Fighting the Reverse JPEG Compatibility Attack: Pick your Side

Jan Butora
Patrick Bas
jan.butora@cnrs.fr
patrick.bas@cnrs.fr
Univ. Lille, CNRS, Centrale Lille, UMR 9189 CRIStAL
Lille, France

ABSTRACT

In this work we aim to design a steganographic scheme undetectable by the Reverse JPEG Compatibility Attack (RJCA). The RJCA, while only effective for JPEG images compressed with quality factors 99 and 100, was shown to work mainly due to change in variance of the rounding errors after decompression of the DCT coefficients, which is induced by embedding changes incompatible with the JPEG format. One remedy to preserve the aforementioned format is utilizing during the embedding the rounding errors created during the JPEG compression, but no steganographic method is known to be resilient to RJCA without this knowledge. Inspecting the effect of embedding changes on both variance and mean of decompression rounding errors, we propose a steganographic method allowing resistance against RJCA without any side-information. To reach this goal, we propose a distortion metric making all embedding changes within a DCT block dependent, resulting in a lattice-based embedding. Then it turns out it is enough to cleverly pick the side of the (binary) embedding changes through inspection of their effect on the variance of decompression rounding errors and simply use constant costs in order to enforce their sparsity across DCT blocks. To increase security against detectors in the spatial (pixel) domain, we show an easy way of combining the proposed methodology with steganography designed for spatial domain security, further improving the undetectability for quality factor 99. The improvements over existing non-informed steganography are up to 40% in terms of detector's accuracy.

CCS CONCEPTS

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

KEYWORDS

Steganography, RJCA, rounding errors, binary embedding

IH&MMSec '22, June 27–28, 2022, Santa Barbara, CA, USA

ACM ISBN 978-1-4503-9355-3/22/06…$15.00
https://doi.org/10.1145/3531536.3532955

ACM Reference Format:
Jan Butora and Patrick Bas. 2022. Fighting the Reverse JPEG Compatibility Attack: Pick your Side. In *Proceedings of the 2022 ACM Workshop on Information Hiding and Multimedia Security (IH&MMSec '22), June 27–28, 2022, Santa Barbara, CA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3531536.3532955

1 INTRODUCTION

Steganalysis, the art of detecting hidden information in digital images, has been in the past dominated with machine learning classifiers. Because of the high dimensionality images provide, high dimensional features are typically computed from the images and these features are then fed into the classifiers. Such features include JRM [24], DCTR [20], and GFR [28] in the JPEG domain, and SRM [17] in the spatial domain. With the increase of deep learning over the last decade, the steganalysis is lately mainly performed with Convolutional Neural Networks (CNNs) because of their incredible performance in the pixel (spatial) domain. Some of the early CNNs specifically designed for steganalysis of images include YeNet [31], XuNet [30], Yedroudj-Net [32], and SRNet [5]. However, during the recent ALASKA steganalysis competitions, it was shown that CNNs designed for different computer vision tasks can also be very successful in detecting steganography [9, 11, 12, 33, 34].

Another, more statistical approach to steganalysis aims to build a robust stochastic model for cover or stego image elements. Steganalysis is then performed by observing changes to these underlying models. One of the latest findings of statistical JPEG steganalysis, the so-called Reverse JPEG Compatibility Attack (RJCA) [6, 10], uses rounding errors of decompressed pixels for detecting even very slight modifications. It was shown that these rounding errors can be approximated by a Gaussian distribution wrapped between values $-1/2$ and $1/2$ with zero mean and some variance, while the variance grows rapidly by doing any modification on the original DCT coefficients. This increase of variance can be thus used for a reliable detection. Unfortunately, the variance also depends on the quantization matrix used for the JPEG compression, therefore the attack is limited only to the highest quality factors, where the quantization matrices still contain mainly ones. Nevertheless, the attack is extremely accurate and can detect with accuracy above 99% even very short secret messages. While the detection can be somewhat avoided for small payloads with the knowledge of side-information in form of the DCT rounding errors, to the best of our knowledge there is not a steganographic method effective against RJCA that does not use side-information.

In this paper, we design an embedding scheme that preserves the variance of spatial domain rounding errors and through experimentation we verify that this indeed leads to a more successful deception of the RJCA. However, since the embedding costs of this method do not consider distortion created in the pixel domain, the detection of this method at quality factor 99 can be more reliable by steganalyzing in the spatial domain. We show that this can be limited by combining the proposed costs with other steganographic costs designed for undetectability in the pixel domain.

The rest of the paper is organized as follows: Section 2 introduces the notation, typical optimization problems of steganography and the Reverse JPEG Compatibility Attack. In Section 3, we describe the dataset and two types of detectors used to benchmark security. The description of the proposed method follows in Section 4, which is then evaluated in the rounding error domain as well as in spatial domain. Finally, the paper is concluded in Section 5.

2 PRELIMINARIES

Boldface symbols are reserved for matrices and vectors with element-wise 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 to index 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

$$d_{kl} = \mathrm{DCT}_{kl}(\mathbf{x}) = \sum_{i,j=0}^{7} f_{kl}^{ij} x_{ij}, \quad 0 \le k, l \le 7,$$

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

$$f_{kl}^{ij} = w_k w_l / 4 \cos \pi k(2i+1)/16 \cos \pi l(2j+1)/16,$$

$w_0 = 1/\sqrt{2}$, $w_k = 1$, $0 < k \le 7$, 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

$$y_{ij} = \mathrm{DCT}_{ij}^{-1}(\mathbf{c} \odot \mathbf{q}) \triangleq \sum_{k,l=0}^{7} f_{kl}^{ij} q_{kl} c_{kl}, \quad y_{ij} \in \mathbb{R}.$$

To obtain the final decompressed image, y_{ij} are rounded to integers $x_{ij} = [y_{ij}]$. The spatial domain rounding errors, which are the main focus of this work, are $e_{ij} = y_{ij} - x_{ij}$ and we will often refer to this representation as the *error domain*.

2.1 Embedding Strategies

In today's content-adaptive steganography, two main approaches for determining optimal change rates are used. First, the most popular strategy tries to minimize an expected distortion

$$\sum_i \beta_i^+ \rho_i^+ + \beta_i^- \rho_i^-, \tag{1}$$

such that we communicate the desired relative payload α in bpnzac (bits per non-zero AC DCT coefficient):

$$\sum_i H_3(\beta_i^+, \beta_i^-) = \alpha, \tag{2}$$

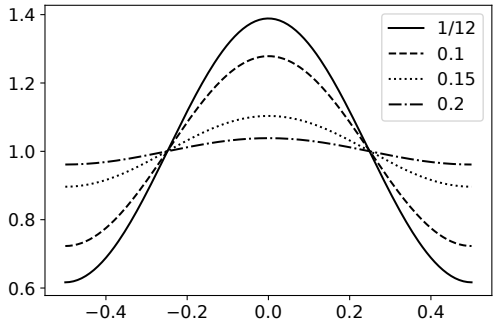

Figure 1: Probability density function of the Wrapped Gaussian distribution $\mathcal{N}_W(0, s)$ for different values of s.

where $H_3(\beta^+, \beta^-)$ is the ternary entropy function,

$$H_3(\beta^+, \beta^-) = -(1-\beta^+-\beta^-)\log(1-\beta^+-\beta^-) - \beta^+ \log \beta^+ - \beta^- \log \beta^-,$$

$\beta_i^\pm$ and $\rho_i^\pm$ are change rates and embedding costs of changing the i-th pixel/DCT coefficient by ±1. Such optimization problem is used in many popular steganographic schemes, such as J-UNIWARD [21], UERD [19], in JPEG domain, and HILL [25], S-UNIWARD [21] in spatial domain. Note that typically these steganographic algorithms yield symmetric costs, $\rho_i^+ = \rho_i^-$, which inherently leads to symmetric change rates $\beta_i^+ = \beta_i^-$. This is a potential drawback of such schemes, because it was shown many times that asymmetric costs can produce better security [2–4, 29].

Interestingly, Ker et al. [23] argued that the embedder should minimize an alternative distortion

$$\sum_i \beta_i^2 \rho_i, \tag{3}$$

where β_i is a (symmetric) probability of changing the i-th cover element by +1 or -1. However they showed that minimizing this quantity with heuristically designed costs is more detectable than using the linear distortion (1). In [7], it was claimed that this unexpected behavior is most likely due to the fact that the distortion costs are closely related to statistical detectability only for some fixed 'design' payload.

The other possible method of embedding is minimizing a quantity related to statistical detectability, which usually boils down to minimizing a deflection coefficient $\delta = \sum_i \beta_i^2 I_i$,where I_i is the steganographic Fisher information [15, 22], while using the same payload constraint (2). Such optimization can be found for example in MiPOD [27] or its JPEG version J-MiPOD [13].

This, obviously, can be thought of as using the Fisher information as a cost in (3). Since we aim in this work at avoiding RJCA by changing some basic statistical features of the images, we consider both distortion functions (1), (3) when designing a steganographic scheme in Section 4, as there is a reason to believe the distortion we are going to measure is closely linked to statistical detectability.

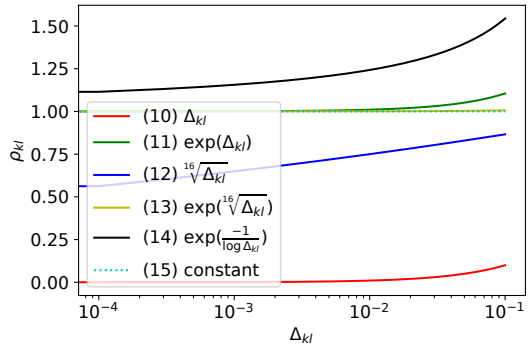

Figure 2: Profiles of the tested cost functions.

Note that to quickly minimize equation (3) for binary embedding with the payload constraint

$$\sum_i H_2(\beta_i) = \alpha,$$

where $H_2(\beta) = H_3(\beta_i, 0)$ is the binary entropy function, a solution is found using the method of Lagrange multipliers. That requires solving for every DCT coefficient

$$\beta_i \rho_i = \lambda \log \frac{1-\beta_i}{\beta_i} \quad \forall i.$$

This we can solve numerically with a look-up table and a binary search over the Lagrange multiplier $\lambda > 0$. Details of this optimization can be found in [18], where the optimization is done for ternary embedding. Note that the only difference compared to the ternary optimization proposed in [18] (Section 4), is tabulating the inverse function to $y = x \log(x-1)$ instead of $y = x \log(x-2)$.

2.2 RJCA

Since the main goal of this paper is countering the power of the RJCA, we first recall in here the statistical models in the error domain that were derived in the original publication. For $X \sim \mathcal{N}(\mu, s)$ with $\mu \in \mathbb{Z}$, the rounding error $X - [X]$ follows a Wrapped Gaussian distribution $X - [X] \sim \mathcal{N}_W(0, s)$, where the probability density function (pdf) $v(x; s)$ of the Wrapped Gaussian is given by

$$v(x; s) = \frac{1}{\sqrt{2\pi s}} \sum_{n \in \mathbb{Z}} \exp\left(-\frac{(x+n)^2}{2s}\right), \tag{4}$$

with $-1/2 \le x < 1/2$. We would like to point out that the variance s is variance of the underlying Gaussian distribution before folding into interval $[-1/2, 1/2)$. If one was to compute the variance of the Wrapped Gaussian distribution, it would be smaller than the original variance s, due to the folding. For simplicity, in the following equations we only use the variance of the Gaussian before folding, as it is easier to use in the pdf (4).

It was shown [6] that the rounding errors e_{ij} of a cover image follow a Wrapped Gaussian distribution

$$e_{ij} \sim \mathcal{N}_W(0, s_{ij}),$$

with variance of the Gaussian ditribution before folding

$$s_{ij} = \frac{1}{12} \sum_{k,l=0}^{7} (f_{kl}^{ij})^2 q_{kl}^2.$$

It is straightforward to verify that for standard quantization matrix at QF 99, these variances range between 0.105 and 0.204, while for QF 100, the variances are exactly $1/12$ for every $i, j = 0, \dots 7$, because the DCT transform is an orthonormal basis and all the quantization steps are equal to 1. We can see the impact of the variance of on Wrapped Gaussian distribution in Figure 1, where we show its probability density function for different variances. We see a clear evolution towards uniform distribution with increasing variances. This is important, because it was also shown that the rounding errors $e_{ij}^{(S)}$ of stego images follow the Wrapped Gaussian distribution with increased variance

$$e_{ij}^{(S)} \sim \mathcal{N}_W(0, s_{ij} + r_{ij}),$$

where the increase of variance depends on the size of the secret message:

$$r_{ij} = \sum_{k,l=0}^{7} (f_{kl}^{ij})^2 q_{kl}^2 (\beta_{kl}^{+} + \beta_{kl}^{-}).$$

To derive these models, two main assumptions were made. First, the rounding errors in the DCT domain are mutually independent and follow uniform distribution between $-1/2$ and $1/2$, which is in many cases a reasonable assumption. Second, it was assumed that the embedding changes are mutually independent and also independent of the DCT rounding errors. While we cannot do anything about the first assumptions, since from a given JPEG image, we cannot reconstruct the DCT rounding errors, we can violate the second assumption in order to make the impact on variance smaller. This we target in Section 4.1.1.

3 BENCHMARKING SETUP

This section describes the datasets as well as the detectors used for evaluating security. These datasets are used in Section 4 to progressively adjust the embedding costs with a feedback from the deep learning detectors.

3.1 Dataset

To experimentally verify our results, we chose the popular BOSSbase 1.01 [1] dataset, consisting of 10,000 uncompressed, grayscale images of size 512×512. The dataset is then randomly split into training, validation, and testing sets of sizes 7,000, 1,000, and 2,000 respectively. Finally, all the images are JPEG compressed using Python3 library PIL with quality factors (QFs) 99 and 100. We then embedded the images with various payloads using J-UNIWARD, SI-UNIWARD [21], and the proposed method called SVP (**S**patial **V**ariance **P**reserving), described in Section 4. The results of SI-UNIWARD are only included as the steganographer's best scenario and should not be directly compared to non-informed SVP. Note that in order to compute the side-information for SI-UNIWARD, we

	Binary	Ternary	Quadratic
SVP (10)	0.9913	0.9995	0.9825
SVP (10)-L	0.9995	0.9998	0.9793
SVP (11)	0.8770	0.9870	0.8993
SVP (11)-L	0.8726	0.9695	0.8631
SVP (12)-L	0.9305	0.9395	0.9125
SVP (13)-L	0.8673	0.9595	0.8780
SVP (14)-L	0.8733	0.9448	0.8891
Constant-L	**0.8431**	0.9885	**0.8431**
JUNI	N/A	0.9980	N/A
SI-UNI	0.6887	0.6859	N/A

Table 1: Accuracy with e-B0 at 0.05 **bpnzac. L means that lattices were used for embedding.**

cannot compress the images with PIL, so instead we compressed the images manually with Scipy package. [1]

3.2 Detectors

For evaluation of the steganographic security, we use two types of detectors. First is EfficientNet-B0 [26], initialized with weights pre-trained on the ImageNet dataset [14]. This detector is then trained in the error domain e_{ij}, and is thus denoted as e-B0, similarly as in [6]. The second detector we use, in order to verify that our method does not introduce detectable artifacts in the pixel domain, is the JIN-SRNet [8] - SRNet [5] pre-trained on ImageNet embedded with J-UNIWARD. Since pre-training of both detectors is executed on color images, they expect three-channel inputs. Thus, we simply replicated the grayscale representation in all three *RGB* channels before feeding the images into the networks. Both detectors were using mixed precision training with 32 images in every mini-batch and OneCycle learning rate (LR) scheduler with maximum LR 10^{-3} at epoch 3. The JIN-SRNet was trained for 50 epochs, while e-B0 only for 15 epochs. Rest of the hyperparameters was kept as in [8], Section 3.2.1. Because the detectability in spatial domain is typically much smaller than in the error domain, we use larger payloads for the spatial domain detector.

4 SPATIAL VARIANCE PRESERVING

We have seen in Section 2.2 that typical steganography increases variance of the Wrapped Gaussian distribution which the rounding errors e_{ij} follow. Figure 1 shows that increasing the variance by even very small amounts rapidly changes the distribution towards uniform. On the other hand, it is easy to realize that decreasing the variance would lead to much narrower distribution, basically Gaussian distribution with small variance (almost not affected by folding). The proposed method which we call *Spatial Variance Preserving* (SVP) therefore aims, as the name suggests, at preserving the variance of (spatial) rounding errors e_{ij} in every 8×8 block.

Let $\mathbf{1}_{kl}$ denote an 8×8 block of zeros, where there is a 1 in the DCT mode (k, l) and $\mathbf{z}_{kl}$ the pixel block (after decompression) of a stego image embedded with $\boldsymbol{\eta}_{kl}$

$$\mathbf{z}(\boldsymbol{\eta}_{kl}) = \mathrm{DCT}^{-1}((\mathbf{c} + \boldsymbol{\eta}_{kl}) \odot \mathbf{q}), \tag{5}$$

where $\boldsymbol{\eta}_{kl} \in \{-\mathbf{1}_{kl}, \mathbf{1}_{kl}\}$. Having $\mathbf{e}$ - the block of rounding errors of the cover image, denote $\mathbf{e}(\boldsymbol{\eta}_{kl})$ - the block of rounding errors after embedding $\boldsymbol{\eta}_{kl}$ into the DCT coefficients:

$$\mathbf{e} = \mathbf{y} - [\mathbf{y}], \tag{6}$$

$$\mathbf{e}(\boldsymbol{\eta}_{kl}) = \mathbf{z}(\boldsymbol{\eta}_{kl}) - [\mathbf{z}(\boldsymbol{\eta}_{kl})]. \tag{7}$$

The variance of cover and stego spatial rounding errors will be denoted as $\sigma = \mathrm{Var}[\mathbf{e}]$ and $\sigma(\boldsymbol{\eta}_{kl}) = \mathrm{Var}[\mathbf{e}(\boldsymbol{\eta}_{kl})]$. Let

$$\Delta_{kl}(\eta) = |\sigma - \sigma(\boldsymbol{\eta}_{kl})|, \tag{8}$$

denote the change in variance of a block of rounding errors when changing the DCT coefficient in mode (k, l) by $\eta \in \{-1, +1\}$. After inspecting $\Delta_{kl}(\eta)$ for every value of η, we fix the embedding polarity in order to preserve the cover variance as closely as possible, by only allowing a change

$$\Delta_{kl} = \min_{\eta} \Delta_{kl}(\eta). \tag{9}$$

Based on (9), we argue that in order to preserve the variance as much as possible, we should avoid making the embedding changes in the opposite direction. While we do not necessarily need to binarize the embedding scheme, allowing ternary embedding results in making embedding changes that modify the variance too much and we can see in Section 4.1 that this leads to severe security deterioration. The ternary embedding that we use for this method implicitly uses asymmetric costs driven by distortions $\Delta_{kl}(-1)$ and $\Delta_{kl}(+1)$.

4.1 Error Domain Distortion

Having some basic measure of the embedding distortion (9), we investigated several non-linear transformations of it, as there is no theoretical evidence that our distortion metric is optimal. We considered five transformations as the final embedding costs:

$$\rho_{kl} = \Delta_{kl}, \tag{10}$$

$$= \exp(\Delta_{kl}), \tag{11}$$

$$= \sqrt[n]{\Delta_{kl}}, \tag{12}$$

$$= \exp(\sqrt[n]{\Delta_{kl}}), \tag{13}$$

$$= \exp\left(\frac{-1}{\log_{10} \Delta_{kl}}\right). \tag{14}$$

Note that for ternary embedding, we would use $\Delta_{kl}(\pm 1)$ instead of Δ_{kl}. These cost functions are visualized in Figure 2. Notice that the x-axis is in log-scale and only goes to 0.1, because the distortion we can introduce is upper bounded by the value $1/12$, since this is the variance of a uniform noise, which is the limiting case for the wrapped Gaussian distribution [6]. During our security evaluation, we found that $n = 16$ was performing the best and for this reason, it was kept at this value.

As mentioned in Section 2.1, we implemented three versions for every cost function under investigation. We minimized the linear distortion (1) with ternary and binary embedding, where the binary case only allows embedding changes given by Δ_{kl}. The

[1] The compression and SVP embedding scripts are available at https://janbutora.github.io/downloads/

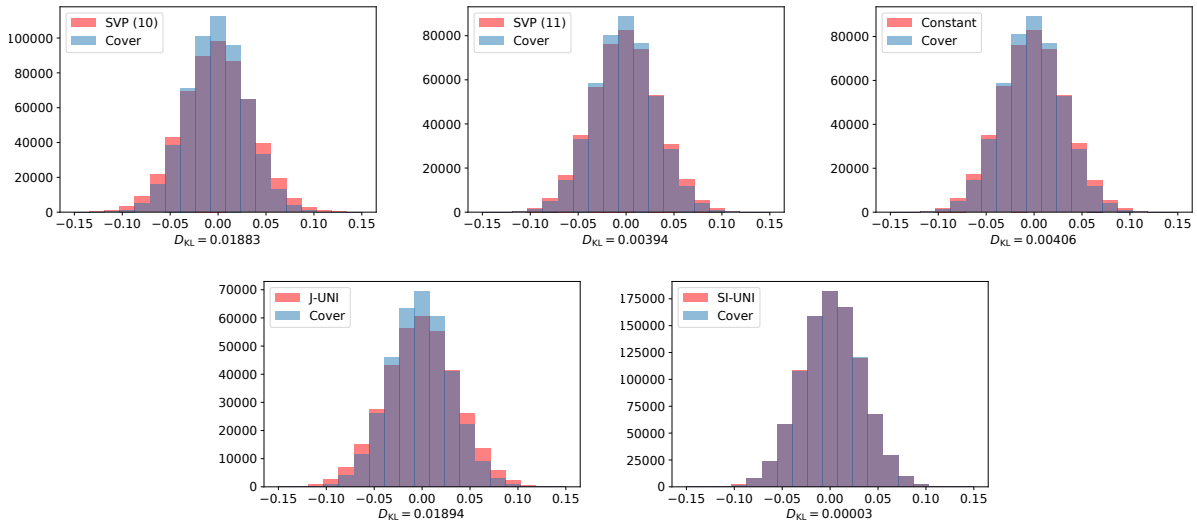

Figure 3: Histogram of per-block means of errors $\mathbb{E}[e]$ of different methods embedded at 0.05 bpnzac across 500 images compressed with QF 100. Only embedded blocks are considered. The number of bins is very small due to their sparsity. Kullback-Leibler divergence D_{KL} between cover and stego distributions is shown below every figure.

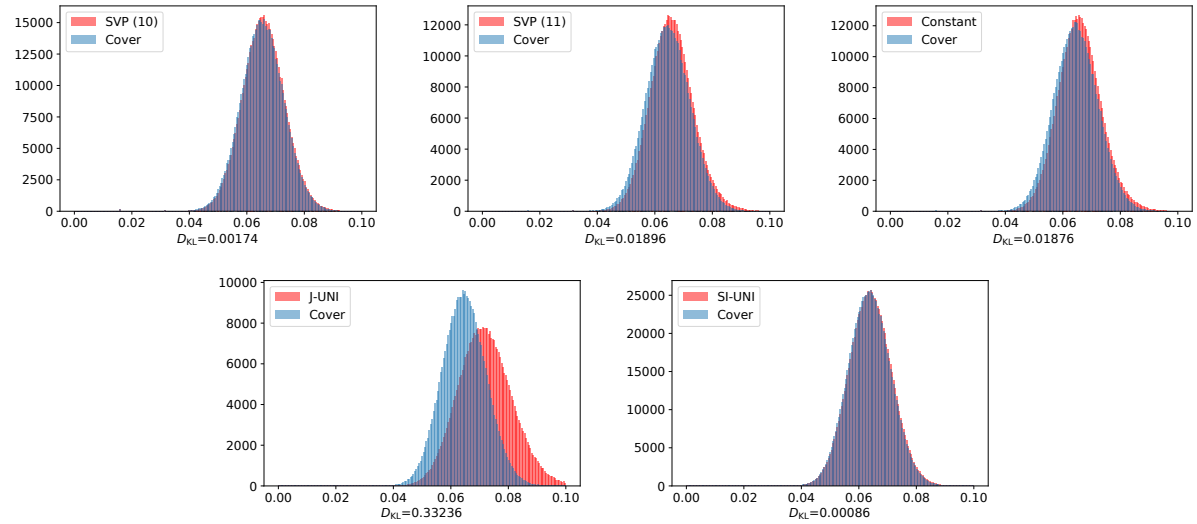

Figure 4: Histogram of per-block variances of errors $\mathrm{Var}[e]$ of different methods embedded at 0.05 bpnzac across 500 images compressed with QF 100. Only embedded blocks are considered. Kullback-Leibler divergence D_{KL} between cover and stego distributions is shown below every figure.

third approach we tested minimizes the quadratic distortion (3) with the binary embedding only. These three embedding strategies will be referred to as *binary*, *ternary*, and *quadratic*. To establish which embedding policy performs the best, we conduct following tests only for images compressed with QF 100 embedded with fixed payload 0.05 bpnzac.

4.1.1 Dependence of Embedding Changes. Because changing one DCT coefficient in an 8×8 block affects every pixel in the block, we cannot simply compute every embedding cost from the cover image, because the distortions (8) created by the embedding changes are not independent. To verify this, we computed all the costs from the cover images at once, embedded the images in our database with payload 0.05 bpnzac, and trained EfficientNet-B0 as explained in Section 3.2. As expected, even the most secure embedding approach was still very detectable, with accuracy 0.9825, see first row in Table 1. Instead, we implemented a lattice approach, in which we

divide the image into 64 non-overlapping lattices (one lattice per DCT mode) and perform the embedding sequentially on randomly selected lattice. Note that this is violating one of the assumptions made in RJCA, as advertised in Section 2.2.

There is one potential issue with such an approach - how much payload do we allocate to every lattice? To keep things simple, we decided to test uniform payload across latices. Note that the embedding can be implemented quite efficiently with parallel computations, since the costs are computed on every block separately. For every block, we only need to perform 2×64 decompressions to test all possible embedding configurations.[2]

Unfortunately, using lattices did not bring much of an improvement to the security (see second row of Table 1). We hypothesize that the costs are bad to begin with. We thus repeated these two experiments with the exponential cost function (11). As can be seen from Table 1, not only does taking the exponential of the value Δ_{kl} increase the security by $\sim$ 10%, but using lattices during the embedding brings additional noticeable improvement.

To verify that embedding the same payload into every lattice does not increase detectability, we implemented the quadratic version of the exponential costs (11), but this time we allocated the payload based on J-UNIWARD change rates - for a given image, we would compute optimal J-UNIWARD's change rates β_i and note down how much payload gets embedded into every mode. The accuracy of this method was 0.9008, which is 4% higher than with uniform payload across lattices. This might be somewhat counter-intuitive, but it should not be too surprising since the steganalysis is not performed in the spatial domain (spatial domain steganalysis is investigated in Section 4.2). To this end, we keep the uniform payload for the rest of our experiments.

Now we are finally ready to compare different cost functions, defined in equations (10)-(14), in terms of immunity against the RJCA. In Table 1, we show the accuracy of correct classification, when embedding different algorithms at 0.05 bpnzac. On one hand, J-UNIWARD does not take the error domain into account at all, hence it is detectable with almost 100% accuracy. On the other hand, the side-informed SI-UNIWARD taking advantage of the DCT rounding errors during embedding (breaking another RJCA's assumption), provides the best security. We can also distinctly see that using ternary embedding makes the steganography much more detectable, even though it decreases the amount of embedding changes. This was also expectable, since ternary embedding allows much bigger distortion of the rounding error variance. The most naive way of using the impact on the rounding error variance (10) is very detectable, but we can clearly observe that transforming Δ_{kl} brings decent benefit in terms of security. Additionally, we have already mentioned that using 64 non-overlapping lattices for embedding, which introduces dependencies between embedding changes, is providing better security.

4.1.2 Sparsity. In the end, the most secure among the tested functions is the exponential (11). We believe that this is due to the uniform profile of the cost function for very small values of Δ_{kl} (up to 10^{-2}), which creates sparsity in the embedding changes, meaning the embedding prefers making very few changes in every DCT block. We would like to remind the reader that using constant

[2]During embedding of every lattice, only 2 possible embedding changes are considered.

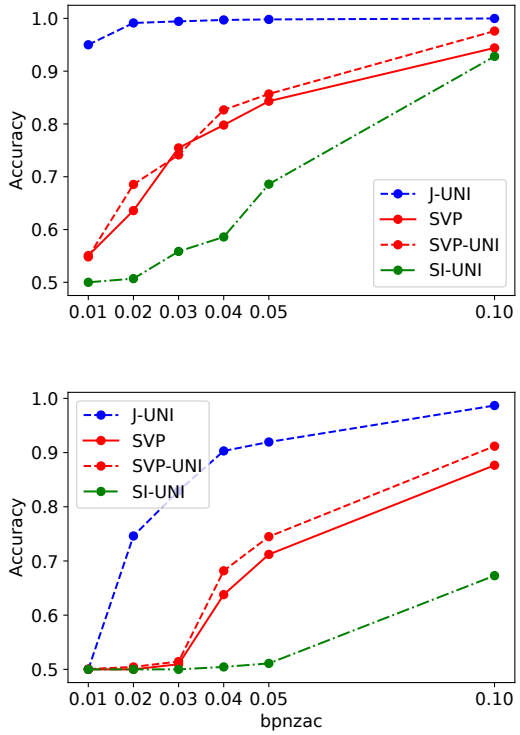

Figure 5: Accuracy with e-B0 (error domain) at payloads between 0.01 and 0.1 bpnzac. Top: QF 100, bottom: QF 99.

costs enforces sparsity, because minimizing (1) is equivalent to minimizing expected L_1 distortion between the cover and stego image. We think that embedding overly adaptive to the error variances negatively impacts the mean of the block's rounding errors, which could lead to increased detectability. This makes intuitive sense, because for RJCA, embedding changes were modeled as zero-mean random variables, allowing us to detect steganography by inspecting changes in variance, but obviously the zero-mean assumption is wrong for a binary embedding performing many changes within the same block. This is experimentally confirmed in Figures 3,4, where we show histograms of empirical block means $\mathbb{E}[\mathbf{e}]$ and variances $\mathrm{Var}[\mathbf{e}]$. As discussed, we see that the SVP methods preserve the variances very nicely, but the means are modified much more for the linear cost function (10). Of course this contributes to the detectability, since CNNs focus on changes in both, variance and mean. Another interesting observation is that J-UNIWARD, the only algorithm not dealing with any rounding errors, disrupts both mean and variance. Lastly, even though SI-UNIWARD makes the most embedding changes among the tested steganographic schemes[3], it perfectly preserves mean and variance in the error domain e_{ij}, which we believe is the main reason of its superior security.

4.1.3 Pick your side. To verify that sparsity of embedding changes (keeping the mean distortion low), as discussed in the previous

[3]This can be seen from the y-axis scale in Figures 3,4.

section, is more important than perfect variance preservation, we implement one last cost function, namely the constant cost function

$$\rho_i^C = c, \quad c > 0. \tag{15}$$

With these costs, we only have to pick the right side for the binary embedding changes, otherwise the embedding scheme would of course be terribly detectable. As for all the previous methods, we pick the side having smaller impact on the rounding error variance based on (9), while using the sequential embedding over the 64 lattices. Notice that minimizing the linear (1) and quadratic (3) distortions with constant costs yields the same embedding probabilities, so in this sense, the two embedding strategies are the same. To verify that this embedding method preserves both mean and variance of the rounding errors, we include it in Figures 4,3. Visually and using the reported KL divergence in the figures, we can observe that this method has a very similar impact on the histograms of rounding error mean and variance as the cost function (11). However, Table 1 show us that the constant costs are 2% more secure against the RJCA, making it the most secure non-informed algorithm available. One last interesting observation from Table 1 can be made, the ternary version of this embedding method (having symmetric costs) is detectable with almost 99% accuracy, verifying that constant costs by themself are not a good idea, but picking the right embedding polarity in a sequential way makes a huge impact on security.

For the rest of the experiments, we only consider the constant cost function (15) with embedding change polarities decided by (9), as it provides the best security. For simplicity, we refer to this method in the rest of this work simply as SVP. The results for QFs 100 and 99 with e-B0 are visualized in Figure 5. We can see large security improvement compared to J-UNIWARD ranging between $6-40\%$ for QF 100 and between $24-40\%$ for QF 99 in terms of accuracy. However, as advertised earlier, we have not considered distortion in the spatial domain yet, thus we investigate this in the following section.

4.2 Spatial Domain Distortion

To verify that the embedding changes of the proposed SVP scheme do not create easily detectable artifacts in the spatial domain, we train the JIN-pre-trained SRNet as explained in Section 3.2 across various payloads. We can observe from Figure 6 that even though the SVP method is only slightly more detectable at QF 100 than J-UNIWARD, it is substantially more detectable at QF 99. Note this does not imply that the SVP scheme would be overall less secure than J-UNIWARD, because the spatial domain experiments were performed on much larger payloads (for detector convergence reasons). The very high detectability on QF 99 could be explained by the fact that higher frequency DCT modes, having larger quantization steps, introduce much more detectable distortion if the embedding changes are not performed carefully. And this was in fact not considered in the SVP method.

To fix this, we propose one of many possible remedies. Since J-UNIWARD's costs are well designed for the spatial domain distortion, we simply combined the SVP and J-UNIWARD's costs. Let ρ_{kl}^U be an embedding cost of SVP scheme and ρ_{kl}^J be corresponding (symmetric) cost coming from J-UNIWARD. Then the resulting cost

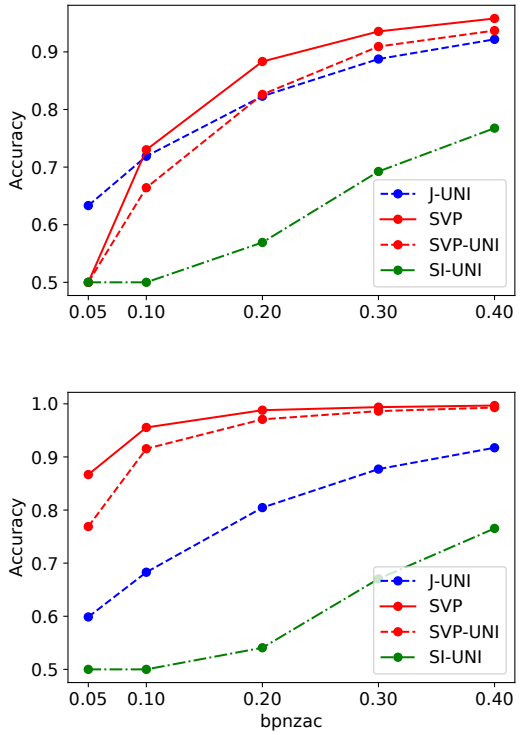

Figure 6: Accuracy with JIN-SRNet (spatial domain) at payloads between 0.05 and 0.4 bpnzac. Top: QF 100, bottom: QF 99.

$\rho_{kl}^{U,J}$ can be computed as

$$\rho_{kl}^{C,J} = \rho_{kl}^C \cdot \rho_{kl}^J. \tag{16}$$

Note that since ρ_{kl}^C are all equal for every DCT coefficient, but only allowing embedding changes causing the smaller distortion (9), therefore we can view them as ternary costs of changing the DCT coefficients by +1 or -1, where one of the costs is infinite (wet cost [16]). Combining the costs in (16) thus leads to binarization of the J-UNIWARD algorithm, since one of the costs stays unchanged, while the cost of opposite direction gets updated to infinity. In other words, we preserve the costs of J-UNIWARD, while only allowing the changes in direction which better preserves the error variance (9). Importantly, the embedding polarities are updated after considering each lattice, while the costs ρ_{kl}^J are computed once from the cover image, since recomputing J-UNIWARD costs for every lattice would be extremely slow. Even though the linear and quadratic embedding strategies are the same for SVP method, they are not the same anymore in this situation. In our experiments we tested both embedding strategies with the quadratic one performing better, therefore it was used in the reported experiments. We will refer to this embedding method as SVP-UNI. As can be seen from Figure 6, modifying the costs in the proposed way does improve security in the spatial domain. However, as a result, the scheme is slightly more detectable in the error domain (see Figure 5).

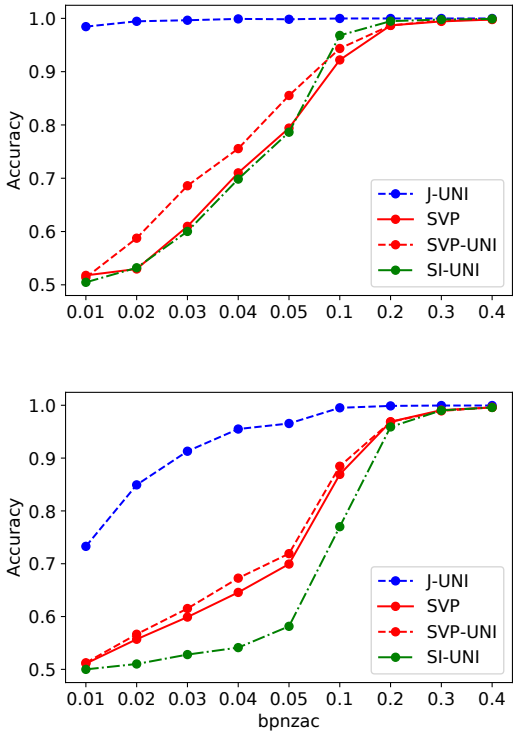

Figure 7: Accuracy with eY-B0 (error+spatial domain) at payloads between 0.01 and 0.4 bpnzac. Top: QF 100, bottom: QF 99.

4.3 Multi-Domain Steganalysis

For the last experiment we conduct in this paper, we would like to remind the reader of what was observed in [6], specifically using a 2-channel inputs for the network detector performs the best: 1 channel for the spatial rounding errors and the other channel for decompressed pixels. Therefore we also tested this strategy in our setting. We trained EfficientNet-B0 as explained in Section 3.2, but we provide 2-channel input as we just described. This version of the detector is denoted as eY-B0. To have the channels on a similar scale, we divide the pixel channel by 255. Also, to make this compatible with the existing EfficientNet structure, we included a 1×1 convolution layer mapping 2 input dimensions into 3 dimensions as a preprocessing step before entering the network. Because it seems it is difficult for the network to figure out the relationship between the two input channels for small payloads (network would not converge), we first train the eY-B0 on payload 0.2 bpnzac for 20 epochs and use curriculum training from the best checkpoint to every other tested payload, while the curriculum training is stopped after 15 epochs. As expected, this detector provides the best overall results for all tested schemes and payloads, with the exception of detecting SVP(-UNI) methods embedded with payloads below 0.05 bpnzac at QF 100, where the e-B0 provides higher detection. We strongly believe that this is due to the preprocessing step we added to the eY-B0 structure, which needs to be tuned longer from initial random weights, and that longer training would make the eY-B0 at least as efficient as the e-B0, but we do not include such experiments due to limited time.

Surprisingly, the SVP method at QF 100 is for payloads above 0.05 bpnzac more secure than the side-informed SI-UNIWARD. This phenomenon probably happens, because the embedding changes of SI-UNIWARD are more correlated with spatial domain distortion, thus making it more detectable in the error domain. To verify this, we can observe the steep increase of detectability of SI-UNIWARD at payload 0.1 bpnzac, QF 100 in Figure 5.

5 CONCLUSIONS

In this paper, we developed a steganographic method SVP (Spatial Variance Preserving) able to resist the Reverse JPEG Compatibility Attack (RJCA). We designed a distortion metric, which measures disturbance in variance of spatial domain rounding errors caused by embedding changes in the DCT domain. With such distortion metric, we showed that allowing only binary instead of ternary embedding changes lead to better security, because ternary embedding can disturb the variances too much. Next, we gained even better security through allowing interactions between embedding changes. This was achieved by splitting the image into 64 non-overlapping lattices (one lattice per every DCT mode) and sequentially updating the proposed distortion in every lattice. Then we have observed that if we try to be overly adaptive to the error variance, we disturb the mean of the errors, resulting in increased detectability. While this can potentially be also addressed in the distortion metric, it would require much more experimentation and is thus left for future work. Instead we point out, that by imposing sparsity of embedding changes, we can better preserve the mean of the rounding errors. Based on this, constant costs were shown to provide best security, as long as we pick the embedding change polarity with feedback from the proposed distortion metric.

While the proposed method works reasonably well only at very small payloads, such as 0.05 bpnzac, it is currently the best a steganographer can do without side-informed steganography, but we have seen that even with the side-information, security is greatly limited by the payload size. The improvements over existing non-informed steganography in terms of resistance against RJCA range between 6-40% for quality factors 99 and 100. In spatial domain, the proposed method is very detectable for quality factor 99. To further improve the security in spatial domain, we show that binarizing an existing embedding scheme with the feedback from the proposed distortion metric achieves better security, even though it results in slightly higher detectability in the error domain. Finally, a multi-domain detector was tested, which suggests that keeping the constant costs is more secure than combining with existing steganography.

In our future work, we plan to investigate a statistical approach for preserving the model of the rounding errors and its effect on steganographic security. Since the side-informed steganography provided the best security and kept the most accurate model of the errors, we believe that knowledge of the side-information is necessary while deriving the underlying model.

ACKNOWLEDGMENTS

This work was granted access to the HPC resources of IDRIS under the allocation 20XX-AD011012855 made by GENCI.

REFERENCES

[1] 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.

[2] S. Bernard, P. Bas, J. Klein, and T. Pevný. Explicit optimization of min max steganographic game. *IEEE Transactions on Information Forensics and Security*, 16:812–823, 2021.

[3] S. Bernard, P. Bas, T. Pevný, and J. Klein. Optimizing additive approximations of non-additive distortion functions. In D. Borghys and P. Bas, editors, *The 9th ACM Workshop on Information Hiding and Multimedia Security*, Brussels, Belgium, June 21–25, 2021. ACM Press.

[4] S. Bernard, T. Pevný, P. Bas, and J. Klein. Exploiting adversarial embeddings for better steganography. 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.

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

[7] J. Butora, Y. Yousfi, and J. Fridrich. Turning cost-based steganography into model-based. In C. Riess and F. Schirrmacher, editors, *The 8th ACM Workshop on Information Hiding and Multimedia Security*, Denver, CO, June 22–25, 2020. ACM Press.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] 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.

[13] R. Cogranne, Q. Giboulot, and P. Bas. Steganography by minimizing statistical detectability: The cases of jpeg and color images. In C. Riess and F. Schirrmacher, editors, *The 8th ACM Workshop on Information Hiding and Multimedia Security*, Denver, CO, 2020. ACM Press.

[14] 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.

[15] T. Filler and J. Fridrich. Fisher information determines capacity of ϵ-secure steganography. In S. Katzenbeisser and A.-R. Sadeghi, editors, *Information Hiding, 11th International Conference*, volume 5806 of Lecture Notes in Computer Science, pages 31–47, Darmstadt, Germany, June 7–10, 2009. Springer-Verlag, New York.

[16] J. Fridrich, M. Goljan, D. Soukal, and P. Lisoněk. Writing on wet paper. In E. J. Delp and P. W. Wong, editors, *Proceedings SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents VII*, volume 5681, pages 328–340, San Jose, CA, January 16–20, 2005.

[17] J. Fridrich and J. Kodovský. Rich models for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 7(3):868–882, June 2011.

[18] J. Fridrich and J. Kodovský. Multivariate Gaussian model for designing additive distortion for steganography. In *Proc. IEEE ICASSP*, Vancouver, BC, May 26–31, 2013.

[19] L. Guo, J. Ni, W. Su, C. Tang, and Y. Shi. Using statistical image model for JPEG steganography: Uniform embedding revisited. *IEEE Transactions on Information Forensics and Security*, 10(12):2669–2680, 2015.

[20] 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.

[21] 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.

[22] A. D. Ker. Estimating steganographic fisher information in real images. In S. Katzenbeisser and A.-R. Sadeghi, editors, *Information Hiding, 11th International Conference*, volume 5806 of Lecture Notes in Computer Science, pages 73–88, Darmstadt, Germany, June 7–10, 2009. Springer-Verlag, New York.

[23] A. D. Ker, T. Pevný, and P. Bas. Rethinking optimal embedding. In F. Perez-Gonzales, F. Cayre, and P. Bas, editors, *The 4th ACM Workshop on Information Hiding and Multimedia Security*, pages 93–102, Vigo, Spain, June 20–22, 2016. ACM Press.

[24] 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.

[25] B. Li, M. Wang, and J. Huang. A new cost function for spatial image steganography. In *Proceedings IEEE, International Conference on Image Processing, ICIP*, Paris, France, October 27–30, 2014.

[26] 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.

[27] V. Sedighi, R. Cogranne, and J. Fridrich. Content-adaptive steganography by minimizing statistical detectability. *IEEE Transactions on Information Forensics and Security*, 11(2):221–234, 2015.

[28] 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.

[29] W. Tang, B. Li, S. Tan, M. Barni, and J. Huang. CNN-based adversarial embedding for image steganography. *IEEE Transactions on Information Forensics and Security*, 14(8):2074–2087, 2019.

[30] G. Xu, H. Z. Wu, and Y. Q. Shi. Structural design of convolutional neural networks for steganalysis. *IEEE Signal Processing Letters*, 23(5):708–712, May 2016.

[31] J. Ye, J. Ni, and Y. Yi. Deep learning hierarchical representations for image steganalysis. *IEEE Transactions on Information Forensics and Security*, 12(11):2545–2557, November 2017.

[32] 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.

[33] 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.

[34] 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.