in this paper only on double compression of grayscale images. We believe that the technique presented here can be extended to work with both chrominance and luminance components of color JPEG images.

In Section 2 we briefly review the basics of JPEG compression and discuss JPEG double compression. Then, we describe some characteristic features in the DCT histogram that are created during double compression. In Section 3, we discuss the problem of estimating the primary quantization matrix and mention some inherent limitations In

Section 4 we present 3 different approaches to the primary matrix estimation, which we have examined. Experiments are described and the results of our neural network classifier are presented in Section 5. Section 6 contains conclusions and future ideas.

2. The double compression phenomenon

2.1 JPEG compression and decompression

In JPEG compression, the image is first divided into disjoint 8×8 pixel blocks. For each block B, the discrete cosine transform (DCT) is calculated using the following formula:

$$D_{ij} = \sum_{k,l=0}^{7} a_{kl}(i,j) B_{kl} \, ,$$

where $a_{kl}(i,j) = \frac{1}{4} w(k) w(l) \cos\frac{\pi}{16} k(2i+1) \cos\frac{\pi}{16} k(2j+1)$ and $w(k) = 1/\sqrt{2}$ for $k = 0$ and $w(k) = 1$ otherwise.

The DCT coefficients (matrix D) are then quantized using a quantization matrix Q:

$$D_{ij}^{q} = \mathrm{round}\left(\frac{D_{ij}}{Q_{ij}}\right), i, j \in \{0, \ldots, 7\}.$$

The quantized coefficients D^{q}_{ij} are then arranged in a zigzag order and compressed using the Huffman encoder. The resulting compressed stream together with a header forms the final JPEG file.

The decompression works in the opposite way. For each block B, the quantized DCT coefficients D^{q} obtained from the JPEG file are multiplied by quantization coefficients stored in the quantization matrix Q: $D_{ij} = Q_{ij} D^{q}_{ij}, \, i, j \in \{0, \ldots, 7\}$. We note that the matrix Q is stored in the JPEG file. Then, the inverse DCT (IDCT) is computed, and finally the result is rounded and truncated to integer values in the interval [0, 255]: B = truncate[round(IDCT(D))].

2.2 JPEG double compression

By double compression we understand repeated JPEG compression of the image with different quantization matrices Q^{1} (primary matrix) and Q^{2} (secondary matrix). The DCT coefficient D_{ij} is said to be double compressed if and only if $Q_{ij}^{1} \neq Q_{ij}^{2}$.

Next, we look at what exactly happens with DCT coefficients D_{ij} (ij-th coefficient from each 8×8 block in the image) during double compression. In the original JPEG image, these coefficients are quantized with Q_{ij}^{1}, which means that in every 8×8 block in the image, the value of the DCT coefficient D_{ij} is a multiple of Q_{ij}^{1}. When the image is decompressed, pixels are integer rounded and truncated to values from the interval [0, 255]. When processed for the second time, the DCT coefficients are computed from these rounded and truncated values. This is the reason why coefficients usually lose their integer values. They are no longer multiples of Q_{ij}^{1}, but they are spread around these multiples. Then, they are quantized with the matrix Q_{ij}^2 and the JPEG file is formed the same way as described above.

The concentration of coefficients around multiples of Q_{ij}^1 and their following quantization by Q_{ij}^2 creates a pattern in the histogram of the values D_{ij} that can be used for identification of the primary quantization factor Q_{ij}^1. For instance, when $Q_{ij}^1 > Q_{ij}^2$, some multiples of Q_{ij}^2 are almost missing in the histogram. In the opposite case, some multiples of Q_{ij}^2 form local maximums or minimums in the histogram of D_{ij}. These missing points, as well as the extremes, can be identified to determine Q_{ij}^1. Experiments showed that especially the missing values ("almost zeros" in the DCT histogram) are a very robust feature providing important information about the primary quantization factors.

Another very interesting property is the double peak. Some ideal cases of double peaks are marked in Figures 1b, 1c, and 1d. A double peak occurs when some integer multiple of Q_{ij}^1 falls between two multiples of Q_{ij}^2 and none of the remaining multiples of Q_{ij}^2 lies closer to Q_{ij}^1. Mathematically, there exist integers k, l, such that $kQ_{ij}^1=[(l-1)Q_{ij}^2+lQ_{ij}^2]/2$. This implies that a double peak can occur only for even multiples of Q_{ij}^2, because k and Q_{ij}^1 are integers and thus $[(l-1)Q_{ij}^2+lQ_{ij}^2]$ must be divisible by 2. Next, we look at what happens during double compression for such a combination of integers k, l.

As stated above, after decompressing a JPEG file and computing DCT, the DCT coefficients are spread around multiples of Q_{ij}^1. It ca be shown that this distribution can be modeled as Gaussian with the mean value at the multiples of Q_{ij}^1. When the value kQ_{ij}^1 is quantized by Q_{ij}^2, most of the coefficients with value less than kQ_{ij}^1 are indeed rounded to $(l-1)Q_{ij}^2$ and the coefficients with values greater or equal to kQ_{ij}^1 are rounded to lQ_{ij}^2. The values $l-1$ and l will exhibit a double peak in the histogram of the coefficient D_{ij} (see Figure 1b-d).

The shape of double peaks also depends on the implementation of the DCT transform. Due to computational efficiency, some DCT implementations do not compute DCT coefficients with a high accuracy and return coefficients that are quantized by some factor (for instance by 1/8). This causes asymmetry in rounding and more DCT coefficients are quantized to the right of kQ_{ij}^1 than to the left. Thus, more DCT coefficients are quantized to lQ_{ij}^2 than to $(l-1)Q_{ij}^2$.

From our experiments, we conclude that although statistical properties for double peaks exist, the shape of individual instances of double peaks exhibit great variability that is hard to predict. Overall, the presence of double peaks makes the detection of JPEG double compression harder rather than easier.

In order to understand the behavior of histograms of individual DCT coefficients in a double compressed image, we averaged the histograms of the coefficients D_{10} and D_{11} obtained from 100 JPEG files. The multiples of Q_{ij}^2 equal to 0 and 1 were not included in these histograms, because the maximum at zero is not a distinguishing feature and the value at 1 is usually still too big and thus it would have an undesirable effect after histogram normalization. In Sections 2.2.1–2.2.3 we analyze three different combinations of the primary and secondary quantization steps and how they affect the shape of the DCT histogram of individual DCT coefficients in the double compressed image.

2.2.1 $Q_{ij}^1=Q_{ij}^2$ or Q_{ij}^1 is a divisor of Q_{ij}^2

When Q_{ij}^1 is equal to Q_{ij}^2 or when Q_{ij}^1 is a divisor of Q_{ij}^2, we do not expect any special features in the histogram of the coefficient D_{ij}. The histogram should be smooth as shown in Figure 1a (the histogram is normalized).

If a double peak is present, then each multiple of Q_{ij}^1 participates in two double peaks, for instance when $Q_{ij}^1=4$ and $Q_{ij}^2=8$, then odd multiples of 4 split into double peaks, but each multiple of 8 lies between two odd multiples of 4. Thus, one double peak from left and one from right is contributing to each multiple of 8 and the resulting effect is that double peaks should be and usually are unobservable.

2.2.2 $Q_{ij}^1>Q_{ij}^2$

As mentioned in Section 1, when Q_{ij}^1 is greater than Q_{ij}^2, some multiples of Q_{ij}^2 in the histogram of D_{ij} may be missing in the double compressed image. In general, each multiple of Q_{ij}^2 either contains all coefficients that were quantized to one specific multiple of Q_{ij}^1 or it participates in a double peak or it has been almost "zeroed" in the histogram. The first category is always present, while one of the remaining categories may be absent from the histogram of D_{ij}.

Figure 1b shows the ideal case of a normalized histogram where the quantization with $Q_{ij}^1=7$ was followed by quantization with $Q_{ij}^2=2$. We can see double peaks at multiples 3 and 4 (because 7 is rounded either to 6 or 8) and at 10 and 11 (21 is rounded either to 20 or 22.). There are several "almost zeros", because no multiple of 7 is quantized to the following multiples of 2 {2, 5, 6, 8, 9, 12, 13, 15}.

Another example is shown in Figure 1c. The quantization with $Q_{ij}^1=6$ was followed by quantization with $Q_{ij}^2=4$. All possible zeros are affected by double peaks and, as a result, there are no zeros in the histogram. The coefficient value 12 was requantized to 12 (3 in the histogram), 18 is split into a double peak at multiples 4 and 5, 24 was requantized to 24, then a double peak follows, and so on. Because the double peaks are always located between two local maximums, we notice them only as local minimums.

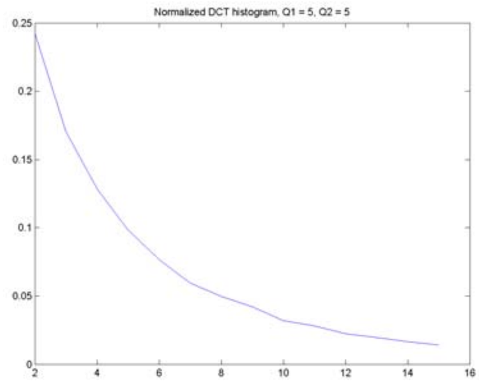

Figure 1a: Normalized DCT histogram, Q_{ij}^1=5, Q_{ij}^2=5 (ideal case).

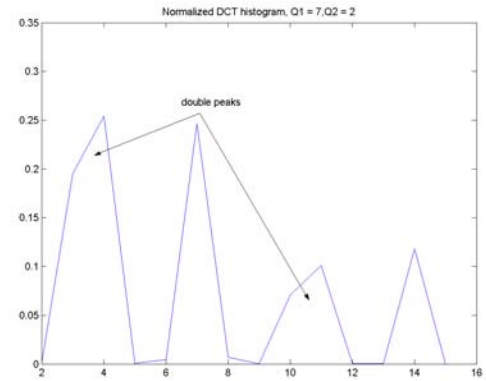

Figure 1b: Normalized DCT histogram, Q_{ij}^1=7, Q_{ij}^2=2 (ideal case).

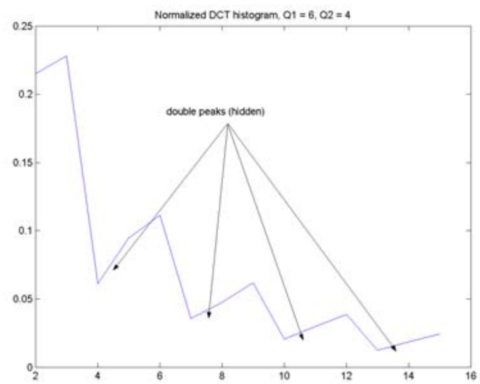

Figure 1c: Normalized DCT histogram, Q_{ij}^1=6, Q_{ij}^2=4 (ideal case).

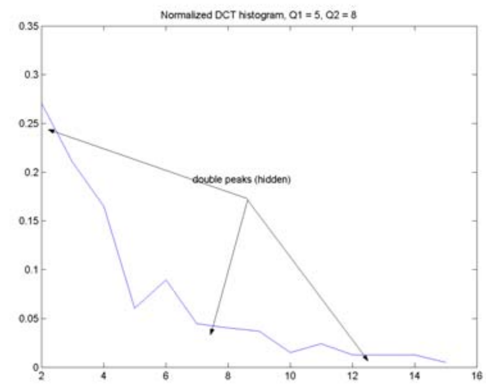

Figure 1d: Normalized DCT histogram, Q_{ij}^1=5, Q_{ij}^2=8 (ideal case).

2.2.3 $Q_{ij}^1<Q_{ij}^2$ but Q_{ij}^1 is not a divisor of Q_{ij}^2

If Q_{ij}^1 is smaller than Q_{ij}^2, all multiples of Q_{ij}^2 are always present in the histogram, however different number of multiples of Q_{ij}^1 are quantized to different multiples of Q_{ij}^2. Also, if some multiple *m* of Q_{ij}^2 is participating in a double peak, there exists at least one more multiple of Q_{ij}^1, that was quantized to this multiple *m*. Below, we show on examples how this mechanism changes the shape of the histogram of D_{ij} in the double compressed image.

Let's look at the example shown in Figure 1d. The quantization coefficient Q_{ij}^1=5 was followed by quantization coefficient Q_{ij}^2=8. The value 16 (multiple 2) is the quantized value 15 plus it participates in a double peak from 20. Similarly, the value 24 (multiple 3) is the quantized value 25 and it also participates in a double peak from 20. The value 32 (multiple 4) is the quantized value of both 30 and 35. Therefore, it should exhibit a local maximum, but because it is still located in the part of histogram that has a steep slope, we notice just a slight change in the slope. The value 40 (multiple 5) is just one requantized value 40 and it is a local minimum. The value 48 (multiple 6) is the quantized value of both 45 and 50 and forms a local maximum. A double peak follows, and so on. We can see that the double peaks at multiples 7 and 8 and at 12 and 13 are almost imperceptible.

3. Problem Formulation, Strategy, and Limitations

The two tasks investigated in this paper are double compression detection and estimation of the primary quantization matrix. In other words, given a JPEG file with the quantization matrix Q^2, we need to determine if the file was previously JPEG compressed and quantized with a different quantization matrix Q^1 and estimate its values.

First, we would like to point out that methods described in this paper focus on double compressed images and may not work properly for files that underwent JPEG compression repeated more than two times (e.g., triple compression). In those cases, the individual DCT histograms will have other, perhaps more complicated characteristics than in the case of double compression.

The problem of reliable double compression detection would be much easier if we could limit ourselves to standard JPEG quantization matrices or matrices that are close to standard ones. However, custom quantization matrices are widely used, especially in digital cameras. Such matrices cannot be matched to any standard quantization matrix. For instance, the following matrix is a commonly used luminance quantization matrix by the Kodak DC 290 camera.

5	5	5	5	5	6	6	8
5	5	5	5	5	6	7	8
5	5	5	5	6	7	8	9
5	5	5	6	7	8	9	10
5	5	6	7	8	9	11	12
6	6	7	8	9	11	13	14
6	7	8	9	11	13	15	16
8	8	9	10	12	14	16	19

The only solution to this problem is to estimate individual quantization steps rather than the whole matrix. The DCT transform is an orthogonal transform, so changes (e.g., quantization) in one coefficient should not affect other coefficients. This property allows estimation of the quantization step Q_{ij}^1 for each DCT coefficient D_{ij} separately. However, we may not be able to estimate all quantization steps, especially those corresponding to high frequencies. This is because digital images of natural scenes have most of their power in low frequencies and thus higher frequency coefficients are often quantized to zero and are more affected by noise. As a result, we can expect to have insufficient statistics in a typical image to correctly determine quantization steps corresponding to higher frequencies (larger $i+j$). Because of these reasons, in this paper, we constrain our analysis to the low-frequency coefficients only. Fortunately, such coefficients are usually the most important for forensic analysis, be it steganalysis of forgery identification.

Another fundamental limitation occurs for certain values of Q_{ij}^1 and Q_{ij}^2 or their relationship. Obviously, we will not be able to distinguish between cases where $Q_{ij}^1=1$ (or there was no previous JPEG compression.) and $Q_{ij}^1= Q_{ij}^2$. In both cases, the DCT coefficients D_{ij} (from all 8×8 blocks in the image) will be multiples of Q_{ij}^2 without exhibiting any characteristic features, such as minimums, maximums, double-peaks, etc. For the same reason, another indistinguishable case is when Q_{ij}^1 is a divisor of Q_{ij}^2.

There is one more important property of DCT coefficients that we quote in this paper – the shape of their histogram. The histograms of all AC coefficients (D_{ij}, with $i+j\neq 0$) have the Laplacian distribution [2].

Although all methods described bellow are designed specifically for AC terms, we strongly believe they could be easily modified to work for the DC term as well. The DC term also exhibits local maximums, minimums, and double peaks that could be the main distinguishing features for our analysis.

4. Methods for Estimation of Primary Quantization Steps

4.1 Compatibility Test

This method was briefly outlined in our previous papers on steganalysis [4] and [5]. As the first step, we calculate the histograms of absolute values of all DCT coefficients of our interest from the image. Let us denote one such histogram as h_o. The image is then cropped (for example by 4 pixels) to disrupt the structure of JPEG blocks. Then, it is JPEG compressed with the set of candidate quantization matrices $Q^{1,1}$, …, $Q^{1,n}$. All these n JPEG files are then decompressed and JPEG compressed again with the quantization matrix Q^2. From all n double compressed JPEG files we obtain histograms $h(Q^{1,1})$, …, $h(Q^{1,n})$. The original quantization matrix can then be estimated as $Q^1 = \arg\min_{Q,1,m} ||h(Q^{1,m})-h_o||$. As a norm, we simply used the sum of absolute values (the L^1 norm).

Although this method is relatively robust and reliable, it requires a limited set of n possible quantization matrices. When dealing with custom quantization matrices, it is necessary to compute minimums for each DCT coefficient of interest D_{ij} separately. The corresponding quantization step for every coefficient D_{ij} is then Q_{ij}^1. From these quantization steps we form a part of the primary quantization matrix Q^1 and, if needed, we can eventually determine the remaining terms in the matrix by finding the closest standard matrix.

For this method, the length of the histogram is an important parameter. Because the histogram of absolute values of DCT coefficients is rapidly decreasing, larger values of DCT coefficients produce insufficient statistics for any method to work. In our experiments, we have tried histogram lengths 10 (the maximal absolute value of DCT coefficient contained in the histogram is $10Q_{ij}^2$), 15 (the maximal value is $15Q_{ij}^2$) and 30 (the maximal value is $30Q_{ij}^2$). According to our experiments, the length 15 has been most useful.

4.1.1 Implementation and Improvement Issues

Because our task is to estimate also custom JPEG quantization matrices, we have always tried to detect double compression quantization steps for each DCT coefficient separately. The simpler version of this method described in [4] and [5] dealt with entire matrices and was used only in the beginning phase of our research.

We should note that although we are dealing with individual DCT coefficients, we are not able to perform DCT decompression, rounding, and truncation in the image domain for a single DCT coefficient. For a given DCT coefficient D_{ij} (and a given quantization matrix Q^2), we must therefore select some candidate matrices $Q^{1,1}$, …, $Q^{1,n}$, such that $Q_{ij}^{1,m}$ are candidate values of quantization steps used for the DCT coefficient D_{ij}, when the image was originally compressed.

In our experiments, we have originally used standard quantization matrices for this purpose, but later we switched to a special kind of quantization matrices – constant quantization matrices. A constant quantization matrix is defined as $Q_{ij}^{1,m} = C_m$, for all i, j, where C_m is the candidate quantization step. This enables us to speed up computations, because we have the same amount of matrices for an arbitrary number of DCT coefficients of our interest. We are no longer required to somehow select special quantization matrices for each DCT coefficient D_{ij} and each candidate quantization step C_m. In our experiments, we have observed that using matrices similar to those actually used when the image was originally first time compressed, yields slightly better results. However, the difference was almost negligible. We attribute this small difference to integer rounding and truncating in the image domain, when the image was decompressed before its second JPEG compression.

Also, as mentioned above, we used the histograms of absolute values of DCT coefficients in our method. Because the histogram of any AC coefficient is approximately an even function, we thus obtain a better statistics for each feature (local maximum, minimum, double-peak) when using histogram of absolute values.

4.1.2 The Estimation Algorithm

For a given JPEG file:

1. Extract the quantization matrix Q^2 from the JPEG file under inspection.

For every DCT coefficient D_{ij} of interest:

2. Obtain the histogram h_0 of absolute values of quantized DCT coefficient D^q_{ij} from the JPEG image.
3. Crop the image (say, by 4 and 4 pixels).
4. Select the quantization matrices $Q^{1,1}$, …, $Q^{1,n}$ for every candidate value of the primary quantization step.
5. JPEG compress the cropped image with quantization matrices $Q^{1,1}$, …, $Q^{1,n}$.
6. Decompress all JPEG files and compress them again with the quantization matrix Q^2.
7. Compute the histograms $h(Q^{1,1})$, …, $h(Q^{1,n}) = h_1$, …, h_n of absolute values of D^q_{ij} from the double compressed cropped images.

8. Compute $k = \arg\min_m\{||h_m - h_0||\}$.
9. The value of $Q_{ij}^{1,k}$ is the estimated primary quantization step for D_{ij} .

4.2 Compatibility Test with Histogram Properties Checking

Because the algorithm described in the previous section does not give satisfactory performance, we have tried to improve it. We have attempted to use information about local maximums, minimums, and double-peaks to verify if the found quantization step is correct and then select a more probable candidate.

It is not an easy task to match the DCT histogram to the expected local maximums, minimums, and double-peaks. Although the histogram of absolute values of AC coefficients is expected to have the Laplacian distribution, this is only a statistical fact. Histograms obtained from specific images may not have some local maximums, minimums, or double-peaks or may have additional ones caused by the image content.

We have modified Step 8 in the algorithm above in following manner. Instead of computing a single minimum, we took 6 minimal values as candidates to match them to expected histogram properties. In our experiments, we have never observed the situation where the true quantization step value would not be among the 5 minimal values.

An important part of our approach is the determining of expected features (local maximums, minimums, and double-peaks). We can then check if a given histogram of a particular DCT coefficient is compatible with these expected features. It is important that not only the histogram exhibits all expected features (in an ideal case) but that it also exhibits no other features that are not expected.

We have tried to determine local maximums, minimums, and double-peaks in histograms by computing their derivative and finding where it changes its sign. Also, we have simply tried to identify maximums and minimums directly from the set of values. Both these approaches were unsuccessful. This is because very often instead of a local minimum or maximum the histogram slope changes but remains decreasing (i.e., it does not form a minimum). Also, the behavior of double-peaks is very unpredictable. Sometimes, there are only a few local minimums that are not too significant and in combination with double-peaks some of them may disappear.

Finally, we have developed an approach that became partially successful. But this approach differs for some combinations of candidate quantization steps for the primary compression $Q_{ij}^{1,k}$ and the quantization step Q_{ij}^{2} used when second JPEG compression was carried out. Next, we will explain the method together with a method for determination of expected features of the DCT histogram for the candidate quantization step $Q_{ij}^{1,k}$.

For each candidate value of the primary quantization step $Q_{ij}^{1,k}$, we compute the following sequence of values 0, $Q_{ij}^{1,k}$, $2Q_{ij}^{1,k}$, $3Q_{ij}^{1,k}$, …. Then, we divide this sequence by Q_{ij}^{2} and finally round it to integers. Thus, we obtain a sequence of integers $I = \{0, i_1, i_2, i_3, \ldots\}$. If Q_{ij}^{2} is even, we also need to determine the locations of possible double-peaks, by finding all $i_l \in I$, such that $\exists k \in Z$, $kQ_{ij}^1=[(i_l-1)Q_{ij}^2+i_lQ_{ij}^2]/2$. We note, that i_l is always present in the sequence I, but i_l-1 may not. Let us denote the set of all i_l as $I_d \subset I$.

If $Q_{ij}^{1,k}$ and Q_{ij}^2 were equal, I would become $I = \{0, 1, 2, 3, ...\}$ and $I_d=\emptyset$. There should be no special features in the histogram in this case. The same situation occurs when $Q_{ij}^{1,k}$ is a divisor of Q_{ij}^2 because then each integer is in the set I exactly $Q_{ij}^2/Q_{ij}^{1,k}$ times. Although all positive integers may also be in I_d, we still do not expect special features in the histogram due to the reasons given in Section 2.2.1. The histogram h_o of quantized DCT coefficient D^q_{ij} should be relatively smooth as shown in Figure 1a. Smoothness can be measured by the absolute value of the second derivative.

If $Q_{ij}^{1,k}$ is greater than Q_{ij}^2, some integers are missing in the sequence I, but then either some or all of these missing integers may be affected by double-peaks. These are cases of integers $f \notin I, f+1 \in I_d$. Let us denote the set of such f as I'_d. To check the compatibility, we split the histogram h_o into two parts. First, we take $h_0(I^c-I'_d)$ and calculate its maximum value. The set I^c is the complement of I in the set of all integers – it contains integers that are not present in I. On the set $I^c-I'_d$, we expect the histogram to be almost zero. In the second part, we first transform double-peaks to single peak equivalents $\forall f \in I_d$, $h_0(f)=h_0(f)+h_0(f-1)$. After this correction, we expect the histogram $h_0(I)$ (reduced to set I) to be smooth without any special features. We can again assess its smoothness by the absolute value of second derivative. Our experiments showed that this is the only effective method how to deal with double-peaks. They can behave arbitrarily, but every double-peak comes from a single multiple of $Q_{ij}^{1,k}$.

The most complicated case is when $Q_{ij}^{1,k}$ was smaller than Q_{ij}^2, but not a divisor of Q_{ij}^2. The sequence I contains all non-negative integers and at least some of them are there multiple times. Let us built the sequence $C=\{c_0, c_1, c_2, \ldots\}$. For each f, c_f is equal to the number of occurrences of the integer f in the sequence I. Now, we expect that exactly c_f multiples of $Q_{ij}^{1,k}$ should round to f-th multiple of Q_{ij}^2, but we still do not count with double-peaks, if they are present. Unfortunately, in this case, we have no other option than model them statistically. If using independent JPEG implementation of the DCT transform, we have experimentally observed and it has been theoretically proven, that on average double-peaks split in ratio approximately 7:10 between two multiples of Q_{ij}^2. Double-peak originates from one multiple of $Q_{ij}^{1,k}$, therefore we must normalize the ratio to get 1 when summing both parts. So, we adjust the sequence C, $\forall f \in I_d$, $c_f=c_f-7/17$ (Entire double-peak was counted as it should round to f-th multiple of $Q_{ij}^{1,k}$, the part, that should round to $(f-1)$-th multiple is subtracted.), $c_{f-1}=c_{f-1}+7/17$. For the f-th multiple of Q_{ij}^2, c_f tells us how many multiples of $Q_{ij}^{1,k}$ were supposedly quantized to it. Experiments showed that it is generally very difficult to determine how exactly the histogram must look to declare a fit. As Figure 1d shows, some local maximums or double-peaks can be almost hidden in the histogram.

We have found a detection method that at least partially worked. If we adjust the histogram h_o with the formula $h_o(f)=h_o(f)/c_f$ for each non negative integer f, we should get something, what is not perfectly smooth, but is smoother than if the histogram does not fit the expected $Q_{ij}^{1,k}$. As in the previous two cases, we can assess smoothness with the absolute value of the second derivative.

4.2.1 Estimation Algorithm

The algorithm starts as the one described in Section 4.1.2. We just replace Steps 8 and 9 and add additional ones. The replaced and added part follows.

8. Sort $\{||h_m-h_o||\}$ and find 6 indices k, for which $||h_k-h_o||$ takes 6 minimum values.

For each candidate quantization step $Q_{ij}^{1,k}$:

9. If $Q_{ij}^{1,k}$ is a divisor of Q_{ij}^2, compute the quantity $s(k) = \max\{\text{abs}(h''_o)\}$ and go to Step 17.
10. Compute the sequence $I=\text{round}(\{0, Q_{ij}^{1,k}, 2Q_{ij}^{1,k}, 3Q_{ij}^{1,k}, \ldots\}/Q_{ij}^2)$. If Q_{ij}^2 is even, determine also the set $I_d=\{i_l\in I|\exists k\in Z, kQ_{ij}^1=[(i_l-1)Q_{ij}^2+i_lQ_{ij}^2]/2\}$. If Q_{ij}^2 is odd, $I_d=\varnothing$.
11. If $Q_{ij}^{1,k}<Q_{ij}^2$, go to Step 14.
12. Compute $I'_d=\{f\notin I \mid f+1\in I_d\}$, $I^c=\{0, 1, 2, 3, \ldots\}-I$ and $\hat{h}_o= h_o$. Adjust $\hat{h}_o$ according to the formula $\forall f\in I_d$, $\hat{h}_o(f)= \hat{h}_o(f)+ \hat{h}_o(f-1)$.
13. Compute the quantity $s(k) = \max\{\max\{\hat{h}_o(I^c-I'_d)\},\max\{\hat{h}''_o(I)\}\}$ and go to Step 17.
14. Compute the sequence $C=\{c_0, c_1, c_2, \ldots\}$ such that for each f, c_f is equal to the number of occurrences of integer f in the sequence I. Adjust C according to the formula $\forall f\in I_d$, $c_f=c_f-7/17$, $c_{f-1}=c_{f-1}+7/17$. Compute $\hat{h}_o= h_o$.
15. For each $f\in I$, adjust $\hat{h}_o$ as $\hat{h}_o(f)= \hat{h}_o(f)/c_f$.
16. Compute the quantity $s(k) = \max\{\text{abs}(h''_o)\}$.
17. Compute $l = \arg\min_k\{s(k)\}$.
18. $Q_{ij}^{1,l}$ is the estimated primary quantization step for D_{ij} .

4.3 Classification Method

Unfortunately, previous methods, described in Sections 4.1, and 4.2, proved to be computationally expensive and not very reliable. This motivated us to investigate alternate approaches. We attempted to find a similarity measure that would classify a normalized DCT histogram for a given DCT coefficient according to “template” histograms averaged over many double compressed images. However, we were unable to find a similarity measure that would perform satisfactorily. Because the task of finding the “closest” histogram can be naturally formulated as a classification problem, we have decided to use neural network classifiers. This approach indeed proved to be by far the most accurate and reliable one.

For the design of the neural network classifier, we have used the Matlab Neural Network Toolbox [1] and chose the two-layer architecture with a log-sigmoid transfer function. A separate network was designed for each value of the second quantization step Q_{ij}^2. All networks take a vector of length 14 as an input $\{h(2), h(3), \ldots, h(15)\}$, where $h(m)$ is the number of values mQ_{ij}^2 in the histogram of absolute values of the DCT coefficient D_{ij}. The hidden layer had 10, 13, 16, 17, or 22 neurons depending on the value of Q_{ij}^2. The value of 22 was chosen for $1\le Q_{ij}^2\le 5$, 16 for $Q_{ij}^2=6$, 10 for $Q_{ij}^2=7$, 17 for $Q_{ij}^2=8$, and 13 for $Q_{ij}^2=9$. The number of neurons in hidden layers was determined according to the performance of trained networks. The output layer had 8 neurons for $Q_{ij}^2>1$ and 9 neurons when Q_{ij}^2=1 because in this case we must also consider the value 1 as a possible value for Q_{ij}^1. In the ideal case, the output vector would consist of one value 1 and zeros. The location of 1 in the vector tells us, which value of Q_{ij}^1 was actually used. In practice, we have used the maximum element in the vector to determine the network output. Note that all elements of the output vector always lay in the interval [0, 1] due to the range of the sigmoid transfer function.

To prepare the training set, we took 100 images from various archives, all originally stored as JPEGs obtained using at least 5 different digital cameras. Their dimensions varied from 700×500 to 2050×1550. To remove possible artifacts due to previous JPEG compression or other processing, we resampled all images to 33% of their original size, converted them to grayscale, and saved as BMPs.

For each network (i.e., for each value of Q_{ij}^2), we have prepared the training set by double compressing each image using matrices with specific quality factors: 61, 65, 70, 75, 79, 84, 88, and 90. These factors were chosen because the lowest frequency quantization steps for these factors are the same, ranging from 2 to 9. Thus, for example, for the secondary quantization step Q_{ij}^2=6, each test image was first compressed using the above 8 matrices, followed by a compression using the factor 75. To create the training set for Q_{ij}^2=1, we also used the quality factor 95 for both primary and secondary quantization matrices.

Then, for each double compressed image we calculated the histogram of absolute values of the quantized DCT coefficient D^q_{ij}. The histogram values for multiples 0 and 1 were removed and the remaining values were normalized (so that their sum was 1). Because the shape of histograms of individual DCT coefficients changes with their frequency, we included in the training set only histograms corresponding to low-frequency coefficients D_{11} and D_{10} (this means that each neural network can be used for estimation of primary quantization steps of low-frequency coefficients, only). Thus, from a database of 100 training images, we obtained a training set with 100×8×2=1600 (for Q_{ij}^2>1) and 1800 (for Q_{ij}^2=1) DCT histograms. For indistinguishable cases (see Sections 2.2.1, and 3), the training sets were modified, so the correct output of the network was the value Q_{ij}^2. The Bayesian regularization, a modification of the Levenberg-Marquardt training algorithm [1] to produce networks that generalize well, has been used for training all of our networks. We used this algorithm because we experienced problems with other commonly used algorithms and we were unable to solve them by adjusting the number of neurons in the hidden layer.

4.4 Estimation Algorithm

For a given JPEG file:

1. Extract the quantization matrix Q^2 from the JPEG file under inspection.

For every DCT coefficient D_{ij} of interest:

2. Obtain the histogram h_0 of absolute values of the quantized DCT coefficient D^q_{ij} from the JPEG image, remove the values $h_0(0)$ and $h_0(1)$, and normalize it.
3. Load the neural network $\mathcal{N}$ corresponding to value of Q_{ij}^2.
4. Put the normalized histogram h_0 as input into the network $\mathcal{N}$.

5. Output of neural network $\mathcal{N}$ gives the estimated primary quantization step for D_{ij}.

5. Experiments and Results

All test images were originally obtained using 4 different digital cameras as JPEGs, only two of them were originally BMPs. To remove JPEG artifacts we have resized JPEG images to 83% of their original size using PaintShop Pro 7 and saved them as BMPs. All images were also converted to grayscale. The resulting images in the set have 5 different sizes ranging from 744×491, the smallest, to the largest 1275×850. We should also note that no image in this testing set was used for training of our neural networks.

At first, we tested the network on single compressed images. We created single-compressed JPEGs from our testing images using 9 different standard JPEG quantization matrices (180 different JPEG files). Then, we tested the neural network on this set by trying to identify the primary quantization step for DCT coefficients D_{01}, D_{11}, and D_{10}. Because for single compressed images, all DCT histograms look the same way as if the primary quantization step was a divisor of Q_{ij}^2 (including 1 and Q_{ij}^2), the network should return the secondary quantization steps from the JPEG file. It has been trained this way.

The network misclassified 4 DCT coefficients out of 3×180=540 examined DCT coefficients in 4 different files. Two files were of the size 1062×797 and two of the size 744×491. In these error cases, the network always estimated 2 as the primary quantization step. This occurred once for secondary quantization steps 5, 7, and twice for the value 9. We reiterate that in single compressed images, all DCT histograms are fundamentally indistinguishable from double compressed images with primary quantization step that are divisors of Q_{ij}^2.

We have continued our experiments on double compressed images. To prepare the test images, we have used standard quantization matrices corresponding to quality factors 61, 65, 70, 75, 79, 84, 88, 90, and 95 for both primary and secondary quantization matrices. We have also included cases when the primary quantization matrix was non-standard (see Section 3). Using all possible combinations of these quantization matrices, we have prepared the total of 900 different JPEG files. The neural networks were tested on this set. We were again looking for primary quantization steps of the same three DCT coefficients D_{01}, D_{11}, and D_{10}.

The method misclassified 25 DCT coefficients in 24 different files. Thus, the error rate of the neural network classifier was below 1% because out of 3×900=2700 examined DCT coefficients, only 25 were misclassified. Table 1 shows the number of misclassified cases for different combinations of quality factors. Note that for each combination there were 20 files and 3 examined coefficients in each file, which means 60 coefficients total. As Table 1 shows, errors are more likely to occur when the primary quantization step is followed by a large secondary step, i.e., when the double compression decreases the image quality.

Primary quantization matrix (quality factor)	Secondary JPEG compression quantization matrix (quality factor)								
	61	65	70	75	79	84	88	90	95
61	2	0	0	0	0	0	0	0	0
65	1	0	0	0	0	0	0	0	0
70	0	1	1	0	0	0	0	0	0
75	0	0	0	0	0	0	0	0	0
79	0	0	0	0	1	0	0	0	0
custom	1	0	0	0	1	0	0	0	0
84	0	0	0	0	0	0	0	0	0
88	2	1	0	0	0	0	0	0	0
90	6	0	3	0	0	0	0	0	0
95	3	0	1	0	1	0	0	0	0

Table 1: Number of misclassified quantization steps for combinations of quantization matrices. Total of 60 quantization steps were estimated for each combination.

Additional experiments were performed with different quantization matrices. These experiments yielded similar results.

We have also investigated the influence of the image size on the number of misclassifications. For this purpose, we took two 1275×850 images from the previously described testing set. We have intentionally selected images that did not yield any misclassifications in our experiments. From these two images, we have created smaller images of 90, 80, 70, 60, 50, 40, 30, 20, and 10% of original dimensions. We have repeated our experiments on this set.

Table 2 shows how the number of errors depends on the image size. For each size, we had 2 files, double compressed by 90 combinations of quantization matrices. In each double compressed file, we have estimated 3 quantization steps for DCT coefficients D_{01}, D_{11}, and D_{10}. In other words, for each image size we have estimated 540 quantization steps.

File-size (% of 1275×850)	10	20	30	40	50	60	70	80	90	100
Number of errors	119	37	11	14	7	0	0	1	0	0

Table 2: Number of misclassified quantization steps (out of 540 estimated) for different file-sizes.

Table 2 shows that the network still performs well for about 50% of the original image dimensions. The results for 30% of original image dimensions are still acceptable considering how small the image is (255×382). Obviously, the results get then progressively worse because very small images do not provide sufficient statistical data for reliable classification.

Another matter of our interest was performance of our current neural network classifier on DCT histograms of coefficients with different frequencies. Provided we have DCT coefficient D_{ij} we define frequency of this coefficient as $i+j$. Originally, we have tried to add frequency parameter to the classifiers and retrain neural networks again. We expected this could improve their performance. However, results of our experiments did not show any noticeable improvement, and this was the reason we abandoned this idea and returned to our original network trained on DCT histograms of coefficients D_{11} and D_{10} (frequencies 1 and 2).

In this moment the question about performance of our current classifier on higher frequencies became apparent. We have made experiments on DCT coefficients of frequency 3 (D_{30}, D_{21}, D_{12}, and D_{03}), we have used the same BMP files as in our first experiments. The results can be seen in the Table 3. For every secondary quantization step we have unique classifier, this classifier estimated four coefficients from 200 double-compressed images with different primary quantization steps. In fact, this classifier estimated 800 primary quantization coefficients.

Secondary quantization step	1	2	3	4	5	6	7	8	9
Number of errors	0	0	0	0	35	1	21	34	64

Table 3: Number of misclassified quantization steps (out of 800 estimated) on frequency 3.

We can see, results for values less than 5 are excellent. For value 5, we think it might be possible to retrain the classifier better, however, the error is still below 4.4%. We can see rising number of errors as secondary quantization step increases. We expect insufficient statistics for higher frequencies and these results correspond to our expectations. However, even for these high values, percentage of errors did not exceed 8% in this particular experiment.

6. Conclusions

We have presented solution to the problem of JPEG double compression detection and estimation of the primary quantization matrix. Our experiments indicate that our best method – the neural network classifier – is sufficiently reliable yielding less than 1% of errors. The estimation is also very fast and significantly outperforms previous approaches described in Sections 4.1 and 4.2.

In this paper, we have focused on the problem of estimating primary quantization steps for selected low-frequency DCT coefficients. However, in practice we may need the entire quantization matrix. Because coefficients of higher frequencies are more often quantized to zero than low-frequency coefficients, we believe that reliable estimation of higher frequency coefficients is not possible due to insufficient statistics. Results of our experiments support this expectation. We recommend first estimating selected low-frequency quantization steps and then estimate the rest of the primary quantization matrix from some standard matrix whose low-frequency steps are close (identical) to the estimated coefficients.

We have identified four limitations of our neural network classifier, some of which, we believe, are fundamental and unavoidable. (1) As already mentioned in this paper, some combinations of quantization steps are indistinguishable from single JPEG compression. This happens when the primary quantization step is a divisor of the secondary quantization step. (2) We have also experimentally documented that sufficiently large images are required for our method to work, because small images do not contain sufficient statistics. (3) Because histograms of multiple compressed JPEGs exhibit different features than histograms of double compressed ones, our neural network estimator would most likely not perform satisfactorily in those cases. (4) It may not be possible to reliably estimate quantization steps for high-frequency coefficients due to insufficient statistics. We have observed insufficient statistics on middle-frequency coefficients. However, base on our experiments, we currently, do not think it is necessary to train another classifiers for middle-frequencies.

In this paper, we only dealt with AC coefficients but we strongly believe that our method could be easily extended to work for the DC term, as well. Although the DC term histogram has a different distribution, the features introduced by double compression are the same. The reliability of the estimates could be improved by testing the estimated primary quantization steps whether they form a plausible matrix. Even custom matrices usually have close values for similar frequencies. The proposed method could also be improved by removing saturated image blocks from processing. It has been shown in [2] and [3] that DCT coefficients from saturated blocks have different statistical properties than coefficients from unsaturated blocks. Finally, a possible extension of our method already mentioned in this paper is to work with chrominance and luminance components in color JPEG images.

As already mentioned in the introduction, reliable estimation of primary quantization steps is quite important in steganalysis [4],[5] and it may find applications in forensic image analysis. Imagine someone creates a digital manipulation by copying an area from one image and pasting it to another image, already stored as JPEG. If we have a suspicion which area of the image has been tampered and if the area is sufficiently large, we can run our algorithm separately on the suspected area and the rest of the image. It is very improbable that the tampered area would yield the same original quantization matrix as the rest of image. That could happen only when the tampered area was copied from a JPEG image saved with the *same* quantization matrix *and* if the 8×8 raster of the copied area matched with the 8×8 raster of the image being tampered.

Acknowledgements

The work on this paper was supported by Air Force Research Laboratory, Air Force Material Command, USAF, under a research grant number F30602-00-1-0521. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation there on. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of Air Force Research Laboratory, or the U. S. Government.

References

[1] Demuth H. and Beale A.: "Neural Network Toolbox User's Guide", version 3, for use with MATLAB, *MathWorks, Inc.*, Natick, MA, 1998.
[2] Fan Z. and de Queiroz R. L.: "Identification of Bitmap Compression History: JPEG Detection and Quantizer Estimation", in *IEEE Transactions on Image Processing*, vol. 12, No. 2, February 2003, pp. 230–235.
[3] Fridrich J., Goljan M., and Du R.: "Steganalysis Based on JPEG Compatibility", Special session on Theoretical and Practical Issues in Digital Watermarking and

Data Hiding, *Multimedia Systems and Applications IV*, Denver, Co, USA, August 19–24, 2001, pp. 275–280.

[4] Fridrich J., Goljan M., and Hogea D.: "Attacking the OutGuess", Proc. *Multimedia and Security, Workshop at ACM Multimedia*, Juan-les-Pins, France, December 6, 2002.

[5] Fridrich J., Goljan M., and Hogea D.: "Steganalysis of JPEG Images: Breaking the F5 Algorithm", Proc. *5th International Workshop on Information Hiding*, Noordwijkerhout, The Nederlands, October 7–9 2002, pp. 310–323.