



A Reliable JPEG Quantization Table Estimator

Tina Nikoukhah, Miguel Colom, Jean-Michel Morel, Rafael Grompone von Gioi

► To cite this version:

Tina Nikoukhah, Miguel Colom, Jean-Michel Morel, Rafael Grompone von Gioi. A Reliable JPEG Quantization Table Estimator. *Image Processing On Line*, 2022, 12, pp.173-197. 10.5201/ipol.2022.399 . hal-03858141

HAL Id: hal-03858141

<https://hal.science/hal-03858141>

Submitted on 17 Nov 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Reliable JPEG Quantization Table Estimator

Tina Nikoukhah, Miguel Colom, Jean-Michel Morel, Rafael Grompone von Gioi

Université Paris-Saclay, ENS Paris-Saclay, CNRS, Centre Borelli, France
 {tina.nikoukhah, colom-barco, morel, grompone}@ens-paris-saclay.fr

Communicated by Jose-Luis Lisani

Demo edited by Tina Nikoukhah

Abstract

JPEG compression is a commonly used method of lossy compression for digital images. The degree of compression can be adjusted by the choice of a quality factor QF . Each software associates this value to a quantization table, which is a 8×8 matrix used to quantize the DCT coefficients of an image. We propose a method for recovering the JPEG quantization table relying only on the image information, without any metadata from the file header; thus the proposed method can be applied to an uncompressed image format to detect a previous JPEG compression. A statistical validation is used to decide whether significant quantization traces are found or not, and to provide a quantitative measure of the confidence on the detection.

Source Code

The reviewed source code and documentation for this algorithm are available from the web page of this article¹. Compilation and usage instruction are included in the `README.md` file of the archive.

Keywords: JPEG compression; quantization; a contrario method; DCT coefficient

1 Introduction

Image forensics [8] aims at revealing the operations undergone by an image during the camera pipeline [3] or afterwards. An important example of such operations is JPEG compression, which depends on a quality factor (QF) parameter. This quality factor is associated to two quantization tables (Q-tables), which are 8×8 matrices of integer values; the JPEG standard does not specifies the Q-tables for each QF, which depend on the particular software used for compression. This work describes a reliable method for estimating the main JPEG quantization table used during compression based solely on the decompressed digital image.

When the image to be analyzed is encoded in a JPEG file, there is no need to detect the JPEG quantization tables, as all the relevant information is contained in the header of the JPEG file

¹<https://doi.org/10.5201/ipol.2022.399>

itself. Indeed, these tables are required for JPEG decompression. Nevertheless, even in that case, it may be interesting to detect traces of a previous JPEG compression. More importantly, in forensic applications one wants to analyze images which may have been converted to any file format and study whether one or more JPEG compressions were applied. For these reasons, the method described here takes only the pixel values of the image as input and uses no metadata information nor header information that may be contained in the image file.

Fan and de Queiroz [5, 6] proposed a method to extract the quantization table using maximum likelihood estimation. Their algorithm gives good results but the estimation performance deteriorates at very high bit rates ($QF > 95$). In [10], Benford's law is applied to the DCT coefficients. The method works by re-compressing the image with several QFs and fitting the different distributions of the DCT coefficients to the proposed law. The QF of the version having the least fitting artifact is chosen and its corresponding Q-table is selected. Both methods require that the possible Q-tables are known in advance and estimate the complete table by looking up in a list of common tables [21]. In [20], a statistical model for quantized coefficients is introduced with better accuracy than the Laplacian model, yet it is time consuming. The closest approach to ours is Ye's method et al. [23]. These authors propose an estimation method based on the power spectrum of the histogram of the DCT coefficients. After low-pass filtering the second derivative of the power spectrum, they count the number of local minima to establish the quantization step.

The method described here uses a statistical test, based on Desolneux, Moisan and Morel's *a contrario* theory [4], to control the number of false detections. This test allows one to compute a quantitative measure of the confidence associated to each element of the table, and to reject the estimation when this measure is not good enough. When no element of the quantization table is found, this may indicate that the image has not gone through a JPEG compression, or that an operation has been done after the compression that tampered the JPEG history of the image.

This work is organized as follows. Section 2 provides a brief summary of the JPEG algorithm. Then, the JPEG quantization table estimation method is described in Section 3. Several experiments are analyzed in Section 4. Finally, Section 5 concludes the paper.

2 JPEG Compression

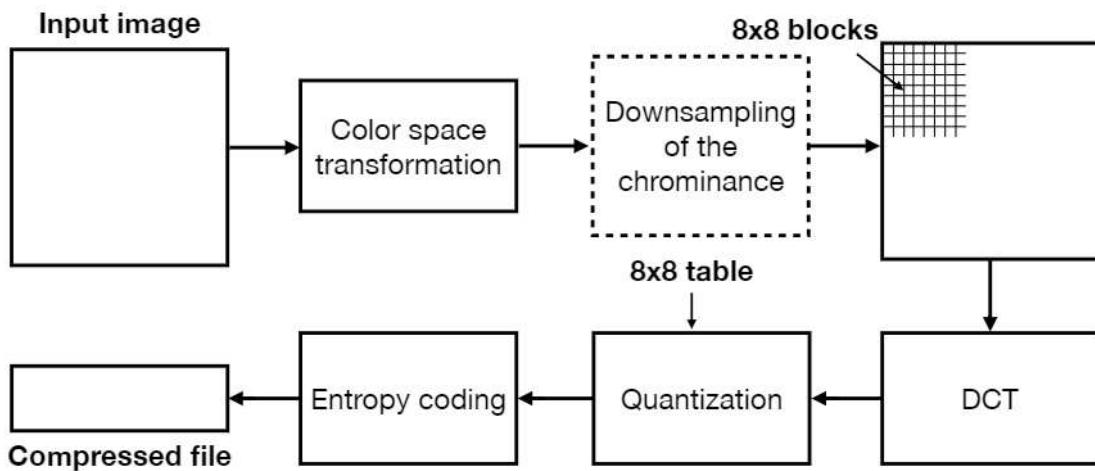


Figure 1: The JPEG compression pipeline.

The JPEG algorithm (ISO/IEC 10918 — ITU-T Recommendation T.81 [1]) is currently the most common method for compression of digital photography. The encoding process, shown in Figure 1, is detailed in the following. The first step is to perform a color space transformation from RGB to

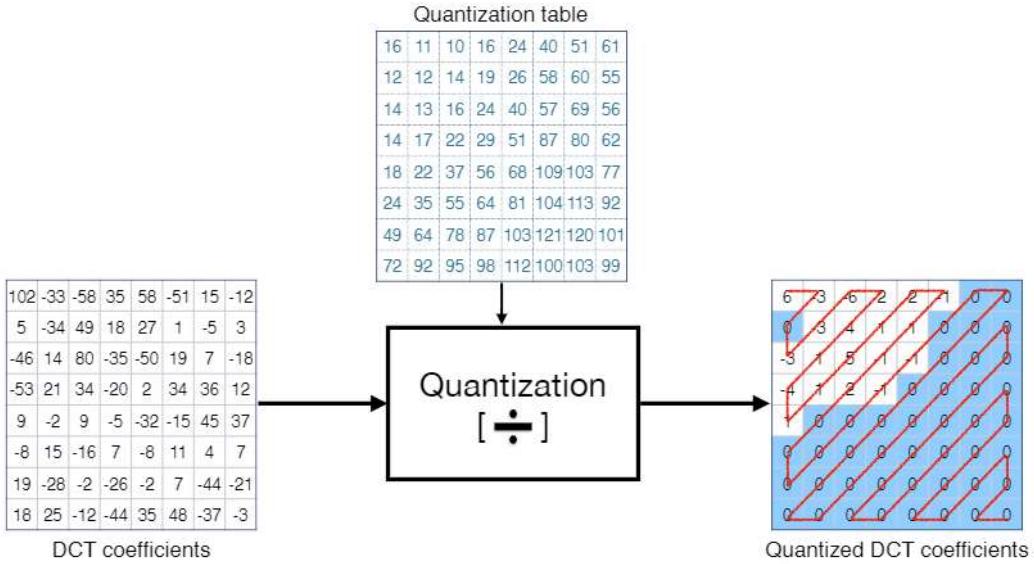


Figure 2: The impact of quantization on an example DCT block. Each DCT coefficient is quantized by the associated value in the quantization matrix. Rounding leads to setting to zero many of the high frequency coefficients. Each block is scanned in a zig-zag pattern to be encoded as a vector with a sequence of zeros.

$Y C_B C_R$, where Y is the luminance component and C_B and C_R are the chrominance components of the blue and red difference. The chroma channels are optionally downsampled in the JPEG algorithm; the most common options are using full chroma resolution (noted 4:4:4), downsampling by a factor of 2 only in the horizontal direction (noted 4:2:2), or reduction by a factor of 2 in both the horizontal and vertical directions (noted 4:2:0).

The value 127 is subtracted from each pixel to obtain values with a distribution that is roughly centered around zero. Then, each channel is divided into non-overlapping 8×8 blocks and each block is processed independently. The type II 2D Discrete Cosine Transform (DCT) is applied to each block and the coefficients are quantized according to a given table (Figure 2 shows an example). This quantization table provides a factor for each DCT component and determines the compression level; the larger the factors, the lower the resulting file size, but also the lower the image quality. Quantization tables may have an associated quality factor QF , which is an integer in the range from 1 (high compression ratio, worst quality) to 100 (lowest compression ratio, best quality). Note that different Q-tables may be associated to the same QF factor; the Q-tables are one of the primary sources of variability among JPEG encoders [7].

Figure 3 shows a 3000×2000 pixels JPEG picture and its quantization table. In natural images, most of the energy is concentrated in the low and medium frequencies. Thus, the quantization tables usually have larger values (entailing more compression) for higher frequencies, as it is the case in the figure.

A lossless compression by Run-Length Encoding (RLE) exploits the long series of zeros at the end of each vector. A Huffman code then allows a last lossless compression of the data, to which a header is finally added to form the final JPEG file. The quantization table is stored in the header of the JPEG file to allow for decoding.

For each quality factor, there are two associated Q-tables, one for the luminance channel and the second one for the chroma components. In this work, we focus on the luminance channel. The method presented here could be extended to the chroma channels and estimate the second quantization table; this requires, however, knowing the chroma subsampling factors or to estimate them.



2	1	1	2	2	4	5	6
1	1	1	2	3	6	6	6
1	1	2	2	4	6	7	6
1	2	2	3	5	9	8	6
2	2	4	6	7	11	10	8
2	4	6	6	8	10	11	9
5	6	8	9	10	12	12	10
7	9	10	10	11	10	10	10

Figure 3: JPEG-encoded picture and its quantization table for the Y channel. The JPEG quality factor of this table is $QF = 95$.

3 Q-Table Estimation

The method proposed here starts by analyzing the DCT coefficients of the luminance channel of the image. The method focuses on the 63 AC coefficients and leaves the DC coefficient (which has different properties) untreated. The histogram of each of the 63 coefficients is analyzed and each quantization value between 1 and 255 is evaluated. A statistical test based on the a contrario theory [4] is used to decide which quantization values are significant for each coefficient, and to select the best quantization value among the significant ones. In principle, the method does not require that the JPEG blocks be aligned to the standard JPEG grid starting at pixel $(0, 0)$. This can be false if, for instance, the image has been cropped after JPEG compression. In such a case, the same algorithm could be applied to each of the 64 possible JPEG grid origins; valid detections would only be produced for the correct alignment. Nevertheless, due to the existence of reliable JPEG grid detectors [17, 18, 19] this exhaustive search can be avoided. The presented algorithm therefore assumes that the JPEG blocks are aligned to the standard grid. However, the statistical tests will not make such hypothesis, so that, if required the method could be applied to all the possible 64 JPEG grid origins to find the right alignment. (This is important when the algorithm is used as a detector of falsified regions: such regions may have a different quantization table and their origin differ from the one of the main image.) Algorithm 1 provides a pseudo-code of the full method whose steps are detailed below.

3.1 The DCT Coefficient Histogram

The JPEG DCT quantization step has a clear effect on the histogram of the DCT coefficients of an image. This effect is illustrated in Figure 4 where the histogram of the same DCT coefficient is plotted before and after the JPEG compression. Even when the pixel values are integers, the DCT coefficients are real values. The JPEG quantization step transforms each DCT coefficient into an integer, multiple of the quantization value q [9]. Then, the JPEG decompression process transforms the integer DCT coefficients into real pixel values, which are later rounded to integer pixel values. Due to the latter rounding, the DCT coefficients of the uncompressed image are no longer integers but present a narrow distribution around the quantization values, as can be seen in Figure 4 (right). The quantization value in Figure 4 is $q = 6$, and therefore the uncompressed coefficients are centered around values 0, 6, -6, 12, -12, and so forth.

Estimating the quantization table reduces to deciding if these periodic peaks are present or not in the histogram. Given one of the 63 AC DCT coefficients c and a candidate quantization value q , the algorithm computes the quantization error for each block b_i of the image. That is, given the value $v_i = \text{DCT}(b_i, c)$, the nearest multiple of q determines the corresponding peak number

Algorithm 1: JPEG Quantization Table Estimation

```

input : Image  $I = (R, G, B)$ 
output: Quantization table  $\mathbf{Q}$ 
output: Associated NFA table
1  $\mathbf{Q}[\cdot] \leftarrow \text{NON VALID}$  initialize as not detected
2  $\mathbf{NFA}[\cdot] \leftarrow \infty$ 
3  $Y \leftarrow \text{Round}(0.299 R + 0.587 G + 0.114 B)$  compute luminance image
4 for  $c \in \{1 \dots 63\}$  do loop over DCT coefficients
5   for  $q \in \{1 \dots 255\}$  do loop over quantization values
6      $s \leftarrow 0$ 
7      $n \leftarrow 0$ 
8     for  $b \in \text{JpegBlocks}(Y)$  do loop over all blocks of the image
9        $v \leftarrow \text{DCT}(b, c)$  compute DCT coefficient number  $c$  on block  $b$ 
10       $V \leftarrow \text{round}\left(\frac{v}{q}\right)$  nearest quantized value
11      if  $V \neq 0$  then avoid central peak
12         $e \leftarrow 2 \left| \frac{v}{q} - V \right|$  normalized quantization error
13         $s \leftarrow s + e$  sum of normalized quantization errors
14         $n \leftarrow n + 1$  number of considered values
15    $\text{nfa} \leftarrow 64 \times 63 \times 255 \times \mathbb{P}(S_n \leq s)$  NFA value (see Equation (11) for the p-value term)
16   if  $\text{nfa} \leq 1$  and  $\text{nfa} < \mathbf{NFA}[c]$  then store detected q
17      $\mathbf{Q}[c] \leftarrow q$ 
18      $\mathbf{NFA}[c] \leftarrow \text{nfa}$  store NFA

```

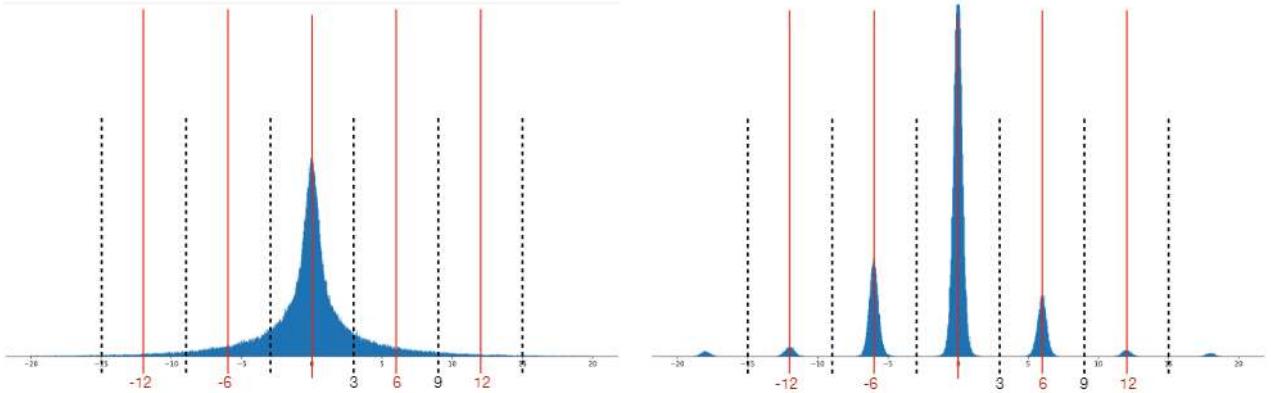


Figure 4: Histogram of a DCT coefficient for an uncompressed image (left) and after JPEG compressed with quantization value $q = 6$ (right).

$V_i = \text{round}\left(\frac{v_i}{q}\right)$. The normalized quantization error is

$$e_i = 2 \left| \frac{v_i}{q} - V_i \right|, \quad (1)$$

where a multiplicative factor 2 is set to normalize the values into the range $[0, 1]$. A normalized quantization error $e_i = 0$ means that the coefficient is an exact multiple of q , whereas $e_i = 1$ means that the coefficient is at mean distance from the previous or the next quantization values.

The error distribution reflects how concentrated the DCT coefficients are around the quantization values. Consider for example the coefficients with values around 6 in the non-compressed image, see Figure 4 (left). The distribution decreases from values 3 to 9, the range of values corresponding to the first peak ($V_i = 1$). However, the normalized quantization errors e_i are the average of the negative errors (values from 3 to 6) and the positive errors (values from 6 to 9). Thus, the normalized quantization error is roughly uniformly distributed for non-quantized DCT coefficients; Figure 5 (left) shows the normalized quantization error distribution corresponding to Figure 4 (left), which is approximately uniform. (This is not true for the central peak ($V_i = 0$) which due to symmetry shows a non-uniform distribution, highly concentrated on small values, even in uncompressed images; for this reason it will be ignored in the proposed method, see step 11 in Algorithm 1.) On the other hand, when the DCT coefficients are quantized, the normalized quantization error distribution has a clear concentration on small values, as shown in Figure 5 (right). The next section presents the statistical test used to decide between these two cases.

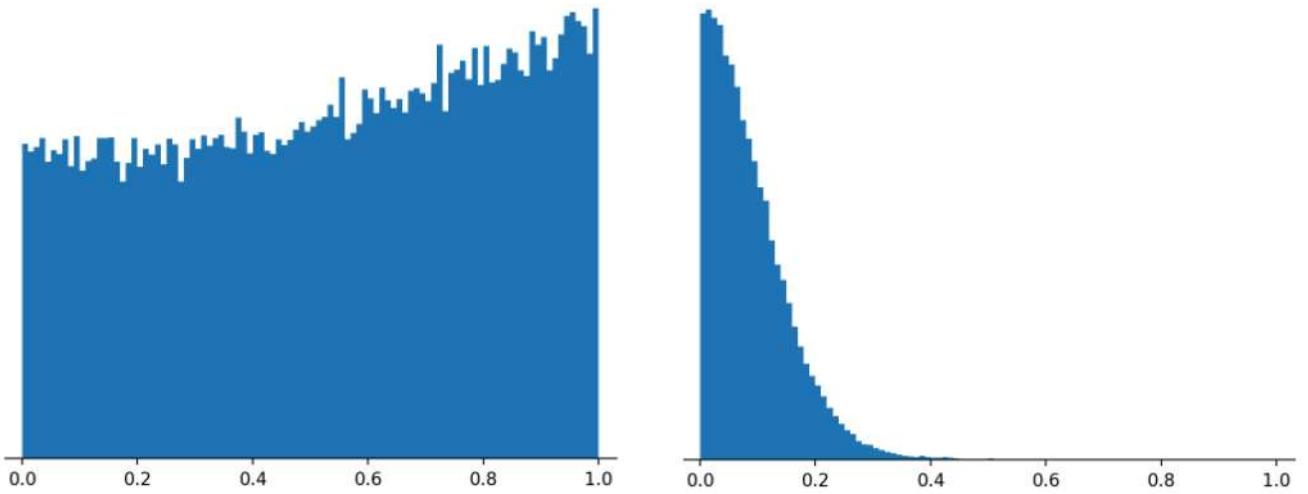


Figure 5: Histogram of the normalized quantization errors of a DCT coefficient for uncompressed (left) and JPEG compressed (right) images. Normalized quantization errors of JPEG compressed images are highly concentrated on small values, while an approximately uniform distribution is observed in uncompressed images (or rather moderately concentrated on large values).

3.2 Statistical Validation

The proposed validation procedure is based on the a contrario theory, which relies on the non-accidentalness principle [16, 22]. Informally, this principle states that there should be no detection in noise. In the words of D. Lowe, “*we need to determine the probability that each relation in the image could have arisen by accident, $P(a)$. Naturally, the smaller that this value is, the more likely the relation is to have a causal interpretation*” [16, p. 39]. This principle has shown its practical use for detection purposes such as segment detection [12], vanishing points detection [15], anomaly detection [2], or forgery detection [19].

In our context, we need to assess whether the DCT coefficients are quantized. This is translated into deciding whether the normalized quantization error distribution is significantly concentrated on small values or not. We define a stochastic null model H_0 where the normalized quantization errors E_i are independent random variables, uniformly distributed in $[0, 1]$. Then, we define the statistic

$$s = \sum_{i=1}^n e_i,$$

where e_i is the normalized quantization error defined by (1). Hence, s can take values in the range $[0, n]$. For non-compressed values one would expect to observe $s \approx \frac{n}{2}$, while a value of $s \approx 0$ would indicate a distribution concentrated around the quantization values. Thus, when the value s is small enough we can reject the null hypothesis H_0 and conclude that the values are indeed quantized. The question is for which values s is considered small enough.

Under the null hypothesis H_0 , s becomes a random variable S_n , the sum of n independent and uniformly distributed random variables E_i . (The subscript n was added to remind the number of elements in the sum.) Thus, S_n follows the Irwin-Hall distribution [14] and its p -value is given by

$$\mathbb{P}(S_n \leq s) = \frac{1}{n!} \sum_{k=0}^{\lfloor s \rfloor} (-1)^k \binom{n}{k} (s - k)^n. \quad (2)$$

Given an observed value s , $\mathbb{P}(S_n \leq s)$ is the probability of obtaining by chance such a small value under H_0 . When this probability is small enough, there exists evidence to reject the null hypothesis and declare that a significant quantization step was detected.

However, the multiplicity of tests needs to be taken into account when considering that the probability is small enough. To use an analogy, even if the odds of each individual lottery ticket is 1/1000, the chances of winning are very high when buying 1000 tickets. Similarly, if 1000 tests were performed, it would not be surprising to observe an event that appears with probability 1/1000 under random conditions. The number of tests N_T needs to be included as a correction factor, as it is standard in statistical multiple hypothesis testing [11]. The null hypothesis H_0 is rejected when $\mathbb{P}(S_n \leq s) \leq \frac{\varepsilon}{N_T}$ for a predefined value ε ; this is called the Bonferroni correction. Equivalently, and following the a contrario literature, we define the Number of False Alarms (NFA) of a candidate (c, q) as

$$\text{NFA}(c, q) = N_T \mathbb{P}(S_n \leq s), \quad (3)$$

and detections are declared when $\text{NFA}(c, q) \leq \varepsilon$, implying that the DCT coefficient c presents significant quantization traces for value q . One can show [4] that under the null hypothesis H_0 the expected number of false alarms with $\text{NFA} \leq \varepsilon$, is bounded by ε

$$\mathbb{E}_{H_0} \left[\sum_{(c,q) \in \mathcal{N}_T} \mathbf{1}_{\text{NFA}(c,q) \leq \varepsilon} \right] < \varepsilon,$$

where \mathcal{N}_T is the set of N_T tests. As a result, ε corresponds to the mean number of false detections per image under H_0 . In most practical applications, the simple value $\varepsilon = 1$ is suitable; we will set it once and for all in our application as well. With this choice, the expected number of false detections per image is guaranteed to be upper-bounded by 1.

In the present case, one test is performed for each of the 63 DCT coefficients, and in each case 255 quantization values are tried. Also, at least in principle, all possible 64 JPEG grid origins could be tried. Then, the number of tests is

$$N_T = 64 \times 63 \times 255.$$

Finally, a candidate q is accepted as a significant quantization of coefficient c when

$$\text{NFA}(c, q) = 64 \times 63 \times 255 \times \mathbb{P}(S_n \leq s) \leq 1. \quad (4)$$

It is possible that more than one quantization candidate q leads to a significant validation. This can happen for instance for a divisor q' of the correct $q = m \cdot q'$. For example, the histogram in Figure 4 (right) has a quantization factor $q = 6$. A significant quantization is also obtained when

evaluating the same histogram with $q' = 3$; indeed, the histogram *is* quantized with such step, only that it is not the best fit, as half the quantization values are not represented. Fortunately, the statistics of the normalized quantization errors allows one to pick the best value. Figure 6 plots the normalized quantization error distribution when the histogram of Figure 4 (right) is analyzed with quantization values 3, 6, 12, and 13. For $q = 3$ and $q = 6$ the distributions are concentrated on the small values, leading to correct significant validations of both cases. However, the distribution is more concentrated on small values with the right $q = 6$ value. Thus, the corresponding NFA is smaller. As a result, if more than one quantization value is significant for a given coefficient, the one with smaller NFA value is selected. Note that a multiple of the right quantization value does not lead to a significant detection. See the case with $q = 12$ in Figure 6 as an example. Indeed, half the histogram peaks are placed at the interval ends, leading to a U distribution which does not pass the statistical test. A wrong quantization value leads to a mixed distribution that is not validated either, see the case with $q = 13$.

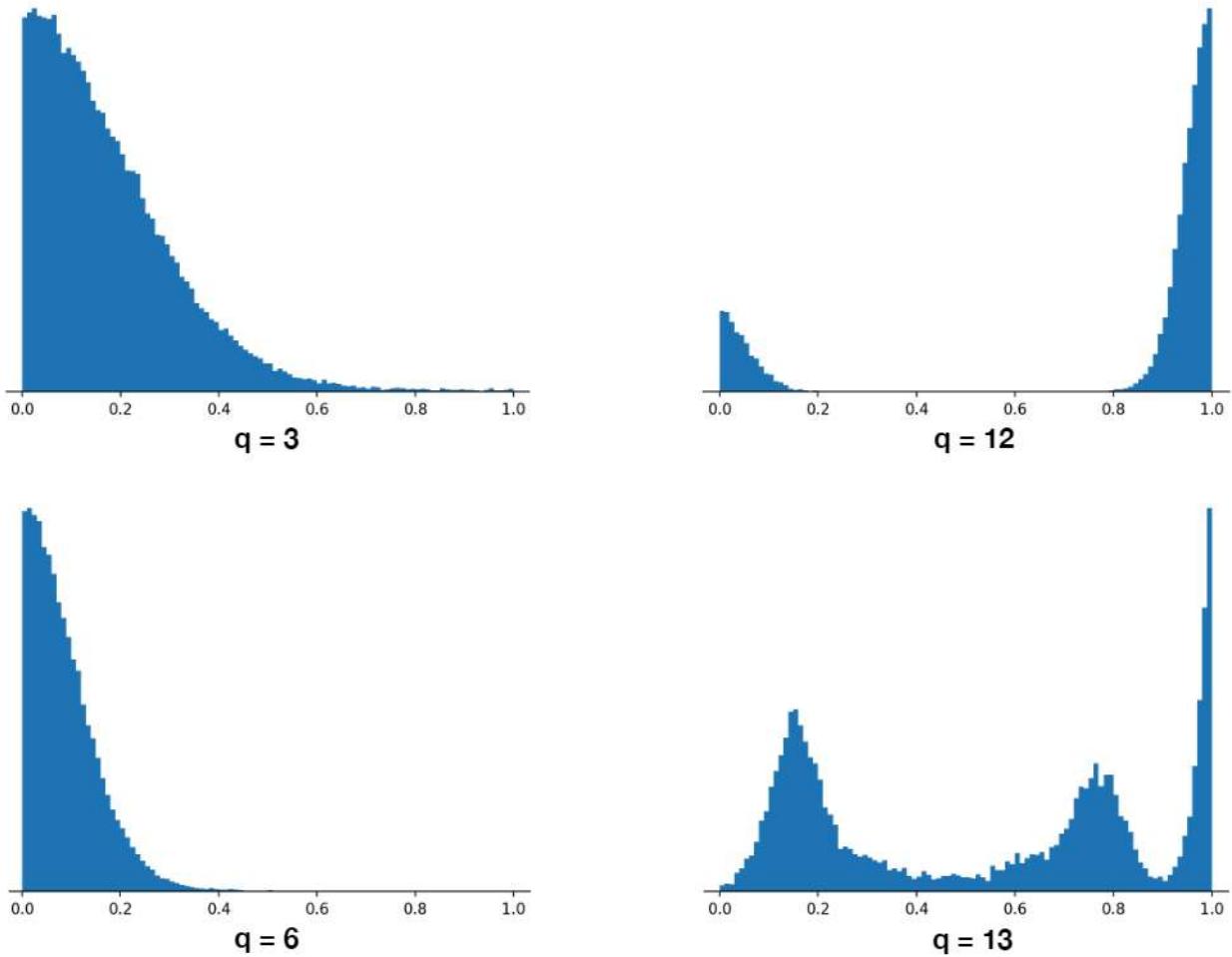


Figure 6: Normalized quantization error distributions for the coefficient in Figure 4 (below) when evaluated with quantization values of $q = 3$, $q = 6$, $q = 12$ and $q = 13$.

Some clarifications are required concerning the implementation of the p -value term in Equation (2). Computing the actual sum would be too expensive and may also lead to numerical representation problems. Indeed, the NFA may reach very small values, which may underflow the usual IEEE 754 number representation. Our implementation in the C programming language, which uses IEEE 754 number representation, computes $\log_{10}(\text{NFA})$ instead of NFA, allowing for a larger numeric range. Any logarithm base is equally useful for this purpose; the 10 base makes it slightly easier to read the order of magnitude of the NFA values. Of course, the test must now compare $\log_{10}(\text{NFA})$

to $\log_{10}(\varepsilon)$, which for $\varepsilon = 1$ is zero.

To ease the computational burden, a simpler upper-bound is used to approximate the actual sum in Equation (2). Because an upper-bound is used, if the approximated NFA is smaller than the detection level, the same is true for the exact NFA. Thus, no false detection results from this approximation. Conversely, it is possible to fail to make a detection when the actual value is below the detection level while the approximation is not. Nevertheless, this does not represent an important risk as quantized DCT coefficient distributions usually result in very small probability terms, and even the upper bound usually takes very small values. Numerical experiments confirmed that the impact of this approximation is limited.

Two different upper-bounds for the p -value are actually used to approximate its number. First, it can be shown that the first term of the sum in Equation (2) is an upper bound to the full sum,

$$\mathbb{P}(S_n \leq s) \leq \frac{s^n}{n!}. \quad (5)$$

This term can be easily computed using Stirling's formula to get an lower-bound on the factorial

$$\sqrt{2\pi n} \left(\frac{n}{e} \right)^n < n!, \quad (6)$$

resulting in

$$\mathbb{P}(S_n \leq s) \leq \frac{s^n}{n!} \leq \frac{s^n}{\sqrt{2\pi n}} \left(\frac{e}{n} \right)^n. \quad (7)$$

Applying the logarithm,

$$\log_{10} \mathbb{P}(S_n \leq s) \leq n \log_{10} \left(\frac{s \cdot e}{n} \right) - \frac{1}{2} \log_{10}(2\pi n). \quad (8)$$

The second upper-bound is obtained from Hoeffding's inequality [13]. The sum S_n satisfies the condition of the inequality, leading to the following upper-bound

$$\mathbb{P}(S_n \leq s) \leq e^{-2 \frac{(\frac{n}{2} - s)^2}{n}} \quad (9)$$

for $s < \frac{n}{2}$, which is the interesting case. In logarithm,

$$\log_{10} \mathbb{P}(S_n \leq s) \leq -2 \frac{(\frac{n}{2} - s)^2}{n} \log_{10} e. \quad (10)$$

Both upper-bounds are needed to obtain a reasonable good approximation of the p -value term for the whole range $s \in (0, \frac{n}{2})$. Indeed, the first term approximation (Equation (5)) is tight for small values of s (even exact for $s \leq 1$), but it degrades when $s \approx \frac{n}{2}$. On the other hand, the upper-bound obtained from Hoeffding's inequality is quite rough for small values and gets better for $s \approx \frac{n}{2}$. Thus, the p -value approximation is obtained as the minimum of the two approximations, leading to a reasonable approximation for the whole range. All in all,

$$\log_{10} \mathbb{P}(S_n \leq s) \approx \begin{cases} \min \left(n \log_{10} \left(\frac{s \cdot e}{n} \right) - \frac{1}{2} \log_{10}(2\pi n), -2 \frac{(\frac{n}{2} - s)^2}{n} \log_{10} e \right), & s < \frac{n}{2}, \\ 0, & s \geq \frac{n}{2}. \end{cases} \quad (11)$$

This is the approximation used in step 15 of Algorithm 1.

There is a potential pitfall in using this approximation. The NFA is used to decide when a histogram is indeed quantized, but also to select the best quantization factor q when more than one satisfies the detection threshold. In such a case, the value q leading to a smaller NFA is selected.

	1	1	2	2	4	5	6
1	1	1	2	3	6	6	6
1	1	2	2	4	6	7	6
1	2	2	3	5	9	8	6
2	2	4	6	7	11	10	8
2	4	6	6	8	10	11	9
5	6	8	9	10	12	12	10
7	9	10	10	11	10	10	10

	10^{-405}	10^{-360}	10^{-9725}	10^{-7960}	10^{-6098}	10^{-3245}	10^{-1952}
10^{-377}	10^{-337}	10^{-292}	10^{-8020}	10^{-8416}	10^{-3197}	10^{-2151}	10^{-1551}
10^{-322}	10^{-290}	10^{-8307}	10^{-7292}	10^{-5762}	10^{-2587}	10^{-1393}	10^{-1253}
10^{-322}	10^{-8029}	10^{-7462}	10^{-7290}	10^{-3217}	10^{-1257}	10^{-912}	10^{-914}
10^{-7667}	10^{-6814}	10^{-5360}	10^{-2710}	10^{-1744}	10^{-701}	10^{-457}	10^{-416}
10^{-5823}	10^{-4051}	10^{-2376}	10^{-1823}	10^{-998}	10^{-527}	10^{-289}	10^{-305}
10^{-2373}	10^{-1864}	10^{-1177}	10^{-768}	10^{-481}	10^{-269}	10^{-233}	10^{-262}
10^{-1236}	10^{-793}	10^{-573}	10^{-418}	10^{-292}	10^{-262}	10^{-232}	10^{-276}

Table 1: Quantization table and associated NFA values obtained by the proposed method for the image in Figure 3. Notice that the proposed method does not estimate the DC coefficient (upper-left empty position).

This could lead to a wrong selection if two candidates take values where NFA approximation errors are very different. Indeed, the approximation error is not uniform on the whole range of values of n and s . Thus, the factor with larger NFA value but smaller approximation error may be preferred over the real quantization factor if the approximation error of the latter is large enough. This effect could be even worse if the different upper-bounds are used in each case. An alternative would be to select the best quantization factor among the significant ones by comparing $\frac{s}{n}$ instead of the approximated NFA values. Nevertheless, we did not observe this problem so its impact is probably limited.

Table 1 shows the quantization table and associated NFA values obtained by the proposed method applied to the image from Figure 3. In this case, no error was made and 63 elements of the Q-table were estimated. Notice the very small NFA values, implying a very high confidence on the result. We also observe that the NFA values are less significant in the high frequencies; the reason is that these coefficients have fewer samples. Indeed, most values are put to zero as it is illustrated in Figure 2. The NFA values are also less significant for the coefficients with the quantization factor $q = 1$. As one can see in Figure 7, the groups of values around each multiple of q are very close to each other, leading to a less concentrated distribution of the normalized errors, and resulting in a less significant detection.

3.3 Computational Complexity

The first step of the algorithm is obtaining the Y channel from the (R, G, B) colorspace. After that, the DCT of all 8×8 blocks in the Y channel is computed. The number of operations is proportional to the number of blocks in the image in both steps. Then, 63×255 statistical tests are performed, one for each quantization candidate and each DCT coefficient. The computation of the normalized quantization errors and their sum is again proportional to the number of blocks in the image. All in all, the number of operations required is proportional to the number of blocks. Thus, the complexity of the method grows linearly with the number of image pixels.

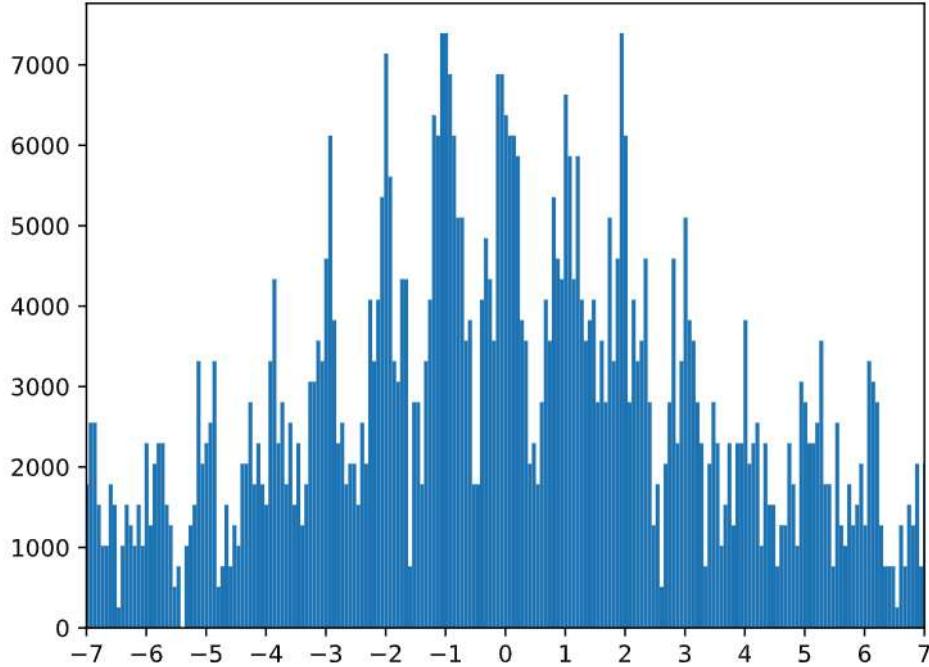


Figure 7: An example of histogram of a DCT coefficient with $q = 1$.

4 Experiments

This section illustrates the strengths and limitations of the proposed method by performing several experiments. For this, an initial uncompressed image was cropped to different sizes, the obtained images were then JPEG compressed using different quality factors. In all cases, the JPEG compression was performed using the ImageMagick package². The estimated Q-table values are displayed in courier font and non-detections are represented by a dash. The actual Q-tables encoded on the JPEG headers were extracted using the DJPEG tool created by the Independent JPEG Group³.

4.1 Uncompressed Images

When the proposed algorithm is applied on uncompressed images, it should produce no detection as the DCT coefficients will show no sign of quantization. The normalized quantization errors should approximately follow the null model H_0 , leading to a NFA larger than 1. The following table shows an example, obtained on the non-compressed image of Figure 3:

²<https://imagemagick.org>

³The command used was: `djpeg -verbose -verbose IMAGE.JPG > /dev/null`

estimated quantization matrix:

```
- - - - - - -  
- - - - - - -  
- - - - - - -  
- - - - - - -  
- - - - - - -  
- - - - - - -  
- - - - - - -  
- - - - - - -  
- - - - - - -
```

associated $\log_{10}(\text{NFA})$ values:

	5.6	5.7	5.7	5.6	5.7	5.7	5.7
5.4	5.5	5.7	5.7	5.8	5.8	5.8	5.7
5.7	5.3	5.7	5.7	5.8	5.7	5.5	5.7
5.5	5.7	5.7	5.6	4.9	5.5	5.5	5.8
5.5	5.4	5.3	5.4	4.5	5.7	5.7	5.7
5.6	5.5	5.5	5.7	5.4	5.7	5.8	5.4
5.6	5.9	5.6	5.8	5.7	5.6	5.7	6.0
5.7	5.8	5.6	5.7	5.7	5.8	5.8	6.0

Indeed, all the NFA values are larger than 1 (the $\log_{10} \text{NFA} > 0$) and no coefficient was detected as quantized.

4.2 Impact of the JPEG Compression Quality

Figures 8 and 9 illustrate the impact of the JPEG compression quality (as determined by the quality factor QF) on the obtained result.

When the JPEG compression is stronger (lower image quality), as shown in Figure 9, some of the high-frequency DCT coefficients are missing. Indeed, more and more DCT are put to zero, due to the strong quantization value. Hence, very few samples are left to perform the statistical test. Recall that the validation step only takes into consideration the non-zero coefficients.

In all cases, the detected elements of the Q-table provided the right value (as verified comparing with the table encoded in the JPEG header). Thus, no false detection was produced. Concerning the missing values, in some cases the Q-table can be completed when a known table is identified with the detected values, as proposed in methods like [10].

4.3 Impact of the Image Size

Figure 10 illustrates the impact of the image size on the result quality. The proposed algorithm was applied to different crop sizes of the same image as before, JPEG compressed with $QF = 93$. One can observe that the larger the image, the more elements of the Q-table are detected by the method. Indeed, the larger the image, the more non-null DCT coefficients and the better the statistical information to validate the a contrario test.

As said before, what is important is the number of non-null DCT coefficients. As a result, the number of elements detected by the method for a given image size may depend on the contents of the image. Figure 11 shows the result obtained on two equal sized crops of the same image. The first one corresponds to a flat region of the image, where most high-frequency DCT coefficients are null or have a small value. On the other hand, the second crop corresponds to a highly textured part of the image, with more high-frequency contents and thus more non-null DCT coefficients. As expected, more elements of the Q-table are detected in the second case.



$QF = 100$

estimated quantization matrix:							
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1



$QF = 99$

estimated quantization matrix:							
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	2	2	1
1	1	1	1	1	2	2	2
1	1	1	1	2	2	2	2
1	1	2	2	2	2	2	2
1	2	2	2	2	2	2	2



$QF = 98$

estimated quantization matrix:							
1	1	1	1	2	2	2	2
1	1	1	1	1	2	2	2
1	1	1	1	2	2	3	2
1	1	1	1	2	3	3	2
1	1	1	2	3	4	4	3
1	1	2	3	3	4	5	4
2	3	3	3	4	5	5	4
3	4	4	4	4	4	4	4



$QF = 95$

estimated quantization matrix:							
1	1	2	2	4	5	6	
1	1	1	2	3	6	6	6
1	1	2	2	4	6	7	6
1	2	2	3	5	9	8	6
2	2	4	6	7	11	10	8
2	4	6	6	8	10	11	9
5	6	8	9	10	12	12	10
7	9	10	10	11	10	10	10

Figure 8: Different quality factor JPEG images and their estimated table by the proposed method. All the estimated values are correct.



$QF = 90$

estimated quantization matrix:							
2	2	3	5	8	10	12	
2	2	3	4	5	12	12	11
3	3	3	5	8	11	14	11
3	3	4	6	10	17	16	12
4	4	7	11	14	22	21	15
5	7	11	13	16	21	23	18
10	13	16	17	21	24	24	20
14	18	19	20	22	20	21	20



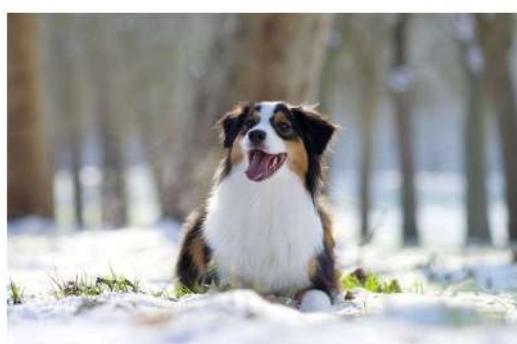
$QF = 80$

estimated quantization matrix:							
4	4	6	10	16	20	24	
5	5	6	8	10	23	24	22
6	5	6	10	16	23	28	22
6	7	9	12	20	35	32	25
7	9	15	22	27	44	-	-
10	14	22	26	32	42	-	-
20	26	31	35	41	-	-	-
29	37	38	-	-	-	-	-



$QF = 70$

estimated quantization matrix:							
7	6	10	14	24	31	37	
7	7	8	11	16	35	36	33
8	8	10	14	24	34	41	34
8	10	13	17	31	52	-	-
11	13	22	34	41	-	-	-
14	21	33	38	-	-	-	-
29	38	47	-	-	-	-	-
43	55	-	-	-	-	-	-



$QF = 60$

estimated quantization matrix:							
9	8	13	19	32	41	49	
10	10	11	15	21	46	48	-
11	10	13	19	32	46	55	45
11	14	18	23	41	-	-	-
14	18	30	45	54	-	-	-
19	28	44	51	-	-	-	-
39	51	62	-	-	-	-	-
58	-	-	-	-	-	-	-

Figure 9: Different quality factor JPEG images and their estimated table by the proposed method. Dashes (-) indicate non-detected values (no candidate quantization value was validated as the NFA was bigger than 1). All the estimated values are correct.

size 3000×2000

estimated quantization matrix:								
2	1	2	3	6	7	9		
2	2	2	3	4	8	8	8	
2	2	2	3	6	8	10	8	
2	2	3	4	7	12	11	9	
3	3	5	8	10	15	14	11	
3	5	8	9	11	15	16	13	
7	9	11	12	14	17	17	14	
10	13	13	14	16	14	14	14	

size 1500×1000

estimated quantization matrix:								
2	-	2	3	6	7	9		
2	2	2	3	4	8	8	8	
2	2	2	3	6	8	10	8	
2	2	3	4	7	12	11	9	
3	3	5	8	10	15	14	11	
3	5	8	9	11	15	16	13	
7	9	11	12	14	17	17	14	
10	13	13	14	16	14	14	14	

size 750×500

estimated quantization matrix:								
2	-	2	3	6	-	-	-	-
2	2	2	3	4	-	-	-	-
2	2	2	3	6	-	-	-	-
2	2	3	4	-	-	-	-	-
3	3	5	-	-	-	-	-	-
3	5	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-

size 75×50

estimated quantization matrix:								
2	-	2	-	-	-	-	-	-
2	2	2	3	-	-	-	-	-
2	-	2	3	-	-	-	-	-
2	-	-	-	-	-	-	-	-
3	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-

Figure 10: Different crops of the same JPEG-compressed image ($QF = 93$) and their estimated table by the proposed method. The estimated values are correct and non validated detections (with a NFA bigger than 1) are set to -.



estimated quantization matrix:							
2	-	2	-	-	-	-	-
2	2	2	3	-	-	-	-
2	-	2	3	-	-	-	-
2	-	-	-	-	-	-	-
3	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-

size 75×50 

estimated quantization matrix:							
2	-	2	3	6	7	-	-
2	2	2	3	4	8	8	-
2	2	2	3	6	8	10	8
2	2	3	4	7	12	11	9
3	3	5	8	10	-	-	-
3	5	8	9	-	-	-	-
7	9	11	12	-	-	-	-
10	13	-	-	-	-	-	-

size 75×50

Figure 11: Crop in a flat area and in a textured area of size 75×50 of the JPEG-compressed image ($QF = 93$) and its estimated table by the proposed method. The estimated values are correct and non validated detections (with a NFA bigger than 1) are set to $-$.

4.4 Aligned Double JPEG Compression

When an image has gone through several compressions, only the last Q-table is in the header. However, the DCT coefficients have gone through more than one quantization. Figure 12 shows an image first compressed at quality $QF_1 = 90$, then $QF_2 = 98$; the figure also shows the quantization table for the Y channel stored in the JPEG header, and the estimated quantization table.



Image

1	1	1	1	1	2	2	2
1	1	1	1	1	2	2	2
1	1	1	1	2	2	3	2
1	1	1	1	2	3	3	2
1	1	1	2	3	4	4	3
1	1	2	3	3	4	5	4
2	3	3	3	4	5	5	4
3	4	4	4	4	4	4	4

Table in JPEG header

	2	2	3	5	8	10	12
2	2	3	4	5	12	12	11
3	3	3	5	8	11	15	11
3	3	4	6	10	18	15	12
4	4	7	11	15	4	20	15
5	7	11	12	15	20	5	4
10	12	15	18	20	25	25	20
15	4	20	20	4	20	20	20

Estimated table

Figure 12: Double compressed image, its quantization table for the Y channel stored in the header, and the estimated quantization table. Most of the estimated values correspond to the $QF = 90$ table; but the coefficients highlighted in red show a ± 1 error, and the coefficients highlighted in blue show the value of the $QF = 98$ table.

The estimated table is very different from the one in the header and is also not the same as the one for $QF = 90$, which can be seen in Figure 9. Most of the estimated coefficients correspond to $QF = 90$, which imposes a stronger quantization. Nevertheless, five of the estimated coefficients

correspond to the $QF = 98$ table (highlighted in blue), and 17 coefficients show a ± 1 error relative to the actual value in the $QF = 90$ table (highlighted in red).

4.5 Unknown Grid Origin

By default, the proposed algorithm assumes a standard JPEG grid with origin at $(0, 0)$. This is not an important assumption, as there are reliable methods to estimate the grid origin when JPEG traces are present [17, 18, 19]. Nevertheless, when the proposed algorithm is applied on an image having its grid origin different from $(0, 0)$, there should be no detection. For instance, this can happen when the image has been cropped after its compression. This is mainly what happens except in some cases when one of the two coordinates (horizontal or vertical) is aligned with the correct grid.

To illustrate this behavior, the same test image used before was JPEG compressed with $QF = 90$ and then cropped so as to obtain different effective grid origins. All the 64 possible grid origins were generated and Figures 13, 14 and 15 show six particularly interesting cases; no detection was made on the remaining cases. As expected, the quantization table is correctly detected when the grid origin is $(0, 0)$. Also, no detection is performed when the grid origin is $(4, 4)$, completely out of phase with the tested grid $(0, 0)$. Nevertheless, there are some detections when there is a grid shift of just ± 1 in the vertical or horizontal direction, i.e., for the grids $(0, 1)$ or $(0, 7)$ (Figure 14), or for grid $(1, 0)$ or $(7, 0)$ (Figure 15).

The DCT is a separable transform, so it can be interpreted as the superposition of an horizontal and a vertical 1D DCT transforms. When translating the grid one position horizontally, the vertical components remain almost the same. For this reason, in some cases the coefficient quantization is still detectable after a one pixel horizontal or vertical shift of the grid origin. Notice that the detected quantization values correspond in most cases to the same quantization values as in the $QF = 90$ table, with an error of ± 1 . Also, in all those cases, the detections found on the wrong grid origin are clearly less meaningful (with a larger NFA) compared to the correct grid alignment. These shifted grid detections can only be observed on large images; on small images, with fewer samples, with the degraded NFA, they will not lead to meaningful detections. Thus, it should be possible to identify the right grid origin by selecting the grid with more meaningful detections. This would work, however, under the assumption that a single JPEG compression was performed. The next section shows an experiment with double JPEG compression at different grid positions.

4.6 Non-Aligned Double JPEG Compression

Figures 16, 17 and 18 show all the detected quantization factors among all the possible grid origins for a non-aligned doubly compressed image. Indeed, the image was first compressed at quality $QF_1 = 90$, then cropped by removing 4 lines and 4 columns. The image was then compressed at quality $QF_2 = 98$. As expected, the proposed algorithm detects correctly both quantization tables when the analysis is performed on the right grid origins, see Figure 16. But as explained in the previous section, the algorithm also produces detection on the grids with ± 1 horizontal or vertical shifts. Figure 17 shows the spurious detection around the grid $(4, 4)$ and Figure 18 shows the spurious detections around the grid $(0, 0)$.

This example shows that selecting the correct grid origin is not trivial when the aim is also to detect multiple possible JPEG compressions. More work is required to evaluate the possibility of solving this combined problem.

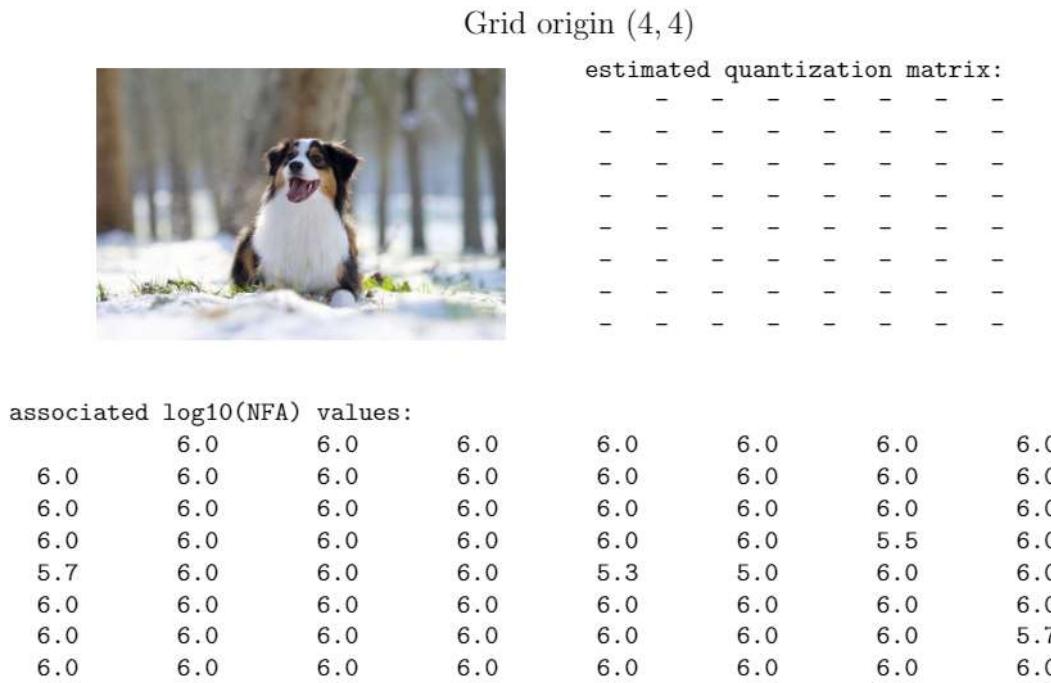
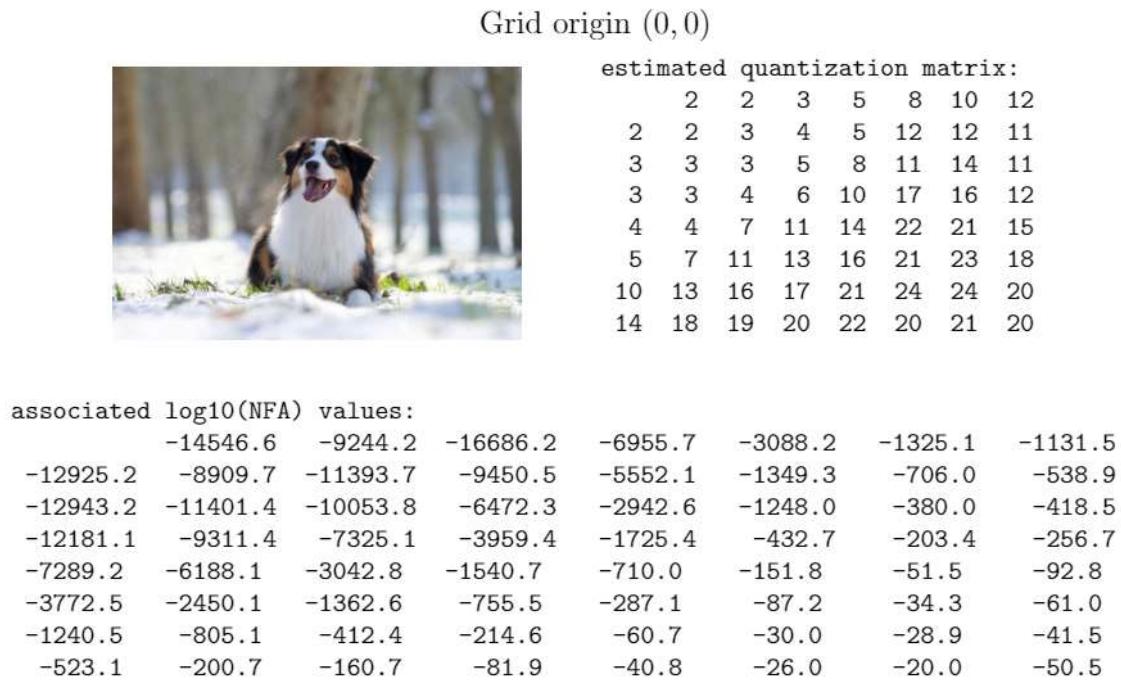


Figure 13: Different crops of the JPEG image with quality 90 and their estimated table by the proposed method. Dashes (–) indicate non-detected values (no candidate quantization value was validated as the NFA was bigger than 1). All the estimated values are correct.

Grid origin (0, 1)							
estimated quantization matrix:							
-	2	2	3	5	7	9	11
-	-	-	3	4	-	2	2
-	-	-	-	-	-	-	2
-	-	-	-	-	-	-	2
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	19	19	21	19	-	19

associated log10(NFA) values:							
4.2	-42.8	-27.8	-622.5	-111.9	-215.7	-99.1	-206.4
6.0	6.0	6.0	-11.4	-18.9	6.0	-2.0	-19.8
6.0	6.0	6.0	6.0	6.0	6.0	5.7	-26.9
6.0	6.0	6.0	6.0	6.0	6.0	6.0	-42.2
6.0	6.0	6.0	6.0	6.0	6.0	6.0	3.0
6.0	6.0	6.0	6.0	5.1	6.0	5.6	5.3
6.0	6.0	6.0	6.0	6.0	6.0	5.3	5.9
1.2	4.2	-10.0	-5.6	-2.8	-0.3	1.4	-5.5

Grid origin (0, 7)							
estimated quantization matrix:							
-	2	2	3	5	7	9	11
-	-	-	3	4	-	2	2
-	-	-	-	-	-	-	2
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	19	21	19	20	19

associated log10(NFA) values:							
5.0	-56.6	-20.6	-635.2	-110.9	-240.6	-115.5	-202.0
6.0	6.0	6.0	-14.9	-23.7	5.8	-1.5	-18.7
6.0	6.0	6.0	6.0	6.0	6.0	5.8	-30.9
6.0	6.0	6.0	6.0	6.0	6.0	6.0	-41.1
6.0	6.0	5.6	5.6	6.0	5.9	6.0	0.7
6.0	5.6	6.0	6.0	5.9	5.9	5.1	5.8
6.0	6.0	6.0	6.0	6.0	6.0	5.6	6.0
5.2	4.5	3.5	-5.9	-5.3	-4.9	-1.2	-4.7

Figure 14: Different crops of the JPEG image with quality 90 and their estimated table by the proposed method. Dashes (–) indicate non-detected values (no candidate quantization value was validated as the NFA was bigger than 1). All the estimated values are incorrect.

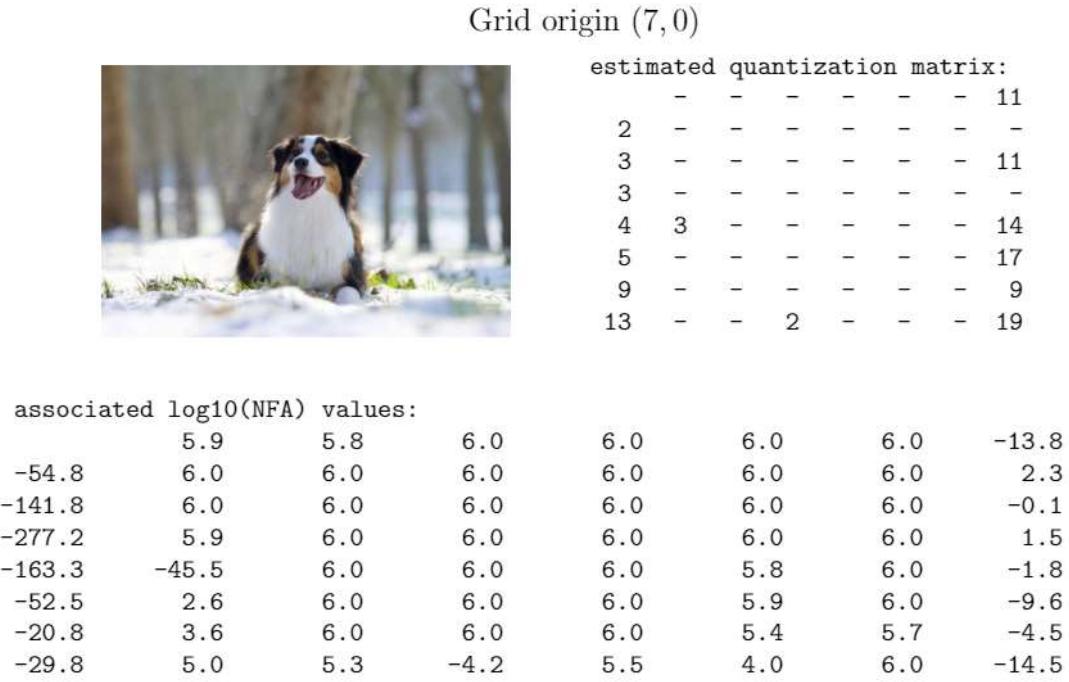
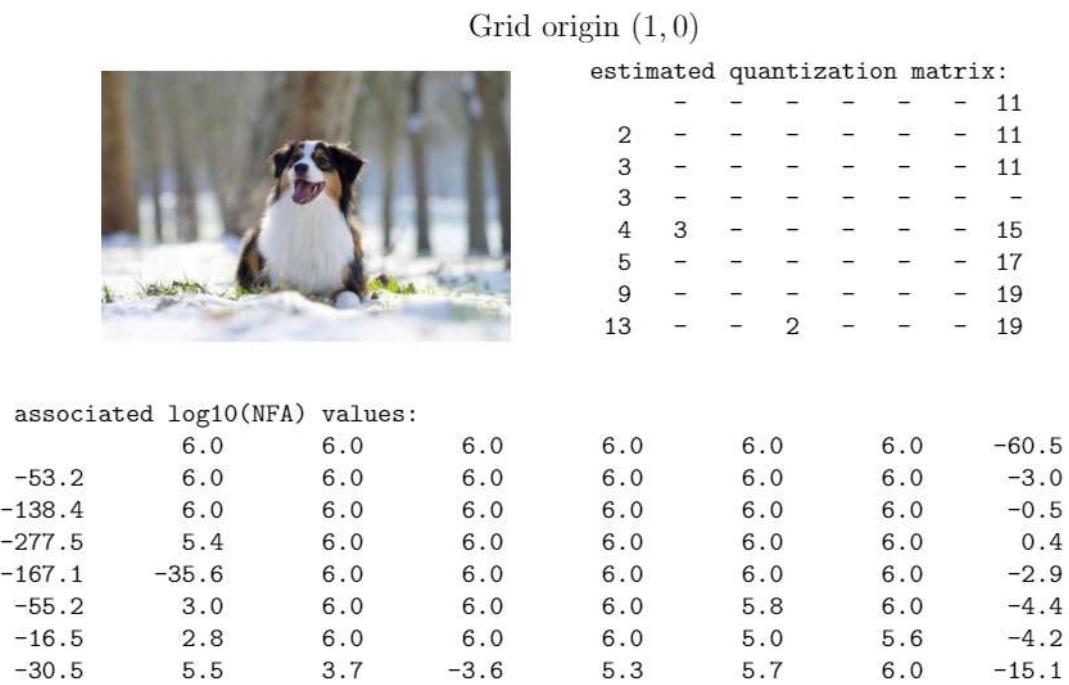


Figure 15: Different crops of the JPEG image with quality 90 and their estimated table by the proposed method. Dashes (–) indicate non-detected values (no candidate quantization value was validated as the NFA was bigger than 1). All the estimated values are incorrect.

Grid origin (0, 0)



estimated quantization matrix:

1	1	1	1	2	2	2
1	1	1	1	1	2	2
1	1	1	1	2	2	3
1	1	1	1	2	3	3
1	1	1	2	3	4	4
1	1	2	3	3	4	5
2	3	3	3	4	5	5
3	4	4	4	4	4	4

associated log10(NFA) values:

-557.4	-478.1	-421.7	-327.3	-7358.6	-1949.0	-3950.1
-537.4	-468.9	-434.2	-379.3	-268.2	-5240.1	-1696.7
-474.5	-420.4	-330.6	-328.7	-3178.0	-3851.0	-1328.9
-425.5	-382.6	-318.5	-305.7	-3270.6	-3452.3	-1337.2
-359.3	-282.8	-142.6	-3433.8	-1679.5	-1721.3	-557.5
-337.4	-288.2	-4033.6	-3112.5	-1829.6	-1586.9	-613.7
-1535.3	-1774.6	-1268.5	-1161.4	-547.2	-591.6	-228.7
-2951.1	-2110.2	-1838.3	-1566.0	-1092.2	-1040.1	-475.6
						-731.5

Grid origin (4, 4)



estimated quantization matrix:

2	2	3	5	8	10	12
2	2	3	4	5	12	12
3	3	3	5	8	11	14
3	3	4	6	10	17	16
4	4	7	11	14	22	21
5	7	11	13	16	21	23
10	13	16	17	21	24	23
14	18	19	20	22	20	21
						21

associated log10(NFA) values:

-821.4	-409.2	-4115.8	-3221.3	-1867.8	-729.1	-675.0
-791.1	-364.8	-2839.6	-3210.6	-2128.6	-889.0	-412.7
-4030.9	-2933.1	-2289.0	-2461.3	-1467.9	-756.5	-220.7
-3727.8	-1832.7	-2243.8	-1561.6	-903.2	-277.2	-105.8
-2957.3	-1550.1	-1376.1	-805.6	-361.5	-95.5	-25.6
-967.6	-997.6	-740.6	-389.5	-159.9	-54.0	-20.4
-533.1	-409.4	-226.6	-105.6	-32.0	-14.1	-15.3
-301.1	-124.3	-94.4	-44.9	-18.9	-15.1	-9.4
						-27.5

Figure 16: Different crops of the JPEG image with quality 90 followed by a crop and a second compression with quality 98 and their estimated table by the proposed method. Dashes (—) indicate non-detected values (no candidate quantization value was validated as the NFA was bigger than 1). All the estimated values are correct for the top estimation and some coefficients are with an error of ± 1 in the bottom one.

Grid origin (3, 4)



estimated quantization matrix:						
-	-	-	-	-	-	11
2	-	-	-	-	-	-
3	-	-	-	-	-	-
3	-	-	-	-	-	-
4	-	-	-	-	-	-
4	-	-	-	-	-	17
9	-	-	-	-	-	19
13	-	-	-	-	-	20

Grid origin (5, 4)



estimated quantization matrix:						
-	-	-	-	-	-	11
2	-	-	-	-	-	-
3	-	-	-	-	-	-
3	-	-	-	-	-	-
4	-	-	-	-	-	-
4	-	-	-	-	-	18
10	-	-	-	-	-	-
13	-	-	-	-	-	20

Grid origin (4, 3)



estimated quantization matrix:						
2	-	3	5	7	9	11
-	-	-	4	-	-	-
-	-	-	-	-	-	-
-	-	-	-	-	-	-
-	-	-	-	-	-	-
-	-	-	-	-	-	-
-	-	-	-	-	-	-
-	-	-	19	23	20	21
-	-	-	19	23	20	21
-	-	-	19	23	20	19

Grid origin (4, 5)



estimated quantization matrix:						
2	-	3	5	8	9	11
-	-	-	4	-	-	-
-	-	-	-	-	-	-
-	-	-	-	-	-	-
-	-	-	-	-	-	-
-	-	-	-	-	-	-
-	-	-	-	-	-	-
-	-	-	19	19	21	-
-	-	-	19	19	21	-
-	-	-	19	19	21	-

Figure 17: Different crops of the JPEG image with quality 90 followed by a crop and a second compression with quality 98 and their estimated table by the proposed method. Dashes (-) indicate non-detected values (no candidate quantization value was validated as the NFA was bigger than 1). All the estimated values are incorrect.

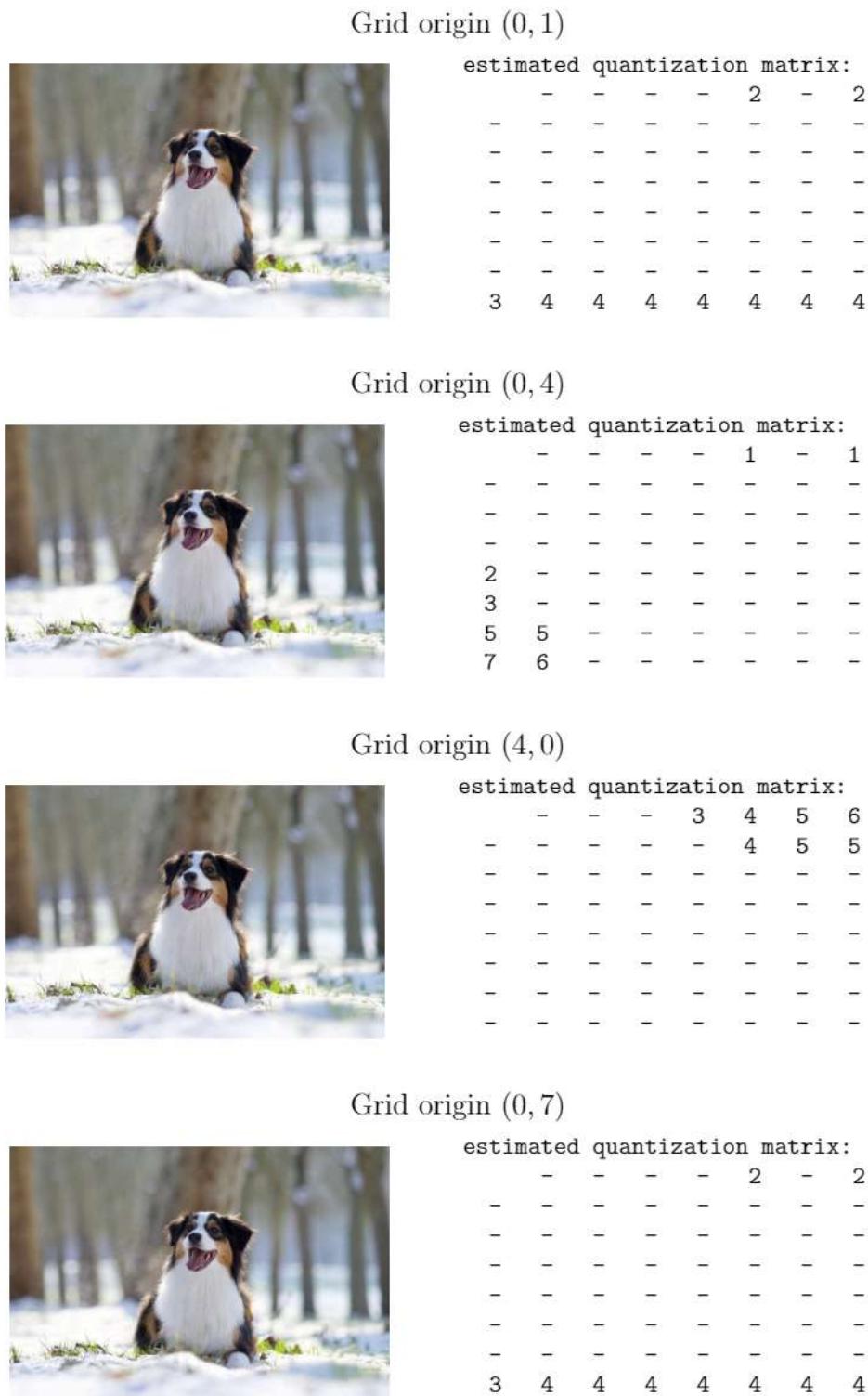


Figure 18: Different crops of the JPEG image with quality 90 followed by a crop and a second compression with quality 98 and their estimated table by the proposed method. Dashes (-) indicate non-detected values (no candidate quantization value was validated as the NFA was bigger than 1). All the estimated values are incorrect.

5 Conclusion

A reliable JPEG quantization table estimation algorithm based on the a contrario theory was described. The method uses only information from the image itself and does not require any data from the file header. The statistical validation step secures the detection, leading to a very small number of false detections. In addition, the method has a linear computational complexity with the size of the image. Future work will focus on extending the method to estimate the Q-table of the chroma channels.

Image Credits



image from Thibault Bordier.

References

- [1] *ITU-T recommendation T.81: Information technology digital compression and coding of continuous-tone still images requirements and guidelines*, tech. report, International Telecommunication Union, 1992.
- [2] A. DAVY, T. EHRET, J-M. MOREL, AND M. DELBRACIO, *Reducing anomaly detection in images to detection in noise*, in IEEE International Conference on Image Processing (ICIP), IEEE, 2018, pp. 1058–1062. <https://doi.org/10.1109/ICIP.2018.8451059>.
- [3] M. DELBRACIO, D. KELLY, M.S. BROWN, AND P. MILANFAR, *Mobile computational photography: A tour*, Annual Review of Vision Science, 7 (2021), pp. 571–604. <https://doi.org/10.1146/annurev-vision-093019-115521>.
- [4] A. DESOLNEUX, L. MOISAN, AND J.-M. MOREL, *From Gestalt Theory to Image Analysis*, Springer, 2008. ISBN 978-0-387-74378-3.
- [5] Z. FAN AND R.L. DE QUEIROZ, *Maximum likelihood estimation of JPEG quantization table in the identification of bitmap compression history*, International Conference on Image Processing (ICIP), (2000), pp. 948–951. <https://doi.org/10.1109/ICIP.2000.901117>.
- [6] ——, *Identification of bitmap compression history: JPEG detection and quantizer estimation*, IEEE Transactions on Image Processing, 12 (2003). <https://doi.org/10.1109/TIP.2002.807361>.
- [7] H. FARID, *Digital image ballistics from JPEG quantization: A followup study*, Department of Computer Science, Dartmouth College, Tech. Rep. TR2008-638, 7 (2008), pp. 1–28.
- [8] ——, *Photo Forensics*, The MIT Press, 2016. ISBN 978-0262035347.
- [9] J. FRIDRICH, M. GOLJAN, AND R. DU, *Steganalysis based on JPEG compatibility*, in Multimedia Systems and Applications IV, vol. 4518, International Society for Optics and Photonics, 2001, pp. 275–281. <https://doi.org/10.1117/12.448213>.
- [10] D. FU, Y.Q. SHI, AND W. SU, *A generalized Benford’s law for JPEG coefficients and its applications in image forensics*, in Security, Steganography, and Watermarking of Multimedia Contents IX, vol. 6505, International Society for Optics and Photonics, 2007, p. 65051L. <http://dx.doi.org/10.1117/12.704723>.

- [11] A. GORDON, G. GLAZKO, X. QIU, AND A. YAKOVLEV, *Control of the Mean Number of False Discoveries, Bonferroni and Stability of Multiple Testing*, Annals of Applied Statistics, 1 (2007), pp. 179–190. <https://doi.org/10.1214/07-AOAS102>.
- [12] R. GROMPONE VON GIOI, J. JAKUBOWICZ, J-M. MOREL, AND G. RANDALL, *LSD: A fast line segment detector with a false detection control*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 32 (2010), pp. 722–732. <https://doi.org/10.1109/TPAMI.2008.300>.
- [13] W. HOEFFDING, *Probability inequalities for sums of bounded random variables*, Journal of the American Statistical Association, 58 (1963), pp. 13–30. <https://doi.org/10.1080/01621459.1963.10500830>.
- [14] S. KOTZ, N. BALAKRISHNAN, AND N.L. JOHNSON, *Continuous multivariate distributions, Volume 1: Models and applications*, vol. 1, John Wiley & Sons, 2004. ISBN 978-0-471-18387-7.
- [15] J. LEZAMA, R. GROMPONE VON GIOI, G. RANDALL, AND J-M. MOREL, *Finding vanishing points via point alignments in image primal and dual domains*, in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 509–515. <https://doi.org/10.1109/CVPR.2014.72>.
- [16] D.G LOWE, *Visual recognition from spatial correspondence and perceptual organization*, in International Joint Conference on Artificial Intelligence, 1985, pp. 953–959.
- [17] T. NIKOUKHAH, J. ANGER, M. COLOM, J-M. MOREL, AND R. GROMPONE VON GIOI, *ZERO: a Local JPEG Grid Origin Detector Based on the Number of DCT Zeros and its Applications in Image Forensics*, Image Processing On Line, 11 (2021), pp. 396–433. <https://doi.org/10.5201/ipol.2021.390>.
- [18] T. NIKOUKHAH, J. ANGER, T. EHRET, M. COLOM, J-M. MOREL, AND R. GROMPONE VON GIOI, *JPEG Grid Detection based on the Number of DCT Zeros and its Application to Automatic and Localized Forgery Detection*, in IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2019.
- [19] T. NIKOUKHAH, R. GROMPONE VON GIOI, M. COLOM, AND J-M. MOREL, *Automatic JPEG grid detection with controlled false alarms, and its image forensic applications*, in IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), 2018, pp. 378–382. <http://dx.doi.org/10.1109/MIPR.2018.00083>.
- [20] T.H. THAI, R. COGRANNE, F. RETRAINT, AND T-N-C. DOAN, *JPEG quantization step estimation and its applications to digital image forensics*, IEEE Transactions on Information Forensics and Security, 12 (2017), pp. 123–133. <https://doi.org/10.1109/TIFS.2016.2604208>.
- [21] G.K. WALLACE, *The JPEG still picture compression standard*, IEEE Transactions on Consumer Electronics, (1991). <https://doi.org/10.1109/30.125072>.
- [22] A.P. WITKIN AND J.M. TENENBAUM, *On the role of structure in vision*, in Human and Machine Vision, Elsevier, 1983, pp. 481–543.
- [23] S. YE, Q. SUN, AND E-C. CHANG, *Detecting digital image forgeries by measuring inconsistencies of blocking artifact*, in IEEE International Conference on Multimedia and Expo, IEEE, 2007, pp. 12–15. <https://doi.org/10.1109/ICME.2007.4284574>.