# Revisiting Perturbed Quantization

Jan Butora, Jessica Fridrich

# Revisiting Perturbed Quantization

Jan Butora and Jessica Fridrich
Binghamton University
Department of Electrical and Computer Engineering
Binghamton, NY 13850
{jbutora1,fridrich}@binghamton.edu

## ABSTRACT

In this work, we revisit Perturbed Quantization steganography with modern tools available to the steganographer today, including near-optimal ternary coding and content-adaptive embedding with side-information. In PQ, side-information in the form of rounding errors is manufactured by recompressing a JPEG image with a judiciously selected quality factor. This side-information, however, cannot be used in the same fashion as in conventional side-informed schemes nowadays as this leads to highly detectable embedding. As a remedy, we utilize the steganographic Fisher information to allocate the payload among DCT modes. In particular, we show that the embedding should not be constrained to contributing coefficients only as in the original PQ but should be expanded to the so-called "contributing DCT modes." This approach is extended to color images by slightly modifying the SI-UNIWARD algorithm. Using the best detectors currently available, it is shown that by manufacturing side information with double compression, one can embed the same amount of information into the doubly-compressed cover image with a significantly better security than applying J-UNIWARD directly in the single-compressed image. At the end of the paper, we show that double compression with the same quality makes side-informed steganography extremely detectable and should be avoided.

## CCS CONCEPTS

• **Security and privacy**; • **Computing methodologies** → **Image compression**;

## KEYWORDS

Steganography, double compression, perturbed quantization, side information, contributing mode

**ACM Reference Format:**
Jan Butora and Jessica Fridrich. 2021. Revisiting Perturbed Quantization. In *Proceedings of the 2021 ACM Workshop on Information Hiding and Multimedia Security (IH&MMSec '21), June 22–25, 2021, Virtual Event, Belgium.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3437880.3460396


*IH&MMSec '21, June 22–25, 2021, Virtual Event, Belgium*

ACM ISBN 978-1-4503-8295-3/21/06…$15.00
https://doi.org/10.1145/3437880.3460396

## 1 INTRODUCTION

Side-informed steganographic schemes are among the most secure steganographic schemes in existence today. The side-information typically comes in the form of rounding errors after some information-reducing processing applied to the (pre)cover image. One such processing is JPEG compression, which is known to provide high levels of security [6, 19, 22, 23, 25, 26, 29–32]. The biggest drawback is that the steganographer needs to have access to the uncompressed image, considering that most imaging devices output images that are already compressed. The embedding method known as Perturbed Quantization (PQ) [23] manufactures side-information by recompressing the JPEG cover image in a way that maximizes the number of coefficients that fall in the middle of the quantization intervals during the second compression, and which are used for embedding. In this paper, we revisit this approach in light of modern tools presently available to the steganographer, such as content-adaptive embedding with costs modulated by the rounding errors [16, 27, 29] implemented using Syndrome Trellis Codes (STCs) [21] rather than the suboptimal wet paper codes [24] used in PQ. Additionally, due to the recent increased interest in embedding into color [1, 2, 14, 16, 41, 44], we extend the embedding to color JPEGs.

We do so while benchmarking the security with rich models [18, 28, 34, 40] and current state-of-the-art convolutional neural networks (CNNs) [5, 42, 43].

In Section 2, we introduce notation and describe the side-information produced by double compression. Section 3 explains the datasets and detectors used for evaluating the proposed method. In Section 4, we derive a rule for selecting the second compression quality that provides, in some sense, the best side-information possible. The original PQ embedding is then modified to be able to embed larger payloads in images compressed with high qualities as well as in color images. Section 5 shows the experimental results on grayscale and color images. In Section 6, we delve into why double compression with the same quality should not be used as a source of side-information. The paper is concluded in Section 7.

## 2 PRELIMINARIES AND NOTATION

Boldface symbols are reserved for matrices and vectors with elementwise multiplication and division denoted $\odot$ and $\oslash$. Rounding $x$ to the closest integer is denoted $[x]$. The set of all integers will be denoted $\mathbb{Z}$. For better readability, we strictly use $i, j$ to index pixels and $k, l$ DCT coefficients. Denoting by $x_{ij}, 0 \le i, j \le 7$, an 8×8 block of pixels, they are transformed during JPEG compression to DCT coefficients $d_{kl} = \mathrm{DCT}_{kl}(\mathbf{x}) \triangleq \sum_{i,j=0}^{7} f_{kl}^{ij} x_{ij}$, $0 \le k, l \le 7$, and then quantized $c_{kl} = [d_{kl}/q_{kl}]$, $c_{kl} \in \{-1024, \dots, 1023\}$, where $q_{kl}$ are quantization steps in a luminance quantization matrix, and

$f_{kl}^{ij} = w_k w_l/4 \cos \pi k(2i+1)/16 \cos \pi l(2j+1)/16$, $w_0 = 1/\sqrt{2}$, $w_k = 1$, $0 < k \leq 7$, are the discrete cosines.

During decompression, the above steps are reversed. For a block of quantized DCTs $c_{kl}$, the corresponding block of non-rounded pixels after decompression is $y_{ij} = \text{DCT}_{ij}^{-1}(\mathbf{c} \odot \mathbf{q}) \triangleq \sum_{k,l=0}^{7} f_{kl}^{ij} q_{kl} c_{kl}$, $y_{ij} \in \mathbb{R}$. To obtain the final decompressed image, $y_{ij}$ are rounded to integers $x_{ij} = [y_{ij}]$ and clipped to $[0, 255]$.

For compression of color images, the *RGB* representation is typically changed to $YC_bC_r$ (luminance, and two chrominance signals) with:

$$\begin{aligned} Y &= 0.299R + 0.587G + 0.114B \\ C_b &= 128 - 0.169R - 0.331G + 0.5B \\ C_r &= 128 + 0.5R - 0.419G - 0.081B \end{aligned} \quad (1)$$

The luminance channel $Y$ is processed as described above, while the chrominance signals are optionally subsampled, then transformed using DCT, and finally quantized with chrominance quantization matrices [39]. In this work we avoid subsampling of chrominance signals because its effect on steganography has not been thoroughly studied yet.

## 2.1 Double Compression and Side Information

This work deals with embedding in JPEG images recompressed with a potentially different quality. The abbreviation SC will stand for single compressed and DC for double compressed images. To distinguish between DCT blocks and pixels of SC and DC images, we will use a superscript to keep track of the number of compressions. The symbol $\mathbf{c}^{(1)}$ represents the DCT block after the first compression, while $\mathbf{c}^{(2)}$ is the DCT block after the second compression. Similarly, $\mathbf{q}^{(1)}$ and $\mathbf{q}^{(2)}$ stand for quantization matrices in the first and second compression, respectively.

To obtain a DC image, a DCT block from the SC image, $\mathbf{c}^{(1)}$, is decompressed into $\mathbf{y}^{(1)} = \text{DCT}(\mathbf{c}^{(1)} \odot \mathbf{q}^{(1)})$, and rounded to integers $\mathbf{x}^{(1)} = [\mathbf{y}^{(1)}]$. We then compress with the second quantization table to obtain the DCT coefficients before quantization $\mathbf{d}^{(2)} = \text{DCT}(\mathbf{x}^{(1)})$. The final DCT coefficients after quantization are $\mathbf{c}^{(2)} = [\tilde{\mathbf{c}}^{(2)}] = [\mathbf{d}^{(2)} \oslash \mathbf{q}^{(2)}]$, where $\tilde{\mathbf{c}}^{(2)}$ are the quantized DCT coefficients before rounding to integers. Finally, the side-information created by recompression are the rounding errors during the last quantization $\mathbf{e} = \tilde{\mathbf{c}}^{(2)} - \mathbf{c}^{(2)}$.

To utilize these rounding errors for embedding, we follow the idea in [19] where the (symmetric) embedding costs $\rho_{kl}$ of changing a DCT coefficient $c_{kl}^{(2)}$ by $+1$ or $-1$ are modulated by the rounding errors:

$$\begin{aligned} \rho_{kl}(\text{sign}(e_{kl})) &= (1 - |2e_{kl}|)\rho_{kl} \\ \rho_{kl}(-\text{sign}(e_{kl})) &= \rho_{kl}. \end{aligned} \quad (2)$$

# 3 EXPERIMENTAL SETUP

This section describes the datasets as well as the detectors used for evaluating security.

## 3.1 Datasets

We work with two datasets to cover both grayscale and color images. The first dataset is a union of the popular BOSSbase 1.01 [3] and BOWS2 [4], each containing 10,000 grayscale images downsampled to $256 \times 256$ using *imresize* with default parameters in Matlab. This union was then randomly split into training, validation, and testing sets with 14,000, 1,000, and 5,000 images, respectively. This dataset was JPEG compressed with Matlab's *imwrite* with several quality factors $Q_1$. The second dataset is ALASKA 2 [15] consisting of three qualities 75, 90, 95, each having 25,000 color images of size $512 \times 512$. This dataset was recently used in ALASKA II Kaggle competition.[1]

The compressed images represent the SC cover images (precovers) in our experiments. To obtain the DC cover images, the SC images are loaded into the *RGB* representation with Matlab's *imread*, converted into the $YC_bC_r$ space via (1) (grayscale images are already loaded as $Y$ channel), rounded to integers, and further compressed with quality $Q_2$ 'manually' using Matlab's *dct2*. This was done in order to obtain the rounding errors $\mathbf{e}$ for the subsequent side-informed embedding. The resulting DCT coefficients were finally rounded to the nearest integers to obtain the DC cover images. As mentioned previously, we never used chrominance subsampling during compression of color images. This development pipeline is visualized in Figure 1.

We use the steganographic algorithm J-UNIWARD [29] for SC images as it is still one of the most secure algorithms for the JPEG domain in grayscale and color images when the development pipeline is not available [12, 16, 45]. For DC images, we use the side-informed version SI-UNIWARD [19] with several modifications, specific to DC images, as explained in the next section.

All experiments are set up in such a way that we always embed the same absolute payload size (in bits) in the SC image as in the DC image in order to answer the main question of this paper: "Can we embed the same amount of information more securely by recompressing the cover image?" The payload size will be expressed in bits per non-zero AC DCT coefficients (bpnzac) of SC cover image. All embedding algorithms are simulated on their corresponding rate–distortion bound (e. g., assuming optimal coding).

## 3.2 Detectors

Inspired by the fact that the best detectors in the recent ALASKA II Kaggle competition were mostly from the EfficientNet family, we attempted to train EfficientNet-B0 and B2 [37] on color images. However, these networks would not converge on the proposed DC steganographic scheme even after trying several different training schedules. Thus, in our experiments we used the SRNet [5] and rich models.

Training the SRNet from scratch, however, was also impossible on the payloads used in this paper. There are many possible ways how to alleviate problems with convergence of a CNN detector. One can for example train on larger payloads first and use transfer learning on smaller payloads [7, 38, 47]. Alternatively, one can train on an 'easier' JPEG quality [8] or train on steganography in SC images first. To avoid confusion with so many different possibilities, we selected JIN[2] pretraining [11], which consists of pretraining

[1] https://www.kaggle.com/c/alaska2-image-steganalysis
[2] JIN stands for **J**-UNIWARD embedded **I**mage**N**et

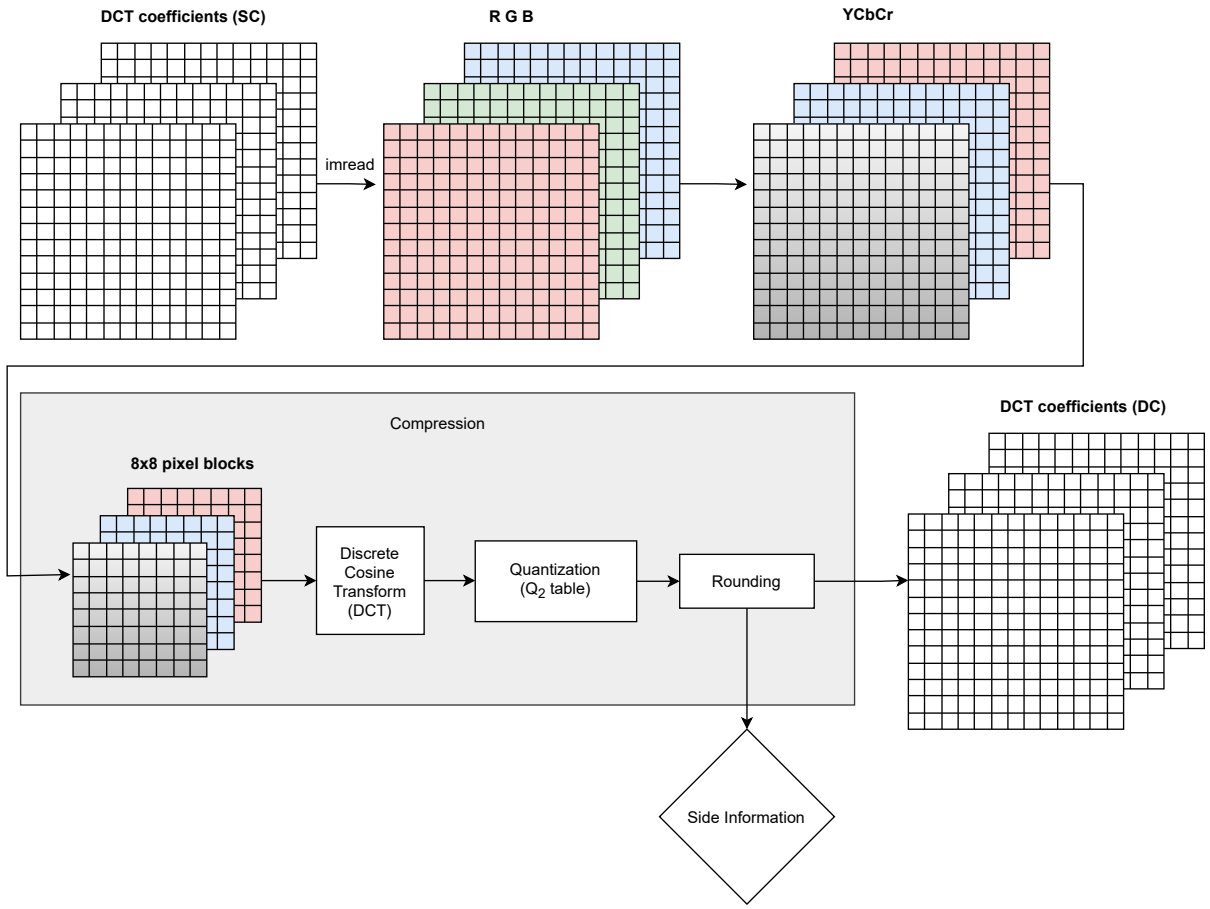


**Figure 1: Double compression pipeline. We start with DCT coefficients of a single compressed (SC) image and end up with DCTs of a double compressed (DC) image.**

on the ImageNet database [20] embedded with J-UNIWARD with uniform random payload between 0.4 and 0.6 bpnzac. This kind of pretraining is suitable for detecting steganography across a variety of embedding schemes embedding both in the JPEG and spatial domain, and for side-informed schemes [11]. All networks used for evaluation in this paper, for SC as well as DC images, were pretrained in this way. Since JIN pretraining is executed on color images, the networks pretrained in this way expect three-channel inputs. Thus, for grayscale images we simply replicated the grayscale representation in all three *RGB* channels. The network detectors were trained for 100 epochs in total on both datasets using mixed precision training with 64 images in every mini-batch, AdaMax optimizer, and weight decay $2\times10^{-4}$. We used OneCycle learning rate (LR) scheduler with maximum LR $10^{-3}$ at epoch 5, division factor 25 and final division factor 10. For easy implementation, PyTorch Lightning[3] framework was used for training our model. For DC images in BOSSbase+BOWS2 database embedded with 0.4 bpnzac, the pair constraint (PC) – forcing cover and its stego version in the same minibatch – was used for the first 50 epochs, otherwise the network would not converge even with the JIN pretraining. For every lower payload (in both datasets), transfer learning from 0.4 bpnzac was used without the PC for 50 epochs only.

For the rich models, we selected the ccJRM [33] and DCTR [28] feature sets with the ensemble classifier [35]. In color images, we use the JRM [34] instead of the cartesian-calibrated [33] ccJRM in order to keep a "manageable dimensionality" – the concatenation of extracted features from all three channels would triple the dimensionality of every feature set.

## 4 PERTURBED QUANTIZATION

In this section, we review some concepts and basic facts from the original PQ method, such as the notion of a "contributing mode" and "contributing DCT coefficient", and justify the selection of the second quality factor for side-informed embedding in recompressed images.

Because double compression can introduce strong artifacts into the distribution of DCT coefficients [17, 46], it is important to avoid such combinations in steganography because the embedding could be very detectable using, e. g., the JPEG Rich Model (JRM). Figure 2 shows a few examples of artifacts due to double compression. In

[3] https://www.pytorchlightning.ai/

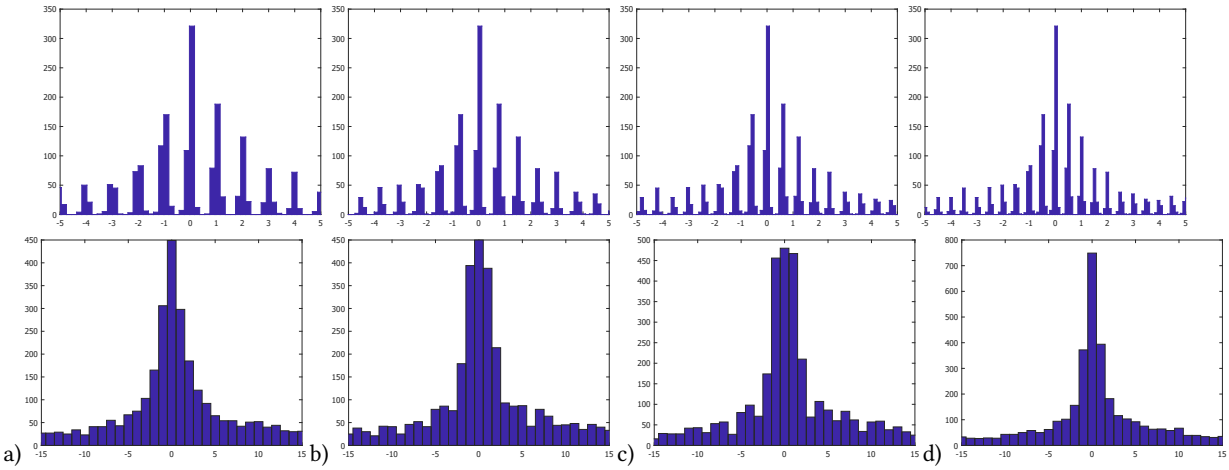


**Figure 2: Histogram of a DCT mode compressed first with quantization step $q_{kl}^{(1)} = 3$ and further compressed with quantization step $q_{kl}^{(2)}$ equal to a) 3, b) 4, c) 5, d) 6. Top: before rounding of the DCT coefficients, bottom: after rounding. The spikes in top row are around multiples of $q_{kl}^{(1)}/q_{kl}^{(2)}$. Only cases b) and d) correspond to contributing modes.**

PQ [23], and in this paper as well, we wish to have after the second compression as many DCT coefficients with rounding errors $|e_{kl}| \sim 1/2$ as possible as the rounding of such coefficients can be intuitively perturbed with little impact on detectability. Note that this is in line with the modern understanding of side-informed steganography [19].

When recompressing a JPEG image compressed with quantization table $\mathbf{q}^{(1)}$ with quantization table $\mathbf{q}^{(2)}$, the DCT mode $(k, l)$, $k, l = 0, \ldots, 7$ is called *contributing* if there exist $m, n \in \mathbb{Z}$ such that

$$m \cdot q_{kl}^{(1)} = n \cdot q_{kl}^{(2)} + \frac{1}{2} q_{kl}^{(2)}. \tag{3}$$

These modes guarantee the existence of DCT coefficients with the absolute value of the rounding error in the DCT domain close to $1/2$. As shown in [10], after recompression the DCT coefficients before rounding to integers follow a Gaussian distribution

$$\tilde{c}_{kl}^{(2)} \sim \mathcal{N}\left( c_{kl}^{(1)} \frac{q_{kl}^{(1)}}{q_{kl}^{(2)}}, \frac{1}{12 (q_{kl}^{(2)})^2} \right), \tag{4}$$

where the mean can be written from (3) as

$$\begin{aligned} \mathbb{E}[\tilde{c}_{kl}^{(2)}] &= c_{kl}^{(1)} \frac{q_{kl}^{(1)}}{q_{kl}^{(2)}} \\ &= n' + \frac{1}{2}, \end{aligned} \tag{5}$$

where it was assumed that $c_{kl}^{(1)}$ and some $n' \in \mathbb{Z}$ play the role of $m, n$ in (3). With such a small variance (4), it follows that after rounding to the nearest integers the rounding errors of these coefficients will be clustered around $\pm 1/2$.

In [23], the following useful theorem is proved.

Theorem 4.1. *The mode $(k, l)$ is contributing if and only if $q_{kl}^{(2)}/g$ is even, where $g = \gcd(q_{kl}^{(1)}, q_{kl}^{(2)})$ is the greatest common divisor of $q_{kl}^{(1)}$ and $q_{kl}^{(2)}$. Furthermore, all contributing multiples $m$ of $q_{kl}^{(1)}$ are expressed by the formula*

$$m = (2n + 1) \frac{q_{kl}^{(2)}}{2g}, \; n \in \mathbb{Z}. \tag{6}$$

In PQ steganography, embedding is executed only in contributing coefficients, which by Theorem 4.1, means in coefficients satisfying $c_{kl}^{(1)} = (2n + 1) \frac{q_{kl}^{(2)}}{2g}$ for some $n \in \mathbb{Z}$. We wish to emphasize that not all coefficients in contributing modes are contributing.

The motivation behind using only these coefficients is simple. It was shown [10] that the rounding errors in the DCT domain after the second compression $e_{kl}$ follow a Gaussian distribution folded into the interval $[-1/2, 1/2]$:

$$e_{kl} \sim \mathcal{N}_F \left( c_{kl}^{(1)} \frac{q_{kl}^{(1)}}{q_{kl}^{(2)}}, \frac{1}{12 (q_{kl}^{(2)})^2} \right), \tag{7}$$

where the mean of this distribution is $\mathbb{E}[e_{kl}] = c_{kl}^{(1)} \frac{q_{kl}^{(1)}}{q_{kl}^{(2)}} - [c_{kl}^{(1)} \frac{q_{kl}^{(1)}}{q_{kl}^{(2)}}]$. It is then clear that for a contributing mode $(k, l)$

$$\begin{aligned} \mathbb{E}[e_{kl}] &= c_{kl}^{(1)} \frac{q_{kl}^{(1)}}{q_{kl}^{(2)}} - [c_{kl}^{(1)} \frac{q_{kl}^{(1)}}{q_{kl}^{(2)}}] \\ &= c_{kl}^{(1)} \frac{q_{kl}^{(1)}/g}{q_{kl}^{(2)}/g} - [c_{kl}^{(1)} \frac{q_{kl}^{(1)}/g}{q_{kl}^{(2)}/g}] \\ &= c_{kl}^{(1)} \frac{u}{2v} - [c_{kl}^{(1)} \frac{u}{2v}] \end{aligned} \tag{8}$$

for some $u, v \in \mathbb{Z}$ coprime because Theorem 4.1 states that the denominators in (8) are even. Then in every case where $v$ divides $c_{kl}^{(1)} \cdot u$ and $2v$ does not divide $c_{kl}^{(1)} \cdot u$ we get the desirable $|\mathbb{E}[e_{kl}]| = 1/2$.

Equipped with this knowledge, we would now like to maximize the number of rounding errors that are close to $1/2$ in absolute value. Because the coefficients of the SC image $c_{kl}^{(1)}$ are given, the easiest way to ensure this for as many coefficients as possible is to let $v$ divide $u$. Since $u, v$ are coprime, this means $u = v = 1$ and thus $q_{kl}^{(2)} = 2q_{kl}^{(1)}$.[4] In this case, a coefficient $c_{kl}^{(1)}$ from a contributing mode is contributing whenever it is odd.

Enforcing the constraint $q_{kl}^{(2)} = 2q_{kl}^{(1)}$, however, would lead to non-standard quantization tables, and thus potentially an easy artifact of embedding. This is why in our work, we limit ourselves to standard quantization tables. Recall that the luminance quantization table for quality factor $Q$ is defined as

$$\mathbf{q}(Q) = \begin{cases} \max\left\{\mathbf{1}, \left\lfloor 2\mathbf{q}(50)\left(1 - \frac{Q}{100}\right)\right\rfloor\right\}, & Q > 50 \\ \min\left\{255 \times \mathbf{1}, \left\lfloor \mathbf{q}(50)\frac{50}{Q}\right\rfloor\right\}, & Q \leq 50, \end{cases} \tag{9}$$

where the luminance quantization table for quality factor 50 is

$$\mathbf{q}(50) = \begin{pmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{pmatrix}. \tag{10}$$

For the chrominance quantization table $\mathbf{q}_C(Q)$ at quality $Q$, the same formula (9) applies with chrominance quantization table at quality 50

$$\mathbf{q}_C(50) = \begin{pmatrix} 17 & 18 & 24 & 47 & 99 & 99 & 99 & 99 \\ 18 & 21 & 26 & 66 & 99 & 99 & 99 & 99 \\ 24 & 26 & 56 & 99 & 99 & 99 & 99 & 99 \\ 47 & 66 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \end{pmatrix}. \tag{11}$$

For simplicity, let us now work only with luminance quantization tables and $Q > 50$

$$\mathbf{q}(Q) = 2\mathbf{q}(50)\left(1 - \frac{Q}{100}\right). \tag{12}$$

Combining with our condition $q_{kl}^{(2)} = 2q_{kl}^{(1)}$, we obtain a relationship between the first and second quality factors $Q_1$ and $Q_2$:

$$\begin{aligned} \mathbf{q}(Q_2) &= 2\mathbf{q}(Q_1) \\ &= 2\left(2\mathbf{q}(50)\left(1 - \frac{Q_1}{100}\right)\right) \\ &= 2\mathbf{q}(50)\left(1 - \frac{2(Q_1 - 50)}{100}\right) \\ &= \mathbf{q}(2(Q_1 - 50)) \end{aligned}$$

[4] This relationship was derived in [23] only experimentally by virtue of Figure 3 in Sec. 4.3.

| $Q$ | (75,50) | (90,80) | (95,90) |
|---|---|---|---|
| SI all - binary | 0.0915 | 0.0222 | 0.0091 |
| SI all - ternary | 0.0957 | 0.0250 | 0.0155 |
| J-UNIWARD | 0.2777 | 0.3599 | 0.3932 |

**Table 1: $P_E$ with DCTR at 0.4 bpnzac of J-UNIWARD in single compressed images, and SI-UNIWARD in double compressed images while embedding into all DCT modes, binary and ternary version. BOSSbase+BOWS2 dataset.**

or

$$Q_2 = 2(Q_1 - 50), \tag{13}$$

as also reported in [23] based on experiments. In this work, we will follow this recipe for the selection of $Q_2$ with one exception for $Q_1 = 100$, because in this case we would declare $Q_2 = 100$ and the embedding would be reliably detected using the Reverse JPEG Compatibility Attack (RJCA) [9, 13]. For this reason, for $Q_1 = 100$, we heuristically choose $Q_2 = 98$ as the largest quality not attackable by the RJCA.

Additionally, the same relationship holds for the chrominance quantization tables, which will help us extend this idea to color images. To relax the notation, from now we denote $Q = (Q_1, Q_2)$ the pair of quality factors used for recompression with $Q_1$ used for SC images.

## 4.1 Naive application of side-information

The most straightforward way to cast the idea behind the PQ within the modern embedding paradigm is to use a modern content-adaptive steganographic method, such as J-UNIWARD, and apply the standard way of incorporating side-information by modulating the embedding costs by the rounding errors obtained during recompression (2). Table 1 shows the comparison of such SI-UNIWARD scheme in DC images with J-UNIWARD in SC images under the assumption that the exact same absolute payload is embedded by both schemes. The side-informed scheme is much more detectable than non-informed J-UNIWARD in SC images. To make sure that the high detectability is not introduced by ternary embedding, we also include the results for the binary version of SI-UNIWARD. Both the binary and ternary versions, however, exhibit a similar level of (in)security.

We now investigate where this high detectability comes from. We will measure the impact of the embedding on the distribution of DCT coefficients from every mode $(k, l)$ using the steganographic Fisher Information

$$I_{kl} = \sum_{m \in \mathbb{Z}} \frac{1}{p_{kl}^{(c)}(m)} \left(\frac{\partial p_{kl}^{(s)}(m)}{\partial \alpha}|_{\alpha=0}\right)^2, \tag{14}$$

where $p_{kl}^{(c)}$ is the cover probability mass function (pmf) of DCT coefficients in mode $(k, l)$, $p_{kl}^{(s)}$ is the pmf of stego images in the same mode, and $\alpha$ is the relative payload size. Since we cannot easily model the stego pmf when using J-UNIWARD, we approximate the Fisher information with real data as

$$\tilde{I}_{kl} \quad = \quad \sum_{m\in\mathbb{Z}} \frac{1}{h^{(c)}_{kl}(m)} \left( \frac{h^{(c)}_{kl}(m) - h^{(s)}_{kl}(m)}{\alpha} \right)^2, \qquad (15)$$

where we use the actual histograms $h^{(c)}_{kl}$ and $h^{(s)}_{kl}$ of the cover and the corresponding stego images embedded with relative payload $\alpha$. We average (15) over 100 randomly chosen images from the BOSSbase dataset and show in Figure 3 the average FI per mode together with the contributing modes for three different qualities $Q$. We used payload $\alpha = 1.1$ bpnzac for the embedding of stego images because for smaller payloads the approximation of the FI (15) does not utilize many changes in histograms and thus does not provide any useful feedback. We can clearly see a relationship between the non-contributing modes and the modes with high $\tilde{I}_{kl}$, which suggests that embedding in these modes is much more detectable. The only notable exception to this is in high frequencies of the lowest tested quality $Q = (75, 50)$. In this case, almost all cover coefficients are equal to zero due to the strong quantization, which leads to inaccurate estimates of the Fisher Information. We further report that the average FI across all non-contributing modes is 2–5 times larger than the average FI in the contributing modes. Remembering that the FI is in the error exponent of the likelihood ratio test, allowing embedding changes in non-contributing modes will have a grave impact on security.

To further support that the embedding into non-contributing modes is the culprit, we show in Figure 4 boxplots of the differences between stego and cover histograms. The differences in histograms exist because of the bias in the SI embedding towards coefficients with large rounding errors due to the nature of the cost modulation (2). For non-contributing modes, these coefficients are located at the peaks of mode histograms (see Figure 2 c)), which after embedding causes a very detectable distortion in the DCT mode histogram because these peaks will get deformed. Contributing modes do not suffer from this because they either have double peaks in histograms, which will be preserved during embedding, or no peaks (except at zero) (see Figure 2 b) and d)).

## 4.2 Restricting the embedding

The results from the previous section give a direction on how to adjust the side-informed embedding in double compressed images in order to avoid introducing changes into structures that exist in the distribution of coefficients of DC images. Constraining the embedding only to contributing multiples of $q^{(1)}_{kl}$ (6) as in the original PQ algorithm seems like the best option, however, this severely limits the capacity of the embedding. Table 2 shows the detection error with DCTR features across a wide range of payloads. Once the payload reaches 0.4 bpnzac at $Q = (95, 90)$, the detection error drops drastically. We verified that these drops indeed correspond to embedding messages that are simply too large to fit only into contributing coefficients. Hence, the embedding algorithm starts making changes in other coefficients, which happens without any content-adaptivity because the embedding spills into forbidden coefficients assigned with the same "wet cost."

Since we cannot embed into all modes securely and embedding only into contributing coefficients seems very limiting in terms of the maximal embeddable payload, we consider embedding into all coefficients from contributing modes because of the smaller impact of the embedding in terms of the FI (15) (see Figure 3 for an example).

Figure 5 shows the embedding capacity as the number of "changeable coefficients" per non-zero AC DCT coefficients of the single

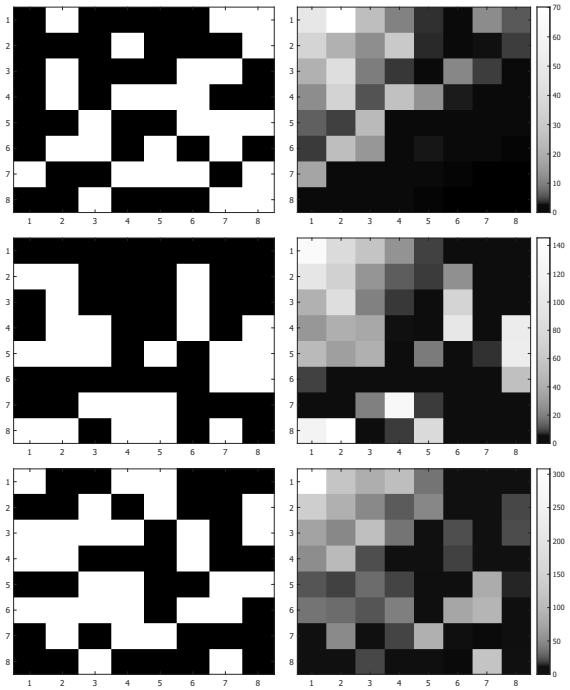

**Figure 3: Top: $Q = (75, 50)$, middle: $Q = (90, 80)$, bottom: $Q = (95, 90)$. Left: in black are contributing DCT modes, in white are non-contributing modes. Right: approximation of FI $\tilde{I}_{kl}$ per mode averaged over 100 images from BOSSbase embedded with 1.1 bpnzac.**

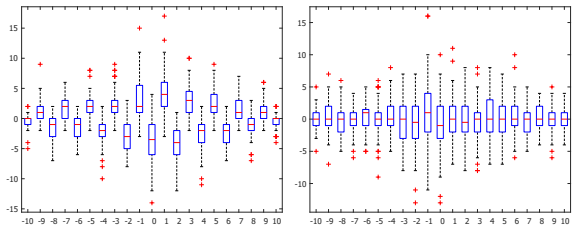

**Figure 4: Boxplots showing the differences between the distribution of DCT coefficients from stego images embedded with SI-UNIWARD (0.4 bpnzac) when embedding into all modes and cover images across 100 randomly selected images from BOSSbase with double compression quality $Q = (90, 80)$. Left: non-contributing mode $(2, 1)$ with quantization steps 2 and 5, Right: contributing mode $(1, 2)$ with quantization steps 2 and 4.**

| Payload | 0.3 bpnzac | | | 0.4 bpnzac | | | 0.5 bpnzac | | | 0.6 bpnzac | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $Q$ | (75,50) | (90,80) | (95,90) | (75,50) | (90,80) | (95,90) | (75,50) | (90,80) | (95,90) | (75,50) | (90,80) | (95,90) |
| Binary contr coefficients | 0.4082 | 0.3871 | 0.4197 | 0.3424 | 0.3381 | 0.0164 | 0.1990 | 0.0385 | 0.0017 | 0.0230 | 0.0026 | 0.0004 |
| Binary, contr modes | 0.4085 | 0.3895 | 0.4477 | 0.3441 | 0.3526 | 0.2940 | 0.2705 | 0.2813 | 0.0167 | 0.2118 | 0.1247 | 0.0002 |
| Ternary, contr modes | 0.4034 | 0.3922 | 0.4536 | 0.3660 | 0.3587 | 0.3530 | 0.2909 | 0.3118 | 0.1929 | 0.2375 | 0.2170 | 0.0284 |

**Table 2: $P_E$ with DCTR of SI-UNIWARD in double compressed images. Comparison between embedding into contributing coefficients and all coefficients in contributing modes. Binary and ternary embedding. BOSSbase+BOWS2 dataset.**

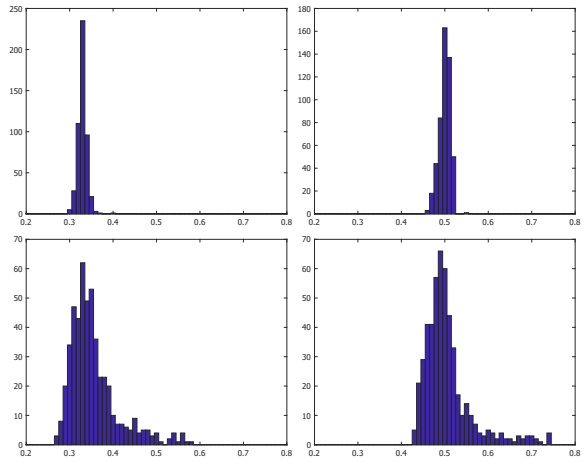

**Figure 5: Average number of changeable coefficients per non-zero AC DCT coefficients over 500 randomly chosen images compressed with quality factor 95. Top: BOSSbase+BOWS2 (grayscale), bottom: ALASKA 2 (color), left: embedding only into contributing coefficients, right: embedding into all coefficients in contributing modes.**

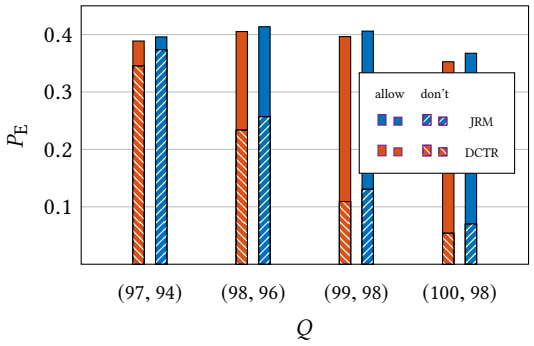


**Figure 6: Detection error $P_E$ of SI-UNIWARD at 0.4 bpnzac in DC images when modes with $q_{kl}^{(1)} = q_{kl}^{(2)}$ are/are not allowed for embedding. BOSSbase+BOWS2 dataset.**

compressed image. Changeable coefficients are either only contributing coefficients or all coefficients from all contributing modes. It was verified for a range of qualities that for grayscale and color images, using all coefficients in contributing modes increases the average embedding capacity by approximately 50%.

For a larger embedding capacity, we therefore relax the embedding restriction by allowing embedding into all coefficients inside contributing modes, not only the contributing coefficients. The results are shown in Table 2, where we can see that for payloads as large as 0.6 bpnzac, the ternary embedding into all coefficients inside contributing modes provides overall best security. Based on this analysis, we will keep using this embedding strategy for the rest of the paper.

*4.2.1 High qualities.* In the derivation of (13), we did not consider the nonlinear dependence of quantization steps on the quality factor due to taking the maximum with one and rounding. While the rounding operation introduces the same nonlinearity for every quantization step regardless of the quality factor applied, the maximum will only be applied for very high quality factors and mainly for low frequency modes. Note that if $Q_2 = 2(Q_1 - 50)$, then $q_{kl}^{(1)} = q_{kl}^{(2)}$ if and only if $q_{kl}^{(2)} = 1$.This introduces an issue that needs to be addressed, because when the maximum starts introducing ones in the second quantization table (this occurs for $Q_2 \geq 93$), we would end up, with our definition of a contributing mode, with very few contributing modes. This is because if a second quantization step is equal to one, then $q_{kl}^{(2)}/\text{gcd}(q_{kl}^{(1)}, q_{kl}^{(2)}) = 1$ is not even, which would effectively prevent us from embedding non-trivial payloads. To this end, we decided to allow embedding into modes with $q_{kl}^{(1)} = q_{kl}^{(2)}$. Such modes are not contributing, but the embedding does not suffer from these modes since this combination of quantization steps does not introduce easily exploitable artifacts (the JRM performs very poorly in these cases [10]). Figure 2 a) also suggests that the histogram of such modes does not start showing any drastic artifacts. Even though the mean of the DCT error (8) is zero in these cases, its variance (7) is equal to 1/12, which still ensures quite a few of the DCT rounding errors to be close to ±1/2. The effect of allowing embedding in these modes can be seen in Figure 6. We verified that the high detectability for the case where we do not allow embedding into modes with $q_{kl}^{(1)} = q_{kl}^{(2)}$ comes from the payload being too large, a problem we have encountered in the previous section too, while trying to embed only into contributing coefficients. Consequently, the embedding changes were made in non-contributing modes without content-adaptivity.

*4.2.2 Color.* Embedding in color images can spread the payload across luminance and the two chrominance channels. Several different payload spreading strategies into the three $YC_bC_r$ channels were recently proposed in [16, 41]. It was reported in [16] that for J-UNIWARD, the CCM (Color Channels Merging), which distributes the payload by minimizing the additive distortion across all three channels, and CCFR (Color Channels Fixed Repartition) with

repartition parameter $\gamma = 0.2$, which puts a fraction of payload into chrominance channels, provide almost the same level of security. We wanted to verify whether this remains true for SI-UNIWARD in DC images. After testing with DCTR on SI-UNIWARD with CCM and CCFR(0.2), we found, to our surprise, that the CCM strategy was much more detectable. We believe CCM should be the optimal strategy for spreading the payload because it distributes the payload automatically without forcing a fixed portion of the payload into chrominance. It was identified that the poor performance of CCM is caused by the discrepancy between the embedding costs in luminance and chrominance channels, which forces a vast majority of the payload into the luminance channel. After careful inspection of the embedding algorithm for SI-UNIWARD, we realized that the culprit was the stabilizing constant $\sigma$ used in J-UNIWARD's distortion function [29]:

$$D(\mathbf{X}, \mathbf{Y}) = \sum_{k=1}^{3} \sum_{u=1}^{n_1} \sum_{v=1}^{n_2} \frac{|W_{uv}^{(k)}(\mathbf{X}) - W_{uv}^{(k)}(\mathbf{Y})|}{\sigma + |W_{uv}^{(k)}(\mathbf{X})|}, \tag{16}$$

where $\mathbf{X}$ and $\mathbf{Y}$ represent the cover and stego images in the pixel domain (in one channel), $n_1, n_2$ are the number of DCT blocks in the vertical and horizontal directions, and $W_{uv}^{(k)}(\cdot)$ the wavelet transformation based on Daubechies 8-tap wavelet directional filter bank. By default, $\sigma$ is set to $2^{-6}$, which would not be an issue if the normalization factor in (16) was on a similar scale for luminance and chrominance channels. While this is true for SC images, for DC images it is not. In fact, $|W_{uv}^{(k)}(\mathbf{X})|, k \in \{1, 2, 3\}$ in chrominance channels can be by several orders of magnitude smaller than in the luminance channel. We believe that this is due to much harsher quantization in chrominance channels of DC images compared to SC images (see the quantization tables (10) and (11)). Thus, we claim that the stabilizing constant has to be smaller in chrominance channels. Keeping the original luminance stabilizing constant $\sigma_Y = 2^{-6}$, in Figure 7 we show $P_E$ of DCTR on SI-UNIWARD with 0.4 bpnzac with the CCM spreading strategy across a range of values for the stabilizing constant in chrominance channels $\sigma_C$. We see that for qualities (90, 80) and (95, 90), $\sigma_C$ is reaching the best security for $\sigma_C = 2^{-15}$. For the lowest quality (75, 50), the most secure $\sigma_C$ is at $2^{-16}$. In order to have a unified setting, we declare $\sigma_C = 2^{-15}$ for every quality combination, even at a loss for the low qualities. With $\sigma_C$ adjusted this way, we searched for optimal $\sigma_Y$. Coincidentally, the default value $\sigma_Y = 2^{-6}$ provides the best performance.

## 5 EVALUATION

To show the benefit of embedding in recompressed images, we contrast the empirical security with embedding in the corresponding single-compressed cover images. To summarize the embedding algorithm, we use ternary embedding in all coefficients belonging to contributing modes and modes with $q_{kl}^{(1)} = q_{kl}^{(2)}$. For color images, we furthermore improved the security by changing the chrominance stabilizing constant $\sigma_C$ of J-UNIWARD's costs. The second quality factor $Q_2$ used for recompression is selected by Eq. (13) with one exception for $Q_1 = 100$ where we set $Q_2 = 98$.

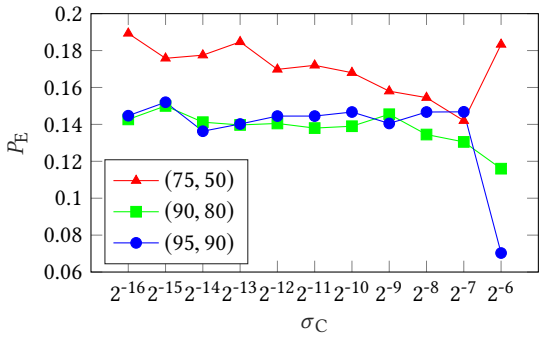


**Figure 7: $P_E$ with DCTR of CCM-SI-UNIWARD in DC images with different values of the stabilizing constant $\sigma_C$ of chrominance channels $C_r$ and $C_b$, with the luminance constant at the default $\sigma_Y = 2^{-6}$. Three qualities (75, 50), (90, 80), and (95, 90) are shown. ALASKA 2 dataset.**

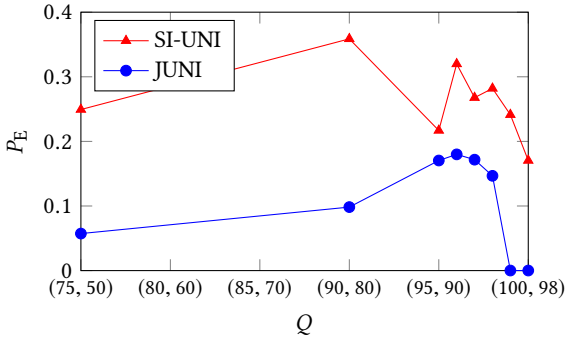


**Figure 8: Detection error $P_E$ of J-UNIWARD in SC images and SI-UNIWARD in DC images at 0.4 bpnzac. Only the best detector's performance is shown. BOSSbase+BOWS2 dataset.**

### 5.1 Grayscale

We test the proposed scheme on a range of qualities with the detectors described in Section 3. We also tested GFR [40] and its selection channel aware version, where we used J-UNIWARD for estimating the selection channel. Both of these feature sets, however, did not bring any improvement over DCTR. For the highest qualities (99, 98) and (100, 98), we also trained e-SRNet [9], SRNet trained on rounding errors of pixel values after decompression, as it is the best detector for the highest quality JPEGs. Only the best detector's detection error $P_E$ on SI-UNIWARD in DC images and J-UNIWARD in SC images with 0.4 bpnzac is shown in Figure 8. For J-UNIWARD, the best detector is always the SRNet, while for the two highest qualities, it is e-SRNet (note the extremely low errors). The best detector for SI-UNIWARD is also SRNet, with one exception at quality (90, 80), where DCTR provides a better detection. The e-SRNet performed substantially worse than SRNet, confirming that the RJCA is not applicable with the quality selection rule (13). Overall, the improvement of embedding in recompressed images when compared to J-UNIWARD ranges between $5 - 25\%$ in terms of $P_E$.

| $Q$ | Detector | 0.1 bpnzac | | 0.2 bpnzac | | 0.3 bpnzac | | 0.4 bpnzac | |
|---|---|---|---|---|---|---|---|---|---|
| | | DC-SI | JUNI | DC-SI | JUNI | DC-SI | JUNI | DC-SI | JUNI |
| (75,50) | ccJRM | 0.4505 | 0.4850 | 0.4439 | 0.4552 | 0.4225 | 0.4078 | 0.4147 | 0.3631 |
| | DCTR | **0.4484** | 0.4697 | **0.4397** | 0.4202 | 0.4034 | 0.3465 | 0.3660 | 0.2777 |
| | SRNet | 0.5000 | 0.3094 | 0.5000 | 0.1764 | **0.3189** | 0.0961 | **0.2493** | 0.0573 |
| (90,80) | ccJRM | 0.4134 | 0.4900 | **0.4057** | 0.4767 | 0.4020 | 0.4588 | 0.3849 | 0.4144 |
| | DCTR | **0.4114** | 0.4860 | 0.4061 | 0.4546 | **0.3922** | 0.4185 | **0.3587** | 0.3599 |
| | SRNet | 0.4469 | 0.3661 | 0.4461 | 0.2522 | 0.4390 | 0.1519 | 0.3884 | 0.0983 |
| (95,90) | ccJRM | 0.4876 | 0.4872 | 0.4881 | 0.4613 | 0.4736 | 0.4351 | 0.3771 | 0.3990 |
| | DCTR | **0.4823** | 0.4941 | **0.4839** | 0.4714 | 0.4536 | 0.4441 | 0.3530 | 0.3932 |
| | SRNet | 0.5000 | 0.4297 | 0.5000 | 0.3292 | **0.3686** | 0.2401 | **0.2169** | 0.1704 |

**Table 3: Detection error $P_E$ of SRNet, ccJRM, and DCTR for various payloads (bpnzac) of J-UNIWARD in SC and SI-UNIWARD in DC images. Boldface represents the best detector of the more secure algorithm at a fixed payload. BOSSbase+BOWS2 dataset.**

To obtain a better understanding of how the algorithms compare for smaller payloads, we trained the SRNet, ccJRM, and DCTR at qualities $(75, 50)$, $(90, 80)$, and $(95, 90)$ for various payloads. The results are shown in Table 3. We can see clear improvement over J-UNIWARD at every payload. Surprisingly, in many cases (especially for the lowest payloads), DCTR provides a better detection than SRNet on SI-UNIWARD. This suggests that the SRNet is not able to collect detection statistics from a somewhat detectable distortion in the DCT domain.

## 5.2 Color

Setting the chrominance stabilizing constant $\sigma_C = 2^{-15}$, we first reevaluate the spreading strategies CCFR and CCM [16]. Table 4 shows DCTR's $P_E$ on the CCFR strategy for several values of the repartition parameter $\gamma$. With increasing quality, the optimal parameter $\gamma$ also needs to grow as the best value of $\gamma$ for every quality is different. Interestingly, CCFR strategy with $\gamma = 0.2$ outperforms CCM at quality $(75, 50)$, but on the other two tested qualities, CCM achieves a better security. In Table 5, we include a comparison between SI-UNIWARD in DC images with $\sigma_C = 2^{-15}$ and J-UNIWARD in SC images across several payloads and several qualities, both schemes using the CCM payload spreading strategy. Using side-information provides an improvement in security up to 18% at quality $(75, 50)$ and payload 0.2 bpnzac. Interestingly, the non-informed J-UNIWARD is more secure in two tested scenarios: $Q = (90, 80)$ at 0.1 bpnzac and $Q = (95, 90)$ at 0.4 bpnzac. The latter is most likely caused by the large embedding payload in DC images because, as can be seen in Figure 5, the embedding capacity in color images has thicker left tail than in grayscale images. This is in line with the significant jumps in $P_E$ of SI-UNIWARD for lower payloads at $Q = (95, 90)$.

| | Repartition parameter $\gamma$ | | | | |
|---|---|---|---|---|---|
| $Q$ | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 |
| (75,50) | 0.1893 | **0.2008** | 0.1835 | 0.1517 | 0.1120 |
| (90,80) | 0.1105 | 0.1110 | **0.1265** | 0.1162 | 0.0772 |
| (95,90) | 0.0645 | 0.0837 | 0.1115 | **0.1247** | 0.0757 |

**Table 4: $P_E$ with DCTR of CCFR-SI-UNIWARD at 0.4 bpnzac in DC images with chrominance stabilizing constant $\sigma_C = 2^{-15}$. ALASKA 2 dataset.**

| $Q$ | Detector | 0.1 bpnzac | | 0.2 bpnzac | | 0.3 bpnzac | | 0.4 bpnzac | |
|---|---|---|---|---|---|---|---|---|---|
| | | DC-SI | JUNI | DC-SI | JUNI | DC-SI | JUNI | DC-SI | JUNI |
| (75,50) | JRM | **0.3362** | 0.4845 | **0.2957** | 0.4547 | **0.2120** | 0.4138 | **0.1210** | 0.3740 |
| | DCTR | 0.3708 | 0.4100 | 0.3478 | 0.2937 | 0.2735 | 0.1867 | 0.1758 | 0.1108 |
| | SRNet | 0.4093 | 0.2516 | 0.3243 | 0.1119 | 0.2736 | 0.0607 | 0.2524 | 0.0327 |
| (90,80) | JRM | 0.2885 | 0.4750 | **0.2658** | 0.4477 | 0.2368 | 0.4025 | 0.1903 | 0.3653 |
| | DCTR | 0.3120 | 0.4473 | 0.2835 | 0.3652 | **0.2085** | 0.2740 | **0.1500** | 0.1947 |
| | SRNet | 0.3978 | **0.3473** | 0.3933 | 0.2236 | 0.3353 | 0.1397 | 0.2394 | 0.0857 |
| (95,90) | JRM | **0.4300** | 0.4305 | 0.4088 | 0.3455 | 0.3310 | 0.2758 | 0.2208 | 0.2248 |
| | DCTR | 0.4305 | 0.4542 | **0.4032** | 0.3800 | 0.2883 | 0.3163 | 0.1520 | 0.2223 |
| | SRNet | 0.5000 | 0.4193 | 0.4268 | 0.3083 | **0.2604** | 0.2211 | 0.1372 | **0.1524** |

**Table 5: $P_E$ of SI-UNIWARD in DC images with $\sigma_C = 2^{-15}$ and J-UNIWARD in SC images, both using CCM strategy. ALASKA 2 dataset.**

# 6 DOUBLE COMPRESSION WITH THE SAME QUALITY

In this section, we investigate the case of side-informed steganography in images that were double compressed with the same quantization table. We included this analysis because the option $Q_1 = Q_2$ avoids introducing any histogram artifacts and it would allow us to embed into every DCT mode, thus significantly increasing the embedding capacity. Furthermore, and most importantly, it is not immediately obvious that side-informed embedding in this setup is extremely detectable and exhibits some very unusual properties, such as higher statistical detectability of smaller payloads than larger payloads. Recompression with the same quality was previously studied for forensic purposes in [36].

As shown in [10], embedding in DC images with $Q_1 = Q_2$ can be attacked with the RJCA. However, this work did not investigate the case of side-informed embedding. Since everywhere in this section it is assumed that $Q_1 = Q_2$, we will again refer to the compression quality simply as $Q$.

The performance of the e-SRNet as implemented in [10] can be seen in Figure 9. Note that the detection errors are much lower than for quality factor rule (13) in Table 3. Moreover, the most peculiar behavior can be observed for $Q < 93$ when the detection of smaller payloads is more reliable. We will now show that the rounding errors $\mathbf{e}$ can actually be partly recovered from the double compressed (and embedded) images with $Q_1 = Q_2$, which is responsible for this peculiar behavior.

## 6.1 Estimating the side information

Let us call the changes in the DCT coefficients introduced during the second compression as *inconsistencies*. In other words, the compression produces an inconsistency at $c_{kl}^{(2)}$ if $c_{kl}^{(1)} \neq c_{kl}^{(2)}$. Figure 10 shows that for $Q < 93$ the second compression does not introduce many inconsistencies mainly because there are no ones in the quantization table. We hypothesize that for lower qualities (where quantization tables do not contain any ones, i. e., $Q < 93$) the following claim holds: the fewer inconsistencies the better the estimate of the rounding error $\mathbf{e}$ can be obtained. Intuitively, this makes sense because if the second compression does not change

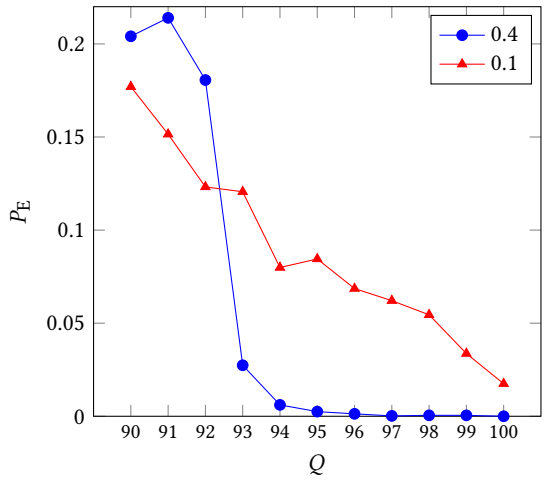


**Figure 9: $P_E$ with e-SRNet of SI-UNIWARD in DC images at 0.1 and 0.4 bpnzac when $Q_1 = Q_2$. BOSSbase+BOWS2 dataset.**

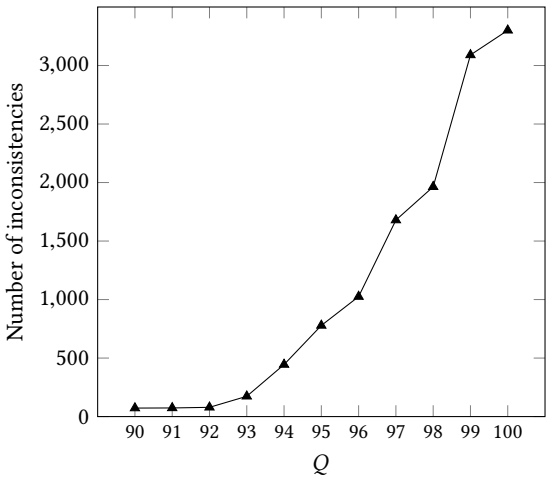


**Figure 10: Average number of inconsistencies across 1000 randomly selected images from BOSSbase with $Q_1 = Q_2$.**

any coefficient in a given DCT block then also the third compression would not change any coefficients. Therefore, one can compute the rounding errors $\mathbf{e}$ used during embedding (and thus nullify the effect of side-information) by simply compressing the DC image once more. Note that the embedding changes can also be considered inconsistencies. If the claim holds, it would immediately mean that we can get a better estimate of $\mathbf{e}$ with decreasing payload.

To estimate the DCT errors $\mathbf{e}$, we decompress a given (double compressed and possibly embedded) JPEG image to the spatial domain $\mathbf{y}^{(2)}$ and round to integers $\mathbf{x}^{(2)} = [\mathbf{y}^{(2)}]$. We then compress $\mathbf{x}^{(2)}$ again with the same quality settings to obtain the DCT coefficients after the third compression $\tilde{\mathbf{c}}^{(3)} = \mathrm{DCT}(\mathbf{x}^{(2)}) \oslash \mathbf{q}$, where $\mathbf{q}$ is the quantization table used in all compression steps. A simple estimate of the rounding error can be computed as $\hat{\mathbf{e}} = \tilde{\mathbf{c}}^{(3)} - [\tilde{\mathbf{c}}^{(3)}]$. This estimate $\hat{\mathbf{e}}$ is strongly correlated with the original $\mathbf{e}$. This is illustrated in

| $Q$ | Detector | DC-SI | JUNI |
|---|---|---|---|
| 75 | SRNet | 0.0585 | 0.0573 |
| | e-SRNet | 0.5000 | 0.5000 |
| 90 | SRNet | 0.0947 | 0.0983 |
| | e-SRNet | 0.2041 | 0.5000 |
| 95 | SRNet | 0.1856 | 0.1704 |
| | e-SRNet | 0.0000 | 0.5000 |

**Table 6: Detection error $P_E$ with SRNet and e-SRNet of J-UNIWARD in SC images and SI-UNIWARD in DC images at 0.4 bpnzac and $Q_1 = Q_2$. BOSSbase+BOWS2 dataset.**

Figure 11, which displays the mean square error (MSE) between the DCT rounding error and its estimate $\mathrm{MSE}(\mathbf{e}, \hat{\mathbf{e}}) = \frac{1}{n}\sum_{i=1}^{n}(e_i - \hat{e_i})^2$. The estimate is computed from cover images and SI-UNIWARD images embedded with 0.1 and 0.4 bpnzac. With increasing payload (increasing number of inconsistencies), the estimate of the errors is getting worse across all qualities, which confirms our insight. For a smaller payload, we have a better estimate of the side-information. To verify that the estimate $\hat{\mathbf{e}}$ can be used for estimating the selection channel, we include in Figure 12 the correlation between $\hat{\mathbf{e}}$ and the difference $\beta^+ - \beta^-$, where $\beta^+, \beta^-$ are the probabilities of changing the coefficients by $+1$ and $-1$, respectively.

This should be thought of more as a proof of concept because the e-SRNet most likely does not compress the image for the third time as it might compute the estimate of the rounding errors in some other, perhaps better way. It turns out that a similar estimate can be achieved by compressing the spatial rounding error $\mathbf{u} = \mathbf{x}^{(2)} - \mathbf{y}^{(2)}$, which is what the e-SRNet is trained on, and computing the rounding error in the DCT domain.

Using $Q_1 = Q_2$ for qualities below 93 will not be beneficial because the rounding errors $\mathbf{e}$ follow the distribution (7), which for $Q_1 = Q_2$ can be simplified as

$$e_{kl} \sim \mathcal{N}_F\left(0, \frac{1}{12(q_{kl}^{(2)})^2}\right). \tag{17}$$

It should be clear that for large quantization steps the errors will be clustered very closely around zero, thus having a negligible effect on the embedding. Moreover, as already mentioned above, for lower qualities there are not many inconsistencies, which is also due to (17). Therefore, the image is virtually identical to its single compressed version and there is not much side-information available. All these observations would suggest that the steganographic security would be very close to the non-informed J-UNIWARD on SC images. This is indeed verified in Table 6 showing that the SRNet on SI-UNIWARD in DC images with $Q_1 = Q_2$ has almost the same performance as on J-UNIWARD in SC images. The only difference is for quality 95, where the side-informed version seems to be slightly more secure thanks to the side-information generated in modes with small quantization steps (17). However, at this high quality the RJCA is already kicking in for the DC images, while not yet for SC images [9, 13], making steganography in DC images highly detectable.

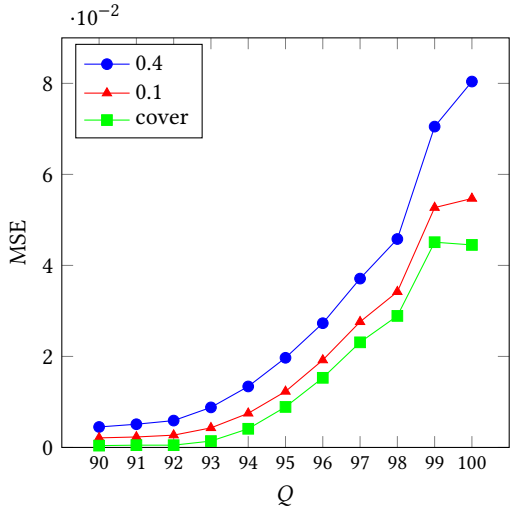


**Figure 11: Average MSE between e and ê across 300 randomly selected images with $Q_1 = Q_2$. The estimate ê is computed from cover images and SI-UNIWARD at 0.1 and 0.4 bpnzac with $Q_1 = Q_2$. BOSSbase+BOWS2 dataset.**

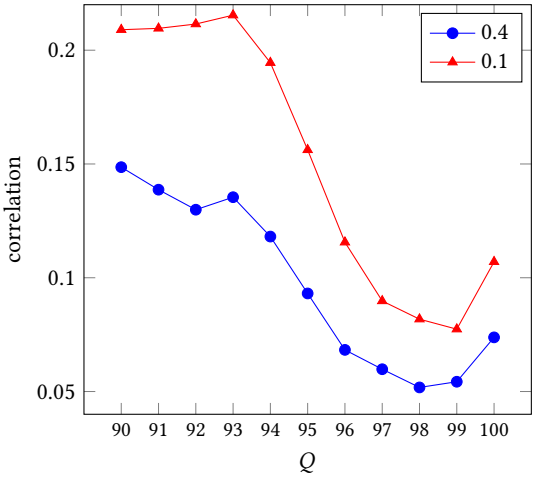


**Figure 12: Correlation between ê and $\beta^+ - \beta^-$ across 300 randomly selected images for SI-UNIWARD at 0.1 and 0.4 bpnzac with $Q_1 = Q_2$. BOSSbase+BOWS2 dataset.**

## 7 CONCLUSIONS

In this paper, we pursued an idea of improving empirical steganographic security by embedding into a recompressed JPEG image instead of the original single compressed image. This idea of generating steganographic side-information by recompressing the JPEG cover image was first explored in the so-called Perturbed Quantization steganography 17 years ago. Surprisingly, when simply adopting modern coding coupled with cost modulation typically used in side-informed embedding, the security of the resulting embedding is extremely poor. This tells us that the side-information generated by recompression needs to be treated differently.

By quantifying the effect of embedding on the distribution of DCT coefficients from specific DCT modes using the steganographic Fisher information, we learned that modes that do not contain any contributing DCT coefficients (coefficients with rounding errors close in absolute value to 1/2 during recompression) exhibit artifacts in their distribution after embedding, which brings the security down. This was remedied by constraining the embedding only to contributing modes. Besides dramatically improving the security, this choice also allowed embedding larger, and thus more practical, payloads than embedding only into contributing DCT coefficients akin to the original PQ. To demonstrate the usefulness of the proposed technique, the empirical security was compared with embedding into the single compressed cover image while fixing the absolute payload in bits.

The method was also adapted for color images with the CCM payload-spreading strategy. To achieve a good security, however, the stabilizing constant of the J-UNIWARD algorithm had to be modified for the chrominance channels due to their different dynamic range.

Finally, we show that generating the side-information by recompressing with the same quantization table makes the embedding algorithm much more detectable because in such cases the side-information can be reliably estimated. This also leads to a bizarre situation for qualities below 93 when the detection power increases with smaller payloads.

## ACKNOWLEDGMENTS

The work on this paper was supported by NSF grant no. 2028119.

## REFERENCES

[1] H. Abdulrahman, M. Chaumont, P. Montesinos, and B. Magnier. Color image steganalysis using correlations between RGB channels. In *Proceedings 10th International Conference on Availability, Reliability and Security (ARES), 4th International Workshop on Cyber Crime (IWCC)*, pages 448–454, Toulouse, France, August 24–28 2015.

[2] H. Abdulrahman, M. Chaumont, P. Montesinos, and B. Magnier. Color image steganalysis using RGB channel geometric transformation measures. *Wiley Journal on Security and Communication Networks*, February 2016.

[3] P. Bas, T. Filler, and T. Pevný. Break our steganographic system – the ins and outs of organizing BOSS. In T. Filler, T. Pevný, A. Ker, and S. Craver, editors, *Information Hiding, 13th International Conference*, volume 6958 of Lecture Notes in Computer Science, pages 59–70, Prague, Czech Republic, May 18–20, 2011.

[4] P. Bas and T. Furon. BOWS-2. http://bows2.ec-lille.fr, July 2007.

[5] M. Boroumand, M. Chen, and J. Fridrich. Deep residual network for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 14(5):1181–1193, May 2019.

[6] M. Boroumand and J. Fridrich. Synchronizing embedding changes in side-informed steganography. In *Proceedings IS&T, Electronic Imaging, Media Watermarking, Security, and Forensics 2020*, San Francisco, CA, January 26–30 2020.

[7] S. Bozinovski and A. Fulgosi. The influence of pattern similarity and transfer learning upon training of a base perceptron b2. In *Proceedings of Symposium Informatica 3-121-5*, 1976.

[8] J. Butora and J. Fridrich. Effect of jpeg quality on steganographic security. In R. Cogranne and L. Verdoliva, editors, *The 7th ACM Workshop on Information Hiding and Multimedia Security*, Paris, France, July 3–5, 2019. ACM Press.

[9] J. Butora and J. Fridrich. Reverse JPEG compatibility attack. *IEEE Transactions on Information Forensics and Security*, 15:1444–1454, 2020.

[10] J. Butora and J. Fridrich. Extending the reverse JPEG compatibility attack to double compressed images. In *Proceedings IEEE, International Conference on Acoustics, Speech, and Signal Processing*, Toronto, Canada, June 6–11, 2021.

[11] J. Butora, Y. Yousfi, and J. Fridrich. How to pretrain for steganalysis. In *The 9th ACM Workshop on Information Hiding and Multimedia Security*, Brussels, Belgium, June 21–25, 2021.

[12] K. Chubachi. An ensemble model using CNNs on different domains for ALASKA2 image steganalysis. In *IEEE International Workshop on Information Forensics and Security*, New York, NY, December 6–11, 2020.

[13] R. Cogranne. Selection-channel-aware reverse JPEG compatibility for highly reliable steganalysis of JPEG images. In *Proceedings IEEE, International Conference on Acoustics, Speech, and Signal Processing*, pages 2772–2776, Barcelona, Spain, May 4–8, 2020.

[14] R. Cogranne, Q. Giboulot, and P. Bas. The ALASKA steganalysis challenge: A first step towards steganalysis "Into the wild". In R. Cogranne and L. Verdoliva, editors, *The 7th ACM Workshop on Information Hiding and Multimedia Security*, Paris, France, July 3–5, 2019. ACM Press.

[15] R. Cogranne, Q. Giboulot, and P. Bas. ALASKA–2: Challenging academic research on steganalysis with realistic images. In *IEEE International Workshop on Information Forensics and Security*, New York, NY, December 6–11, 2020.

[16] R. Cogranne, Q. Giboulot, and P. Bas. Steganography by minimizing statistical detectability: The cases of jpeg and color images. ACM Press, 2020.

[17] J. L. Davidson and P. Parajape. Double-compressed JPEG detection in a steganalysis system. In *Annual ADFSL Conference on Digital Forensics, Security, and Law*, May 30, 2012.

[18] T. Denemark, M. Boroumand, and J. Fridrich. Steganalysis features for content-adaptive JPEG steganography. *IEEE Transactions on Information Forensics and Security*, 11(8):1736–1746, August 2016.

[19] T. Denemark and J. Fridrich. Side-informed steganography with additive distortion. In *IEEE International Workshop on Information Forensics and Security*, Rome, Italy, November 16–19 2015.

[20] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE conference on computer vision and pattern recognition*, pages 248–255, June 20–25, 2009.

[21] T. Filler, J. Judas, and J. Fridrich. Minimizing additive distortion in steganography using syndrome-trellis codes. *IEEE Transactions on Information Forensics and Security*, 6(3):920–935, September 2011.

[22] J. Fridrich. On the role of side-information in steganography in empirical covers. In A. Alattar, N. D. Memon, and C. Heitzenrater, editors, *Proceedings SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics 2013*, volume 8665, pages 1–11, San Francisco, CA, February 5–7, 2013.

[23] J. Fridrich, M. Goljan, and D. Soukal. Perturbed quantization steganography. *ACM Multimedia System Journal*, 11(2):98–107, 2005.

[24] J. Fridrich, M. Goljan, D. Soukal, and P. Lisoněk. Writing on wet paper. In T. Kalker and P. Moulin, editors, *IEEE Transactions on Signal Processing, Special Issue on Media Security*, volume 53, pages 3923–3935, October 2005. (journal version).

[25] Q. Giboulot, R. Cogranne, and P. Bas. JPEG steganography with side information from the processing pipeline. In *Proceedings IEEE, International Conference on Acoustics, Speech, and Signal Processing*, pages 2767–2771, Barcelona, Spain, May 4–8, 2020.

[26] Q. Giboulot, R. Cogranne, and P. Bas. Synchronization Minimizing Statistical Detectability for Side-Informed JPEG Steganography. In *IEEE International Workshop on Information Forensics and Security*, New York, NY, December 6–11, 2020.

[27] L. Guo, J. Ni, and Y. Q. Shi. Uniform embedding for efficient JPEG steganography. *IEEE Transactions on Information Forensics and Security*, 9(5):814–825, May 2014.

[28] V. Holub and J. Fridrich. Low-complexity features for JPEG steganalysis using undecimated DCT. *IEEE Transactions on Information Forensics and Security*, 10(2):219–228, February 2015.

[29] V. Holub, J. Fridrich, and T. Denemark. Universal distortion design for steganography in an arbitrary domain. *EURASIP Journal on Information Security, Special Issue on Revised Selected Papers of the 1st ACM IH and MMS Workshop*, 2014:1, 2014.

[30] X. Hu, J. Ni, W. Su, and J. Huang. Model-based image steganography using asymmetric embedding scheme. *Journal of Electronic Imaging*, 27(4):1 – 7, 2018.

[31] F. Huang, J. Huang, and Y.-Q. Shi. New channel selection rule for JPEG steganography. *IEEE Transactions on Information Forensics and Security*, 7(4):1181–1191, August 2012.

[32] F. Huang, W. Luo, J. Huang, and Y.-Q. Shi. Distortion function designing for JPEG steganography with uncompressed side-image. In W. Puech, M. Chaumont, J. Dittmann, and P. Campisi, editors, *1st ACM IH&MMSec. Workshop*, Montpellier, France, June 17–19, 2013.

[33] J. Kodovský and J. Fridrich. Calibration revisited. In J. Dittmann, S. Craver, and J. Fridrich, editors, *Proceedings of the 11th ACM Multimedia & Security Workshop*, pages 63–74, Princeton, NJ, September 7–8, 2009.

[34] J. Kodovský and J. Fridrich. Steganalysis of JPEG images using rich models. In A. Alattar, N. D. Memon, and E. J. Delp, editors, *Proceedings SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics 2012*, volume 8303, pages 0A 1–13, San Francisco, CA, January 23–26, 2012.

[35] J. Kodovský, J. Fridrich, and V. Holub. Ensemble classifiers for steganalysis of digital media. *IEEE Transactions on Information Forensics and Security*, 7(2):432–444, April 2012.

[36] S. Lai and R. Böhme. Block convergence in repeated transform coding: JPEG-100 forensics, carbon dating, and tamper detection. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3028–3032, 2013.

[37] T. Mingxing and V. L. Quoc. EfficientNet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the 36th International Conference on Machine Learning, ICML*, volume 97, pages 6105–6114, June 9–15, 2019.

[38] S. Ozcan and A. F. Mustacoglu. Transfer learning effects on image steganalysis with pre-trained deep residual neural network model. In *IEEE International Conference on Big Data (Big Data)*, pages 2280–2287, December 10–13, 2018.

[39] W. Pennebaker and J. Mitchell. *JPEG: Still Image Data Compression Standard*. Van Nostrand Reinhold, New York, 1993.

[40] X. Song, F. Liu, C. Yang, X. Luo, and Y. Zhang. Steganalysis of adaptive JPEG steganography using 2D Gabor filters. In P. Comesana, J. Fridrich, and A. Alattar, editors, *3rd ACM IH&MMSec. Workshop*, Portland, Oregon, June 17–19, 2015.

[41] T. Taburet, L. Filstroff, P. Bas, and W. Sawaya. An empirical study of steganography and steganalysis of color images in the JPEG domain. In *International Workshop on Digital Forensics and Watermarking (IWDW)*, Jeju, South Korea, 2018.

[42] G. Xu. Deep convolutional neural network to detect J-UNIWARD. In M. Stamm, M. Kirchner, and S. Voloshynovskiy, editors, *The 5th ACM Workshop on Information Hiding and Multimedia Security*, Philadelphia, PA, June 20–22, 2017.

[43] M. Yedroudj, F. Comby, and M. Chaumont. Yedroudj-net: An efficient CNN for spatial steganalysis. In *IEEE ICASSP*, pages 2092–2096, Alberta, Canada, April 15–20, 2018.

[44] Y. Yousfi, J. Butora, Q. Giboulot, and J. Fridrich. Breaking ALASKA: Color separation for steganalysis in JPEG domain. In R. Cogranne and L. Verdoliva, editors, *The 7th ACM Workshop on Information Hiding and Multimedia Security*, Paris, France, July 3–5, 2019. ACM Press.

[45] Y. Yousfi, J. Butora, E. Khvedchenya, and J. Fridrich. Imagenet pre-trained cnns for jpeg steganalysis. In *IEEE International Workshop on Information Forensics and Security*, New York, NY, December 6–11, 2020.

[46] Y. Zhou, W. W. Y. Ng, and Z. He. Effects of double jpeg compression on steganalysis. In *International Conference on Wavelet Analysis and Pattern Recognition*, pages 106–112, 2012.

[47] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109(1):43–76, 2021.