Forensics Through Stega Glasses: the Case of Adversarial Images

Benoît Bonnet[1][0000−0002−2569−6185], Teddy Furon[1][0000−0002−1565−765X], and Patrick Bas[2][0000−0003−0873−5872]

[1] Univ. Rennes, Inria, CNRS, IRISA, Rennes, France
benoit.bonnet@inria.fr, teddy.furon@inria.fr
[2] Univ. Lille, CNRS, Centrale Lille,UMR 9189, CRIStAL, Lille, France
patrick.bas@centralelille.fr

Abstract. This paper explores the connection between forensics, counter-forensics, steganography and adversarial images. On the one hand, forensics-based and steganalysis-based detectors help in detecting adversarial perturbations. On the other hand, steganography can be used as a counter-forensics strategy and helps in forging adversarial perturbations that are not only invisible to the human eye but also less statistically detectable. This work explains how to use these information hiding tools for attacking or defending computer vision image classification. We play this cat and mouse game using both recent deep-learning content-based classifiers, forensics detectors derived from steganalysis, and steganographic distortions dedicated to color quantized images. It turns out that crafting adversarial perturbations relying on steganographic perturbations is an effective counter-forensics strategy.

Keywords: Adversarial Examples · Steganography · Image Forensics.

1 Introduction

Adversarial examples is an emerging field in Information Forensics and Security, addressing the vulnerabilities of Machine Learning algorithms. This paper casts this topic to Computer Vision, and in particular, to image classification, and its associated forensics counter-part: the detection of adversarial contents.

A Deep Neural Network (DNN) is trained to classify images by the object represented in the picture. This is for instance the well-known ImageNet challenge encompassing a thousand of classes. The state-of-the-art proposes impressive results as classifiers now do a better job than humans with less classification errors and much faster timings. The advent of the AlexNet DNN in 2012 is often seen as the turning point of 'Artificial Intelligence' in Computer Vision. Yet, the recent literature of adversarial examples reveals that these classifiers are vulnerable to specific image modifications. The perturbation is often a weak signal barely visible to the human eyes. Almost surely, no human would incorrectly classify these adversarial images. This topic is extremely interesting as it challenges the 'Artificial Intelligence' qualification too soon attributed to Deep Learning.

The connection between adversarial examples and forensics/anti-forensics is obvious. First, adding an adversarial perturbation to delude a processing is an image manipulation per se and therefore detecting adversarial examples is a forensic task by itself. Second, techniques forging adversarial examples are also used to fool forensics detectors as proposed in [17][2]. In this case, the adversarial attack is a counter-forensics strategy to conceal an image manipulation.

Paper [28] makes the connection between adversarial examples and information hiding (be it watermarking or steganography). Both fields modify images (or any other type of media) in the pixel domain so that the content is moved to a targeted region of the feature space. That region is the region associated to a secret message in information hiding or to a wrong class in adversarial examples. Indeed, paper [28] shows that adversarial examples benefits from ideas proven efficient in watermarking, and vice-versa.

This paper contributes to the same spirit by investigating what both steganography and steganalysis bring to the the "cat-and-mouse" game of adversarial examples. There are two natural ideas:
Steganalysis aims at detecting weak perturbations in images. This field is certainly useful for the defender.
Steganography is the art of modifying an image while being non-detectable. This field is certainly useful for the attacker.

These two sides of the same coin allow to mount a defense and to challenge it in return, as done in other studies [6, 1, 39]. This paper aims at revealing the status of the game between the attacker and the defender at the time of writing, *i.e.* when both players use up-to-date tools: state-of-the-art image classifiers with premium steganalyzers, and best-in-class steganography embedders. As far as we know, this paper proposes three first time contributions:

- Assess robustness of recent models EfficientNet [35] and its robust version [43],
- Apply one state-of-the-art steganalyzer (SRNet [5]) for forensics purposes, *i.e.* to detect adversarial images,
- Use the best steganographic schemes to craft counter-forensics perturbations reducing the detectability: HILL [20] uses empirical costs, MiPod [31] models undetectability from a statistical point of view, while GINA [21, 42] synchronizes embeddings on color channels.

Section 2 reviews the connections between forensics, steganography, and adversarial examples. Our main contribution on counter-forensics and experimental results are detailed in Sect. 3 and 4.

2 Related Works

2.1 Steganalysis for forensic purposes

Steganalysis has always been bounded to steganography, obviously. Yet, a recent trend is to resort to this tool for other purposes than detecting whether an image

conceals a secret message. For instance, paper [27] claims the universality of SRM and LBP steganalyzers for forensic purposes detecting image processing (like Gaussian blurring, gamma correction) or splicing. The authors of [12] used this approach during the IEEE IFS-TC image forensics challenge. The same trend holds as well on audio forensics [23]. As for camera model identification, the inspiration from steganalysis (co-occurrences, color dependencies, conditional probabilities) is clearly apparent in [41].

This reveals a certain versatility of steganalysis. It is not surprising since the main goal is to model and detect weak signals. Modern steganalyzers are no longer based on hand-crafted features like SRM [14]. They are no more no less than Deep Neural Networks like Xu-Net [44] or SRNet [5]. The frontier between steganalysis and any two-class image classification problem (such as image manipulation detection) is blurred. Yet, these networks have a specific structure able to focus on weak signal detection: They avoid subsampling or pooling operations in order to preserve high frequency signals, they need large databases combined with augmentation techniques and curriculum learning [45].

However, this general-purpose strategy based on steganalysis method has some drawbacks. It lacks fine-grained tampering localization, which is often an issue in forensics [11]. Paper [8] goes a step further in the cat-and-mouse game with an counter-forensic method: knowing that the defender uses a steganalyzer, the attacker modifies the perturbation (accounting for a median filtering or a contrast enhancement) to become less detectable.

As for adversarial images detection, this method is not new as well. The authors of [30] wisely see steganalysis detection as a perfect companion to adversarial re-training. This last mechanism fights well against small perturbations. It however struggles in correctly classifying coarser and more detectable attacks. Unfortunately, this idea is supported with a proof of concept (as acknowledged by the authors): the steganalyzer is rudimentary, the dataset is composed of tiny images (MNIST). On the contrary, the authors of [22] outline that steganalysis works better on larger images like ImageNet (ILSVRC-2016). They however use a deprecated classifier (VGG-16 [33]) with outdated steganalyzers based on hand-crafted features (SPAM and SRM).

2.2 Adversarial examples

This paper focuses on white-box attacks where the attacker knows all implementation details of the classifier. To make things clearer, the classifier has the following structure: a pre-processing T maps an image $\mathbf{I}_o \in \{0, 1, \dots, 255\}^n$ (with $n = 3LC$, 3 color channels, L lines and C columns of pixels) to $\mathbf{x}_o = \mathsf{T}(\mathbf{I}_o) \in \mathcal{X}^n$, with $\mathcal{X} := [0, 1]$ (some networks also use $\mathcal{X} = [-1, 1]$ or $[-3, 3]$). This pre-processing is heuristic, sometimes it just divides the pixel value by 255, sometimes this normalization is channel dependent based on some statistics (empirical mean and standard deviation). After normalization, $\mathbf{x}_o$ feeds the trained neural network to produce the estimated probabilities $(\hat{p}_k(\mathbf{x}_o))_k$ of being from class

$k \in \{1, \ldots, K\}$. The predicted class is given by:

$$\hat{c}(\mathbf{x}_o) = \arg\max_k \hat{p}_k(\mathbf{x}_o). \tag{1}$$

The classification is correct if $\hat{c}(\mathbf{x}_o) = c(\mathbf{x}_o)$, the ground truth label of image I_o.

An *untargeted* adversarial attack aims at finding the optimal point:

$$\mathbf{x}_a^\star = \arg \min_{\mathbf{x}:\hat{c}(\mathbf{x}) \neq c(x_o)} \|\mathbf{x} - \mathbf{x}_o\|, \tag{2}$$

where $\|\cdot\|$ is usually the Euclidean distance.

Discovering this optimal point is difficult because the space dimension n is large. In a white-box scenario, all attacks are sub-optimal iterative processes. They use the gradient of the network function efficiently computed thanks to the back-propagation mechanism to find a solution $\mathbf{x}_a$ close to $\mathbf{x}_a^\star$. They are compared in terms of probability of success, average distortion, and complexity (number of gradient computations). This paper considers well-known attacks ranked from low to high complexity: FGSM [16], PGD [25], DDN [29], CW [7].

As outlined in [4], definition (2) is common in literature, yet it is incorrect. The final goal of the attacker is to create an adversarial image $\mathbf{I}_a$ in the pixel domain, not $\mathbf{x}_a$ in $\mathcal{X}^n$. Applying the inverse mapping T^{-1} is not solving the issue because this a priori makes non integer pixel values. Rounding to the nearest integer, $\mathbf{I}_a = [\mathsf{T}^{-1}(\mathbf{x}_a)]$, is simple but not effective. Some networks are so vulnerable (like ResNet-18) that $\mathsf{T}^{-1}(\mathbf{x}_a) - \mathbf{I}_o$ is a weak signal partially destroyed by rounding. The impact is that, after rounding, $\mathbf{I}_a$ is no longer adversarial. DDN is a rare example of a powerful attack natively offering quantized values.

Paper [4] proposes a post-processing Q on top of any attack that makes sure $\mathbf{I}_q = \mathsf{Q}(\mathsf{T}^{-1}(\mathbf{x}_a))$ is *(i)* an image (integral constraint), *(ii)* remains adversarial, and *(iii)* has a low Euclidean distortion $\|I_q - I_o\|$. This paper follows the same approach but adds another constraint: *(iv)* be non-detectable.

Figure 1 shows the characteristic function measuring the probability of success of an attack [4] as a function of the distortion budget (L_2-norm) against landmark classifiers in the history of ImageNet challenge. The characteristic function starts at $1 - \eta$, where η is the accuracy of the classifier: a proportion $1 - \eta$ of original images are naturally adversarial since there are misclassified. As we know, the accuracy of the networks increases as time goes by: AlexNet (2012) [19] < VGG-16 (2015) [33] < GoogLeNet (2015) [34] < ResNet-50 [18] (2016) < EfficientNet-b0 [35] (2019). On the other hand, the robustness to this attack can be measured by the average distortion necessary for hacking the images (cf. Table 1). This reveals a different hierarchy: ResNet-50 and VGG-16 are quite fragile contrary to the old AlexNet. Overall, the recent EfficientNet is both more accurate and more robust.

2.3 Defenses

The literature proposes four types of defenses or counter-attacks against adversarial examples white-box attacks:

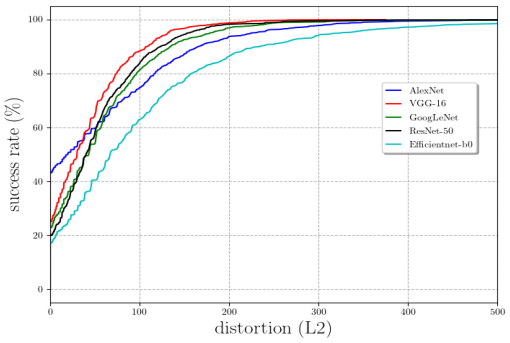

Fig. 1. Characteristic function of attack [4] (PGD in best effort with quantization) against well known (vanilla) classifiers for ImageNet.

Table 1. Robustness of recent classifiers against PGD_2 followed by quantization [4]

	Acc (%)	P_{suc} (%)	$\overline{L_2}$
Alexnet	57.0	100	104
VGG-16	75.0	100	56.5
GoogLeNet	77.2	99.8	72.9
ResNet-50	80.0	97.2	81
Vanilla EfficientNet-b0 [35]	82.8	99.1	115
Robust EfficientNet [43]	84.3	98.5	192

To detect: Being barely visible does not mean that the perturbation is not statistically detectable. This defense analyses the image and bypasses the classifier if detected as adversarial [24]. This is a forensics analysis of adversarial signals.
To reform: The perturbation looks like a random noise that may be filtered out. This defense is usually a front-end projecting the image back to the manifold of natural images [26].
To robustify: At learning, adversarial images are included in the training set with their original class labels. Adversarial re-training robustifies a 'vanilla' trained network [25].
To randomize: At testing, the classifier depends on a secret key or an alea. This blocks pure white-box attacks [37, 38].

This paper evaluates steganalysis as a candidate for the first line of defense against white-box attacks targeting vanilla or robust networks.

2.4 Steganographic costs

Undetectability is usually tackled by the concept of costs in the steganographic literature: each pixel location i of a given cover image is assigned a set of costs $(w_i(\ell))_\ell$ that reflects the detectability of modifying the i-th pixel by ℓ quantum. Usually, $w_i(0) = 0$, $w_i(-\ell) = w_i(\ell)$, and $w_i(|\ell|)$ is increasing. The goal of the steganographer is to embed a message $\mathbf{m}$ while minimizing the empirical steganographic distortion:

$$D(\boldsymbol{\ell}) := \sum_{i=1}^{n} w_i(\ell_i). \tag{3}$$

This is practically achieved using Syndrome Trellis Codes [13]. This paper proposes to use the steganographic distortion (instead of L_1, L_2 or L_∞ norms in adversarial literature) in order to decrease detectability.

Note that this distortion is additive, which is equivalent to consider that each pixel modification yields a detectability independent from the others. Yet, one strategy takes into account potential interactions between neighboring modifications: The image is first decomposed into disjoint lattices to be sequentially embedded where costs are then updated after the embedding over one lattice [21].

This work uses three families of steganographic costs. The first one, HILL [20], is empirical and naive, but has nevertheless been widely used in steganography thanks to its simplicity. The cost map $\mathbf{w}$ associated to ± 1 is computed using two low-pass averaging filters $\mathbf{L}_1$ and $\mathbf{L}_2$ of respective size 3×3 and 15×15 and one high pass filter $\mathbf{H}$: ($*$ means convolution)

$$\mathbf{w} = \frac{1}{|\mathbf{I} * \mathbf{H}| * \mathbf{L}_1} * \mathbf{L}_2, \text{with } \mathbf{H} = \begin{bmatrix} -1 & 2 & -1 \\ 2 & -4 & 2 \\ -1 & 2 & -1 \end{bmatrix}. \tag{4}$$

The second one, derived from MiPod [31], assumes that the residual signal is distributed as $\mathcal{N}(0, \sigma_i^2)$ for the original image, and $\mathcal{N}(\ell_i, \sigma_i^2)$ for the stego image. The variance σ_i^2 is estimated on each pixel using Wiener filtering and a least square approximation on a basis of cosine functions. The cost is the log likelihood ratio between the two distributions evaluated at 0, *i.e.*:

$$w_i(\ell_i) = \ell_i^2 / \sigma_i^2. \tag{5}$$

Unlike HILL, this model handles modifications other than ± 1.

The last one is a cost updating strategy favoring coherent modifications between pixels within a spatial or color neighborhood. It is called GINA [42] and it is derived from CMD [21]. It splits the color images into 4 disjoint lattices per channel, i.e. 12 lattices. The embedding performs sequentially starting by the green channel lattices. The costs on one lattice is updated according to the modifications done on the previous ones as:

$$w_i'(\ell_i) = \frac{1}{9} w_i(\ell_i), \text{ if } \mathsf{sign}(\ell_i) = \mathsf{sign}(\mu_i), \tag{6}$$

with μ_i the average of the modifications already performed in the spatial or colour neighborhood of location i.

2.5 Looking at Adversarial Examples with Stega Glasses

First, note that adversarial images recently became a source of inspiration for steganography: paper [36] proposes the concept of steganography with an adversarial embedding fooling a DNN-based steganalyzer. References [3] and [32] propose both to cast the problem of adversarial embedding as a game-theoretical problem. A protocol to train efficiently new adversaries and to generate less detectable stego contents using a min max strategy is presented in [3]. The reference [32] solves the game between one embedder and one steganalyst using both different levels of adversarial perturbations.

Paper [30] stresses however one fundamental difference between steganography and adverarial examples: Steganalysis has two classes, where the class 'cover' distribution is given by Nature, whereas the class 'stego' distribution is a consequence of designed embedding schemes. On the other hand, a *perfect* adversarial example and an original image are distributed as by the class $\hat{c}(\mathbf{x}_a)$ or $c(\mathbf{x}_o)$, which are both given by Nature.

We stress another major difference: Steganographic embedding is essentially a stochastic process. Two stego-contents derived from the same cover are different almost surely with STC [13]. This is a mean to encompass the randomness of the messages to be embedded. This is also the reason why steganographic embedders turns the costs $(w_i(\ell))_\ell$ into probabilities $(\pi_i(\ell))_\ell$ of modifying the i-th pixel by ℓ quantum. These probabilities are derived to minimize the detectability under the constraint of an embedding rate given by the source coding theorem:

$$R = -n^{-1} \sum_i \sum_{\ell_i} \pi_i(\ell_i) \log_2 \left(\pi_i(\ell_i)\right) \text{ bits.} \tag{7}$$

In contrast, an attack is a deterministic process always giving the same adversarial version of one original image. Adversarial imaging does not need these probabilities.

3 Steganographic Post-Processing

This section presents the use of steganography in our post-processing Q mounted on top of any adversarial attack.

3.1 Optimal post-processing

Starting from an original image, we assume that an attack has produced $\mathbf{x}_a$ mapped back to $\mathbf{I}_a = \mathsf{T}^{-1}(\mathbf{x}_a)$. The problem is that $\mathbf{I}_a \in [0, 255]^n$, *i.e.* its pixel values are a priori not quantized. Our post-processing specifically deals with that matter, outputting $\mathbf{I}_q = \mathsf{Q}(\mathbf{I}_a) \in \{0, \dots, 255\}^n$. We introduce $\mathbf{p}$ the perturbation after the attack and $\mathbf{q}$ the perturbation after our post-processing:

$$\mathbf{p} := \mathbf{I}_a - \mathbf{I}_o \in \mathbb{R}^n, \tag{8}$$

$$\boldsymbol{\ell} := \mathbf{I}_q - \mathbf{I}_o \in \mathbb{Z}^n. \tag{9}$$

The design of Q amounts to find a good $\boldsymbol{\ell}$. This is more complex than just rounding perturbation $\mathbf{p}$.

We first restrict the range of $\boldsymbol{\ell}$. We define the degree of freedom d as the number of possible values for each ℓ_i, $1 \leq i \leq n$. This is an even integer greater than or equal to 2. The range of ℓ_i is centered around p_i. For instance, when $d = 2$, $\ell_i \in \{\lfloor p_i \rfloor, \lceil p_i \rceil\}$. In general, the range is given by

$$\mathcal{L}_i := \{\lceil p_i \rceil - d/2, \ldots, \lceil p_i \rceil - 1, \lceil p_i \rceil, \ldots, \lceil p_i \rceil + d/2 - 1\}. \tag{10}$$

Over the whole image, there are d^n possible sequences for $\boldsymbol{\ell}$.

We now define two quantities depending on $\boldsymbol{\ell}$. The *classifier loss* at $\mathbf{I}_q = \mathbf{I}_a - \mathbf{p} + \boldsymbol{\ell}$:

$$L(\boldsymbol{\ell}) := \log(\hat{p}_{c_o}(\mathbf{I}_a - \mathbf{p} + \boldsymbol{\ell})) - \log(\hat{p}_{c_a}(\mathbf{I}_a - \mathbf{p} + \boldsymbol{\ell})), \tag{11}$$

where c_o is the ground truth class of $\mathbf{I}_o$ and c_a is the predicted class after the attack. When the attack succeeds, it means that $\mathbf{I}_a$ is classified as $c_a \neq c_o$ because $\hat{p}_{c_a}(\mathbf{I}_a) > \hat{p}_{c_o}(\mathbf{I}_a)$ so that $L(\mathbf{p}) < 0$. Our post-processing cares about maintaining this adversariality. This constrains $\boldsymbol{\ell}$ s.t. $L(\boldsymbol{\ell}) < 0$.

The second quantity is the *detectability*. We assume that a black-box algorithm gives the stego-costs $(w_i(\ell))_\ell$ for a given original image. The overall detectability of $\mathbf{I}_q$ is gauged by $D(\boldsymbol{\ell})$ as given by (3). In the end, the optimal post-processing Q minimizes detectability while maintaining adversariality:

$$\boldsymbol{\ell}^\star = \arg \min_{\boldsymbol{\ell}: L(\boldsymbol{\ell}) < 0} D(\boldsymbol{\ell}). \tag{12}$$

3.2 Our proposal

The complexity for finding the solution of (12) a priori scales as $O(d^n)$. Two ideas from the adversarial examples literature help reducing this cost. First, the problem is stated as an Lagrangian formulation as in [7]:

$$\boldsymbol{\ell}_\lambda = \arg\min D(\boldsymbol{\ell}) + \lambda L(\boldsymbol{\ell}). \tag{13}$$

where $\lambda \geq 0$ is the Lagrangian multiplier. This means that we must solve this problem for any λ and then find the smallest value of λ s.t. $L(\boldsymbol{\ell}_\lambda) < 0$.

Second, the classifier loss is linearized around $\mathbf{I}_a$, *i.e.* for $\boldsymbol{\ell}$ around $\mathbf{p}$: $L(\boldsymbol{\ell}) \approx L(\mathbf{p}) + (\boldsymbol{\ell} - \mathbf{p})^\top \mathbf{g}$, where $\mathbf{g} = \nabla L(\mathbf{p})$. This transforms problem (13) into

$$\boldsymbol{\ell}_\lambda = \arg\min \sum_{i=1}^{n} w_i(\ell_i) + \lambda(p_i - \ell_i).g_i. \tag{14}$$

The solution is now tractable because the functional is separable: we can solve the problem pixel-wise. The algorithm stores in $d \times n$ matrix W the costs, and in $d \times n$ matrix G the values $((p_i - \ell_i).g_i)_i$ for $\ell_i \in \mathcal{L}_i$ (10). For a given λ, it computes $W + \lambda G$ and looks for the minimum of each column $1 \leq i \leq n$. In

other words, it is as complex as n minimum findings, each over d values, which scales as $O(n \log d)$.

Note that for $\lambda = 0$, Q quantizes $I_{a,i}$ 'towards' $I_{o,i}$ to minimize detectability. Indeed, if $\ell_i = 0$ is admissible ($0 \in \mathcal{L}_i$ holds if $|p_i| \leq d/2$), then $\mathsf{Q}(I_{a,i}) = I_{o,i}$ at $\lambda = 0$.

On top of solving (14), a line search over λ is required. The linearization of the loss being a crude approximation, we make calls to the network to check that $\mathsf{Q}(\mathbf{I}_a)$ is adversarial: When testing a given value of λ, $\boldsymbol{\ell}_\lambda$ is computed to produce I_q that feeds the classifier. If I_q is adversarial then $L(\boldsymbol{\ell}_\lambda) < 0$ and we test a lower value of λ (giving more importance to the detectability), otherwise we increase it. The search is performed over $\log_2(\mathrm{n})$ steps. The images we used are of dimension 224×224×3 which gives 18 steps. Optimal λ varies widely in value between different images.

3.3 Simplification for quadratic stego-costs

We now assume that the stego-costs obey to the following expression: $w_i(\ell) = \ell^2/\sigma_i^2$ as in (5). This makes the functional of (14) (restricted to the i-th pixel) equals to $\ell_i^2/\sigma_i^2 - \lambda g_i \ell_i + \lambda p_i$ which minimizer is $\tilde{\ell}_i = \lambda g_i \sigma_i^2/2$.

Yet, this value in general is not an integer belonging to $\mathcal{L}_i$ (10). This issue is easily solved because a quadratic function is symmetric around its minimum, therefore the minimum over $\mathcal{L}_i$ is its value closest to $\tilde{\ell}_i$ as shown in Fig. 2. The range $\mathcal{L}_i$ being nothing more than a set of consecutive integers, we obtain a closed form expression:

$$\ell_{\lambda,i} = \min(\max([\lambda g_i \sigma_i^2/2], \lceil p_i \rceil - d/2), \lceil p_i \rceil + d/2 - 1), \tag{15}$$

where $[\cdot]$ is the rounding to the nearest integer. The post-processing has now a linear complexity.

In this equation, the min and max operate a clipping so that $\ell_{\lambda,i}$ belongs to $\mathcal{L}_i$. This clipping is active if $\tilde{\ell}_i \notin \mathcal{L}_i$, which happens if $\lambda \geq \bar{\lambda}_i$ with

$$\bar{\lambda}_i := \begin{cases} \left| \frac{2\lceil p_i \rceil - d}{g_i \sigma_i^2} \right|_+ & \text{if } g_i < 0 \\ \left| \frac{2\lceil p_i \rceil + d - 2}{g_i \sigma_i^2} \right|_+ & \text{if } g_i > 0, \end{cases} \tag{16}$$

where $|a|_+ = a$ if $a > 0$, 0 otherwise. This remark is important because it shows that for any $\lambda > \max_i \bar{\lambda}_i$, the solution $\boldsymbol{\ell}_\lambda$ of (15) remains the same due to clipping. Therefore, we can narrow down the line search of λ to $[0, \max_i \bar{\lambda}_i]$.

4 Experimental Investigation

4.1 Experimental setup

Our experimental work uses 18,000 images from ImageNet of dimension 224×224×3. This subset is split in 1,000 for testing and comparing, 17,000 for training. An

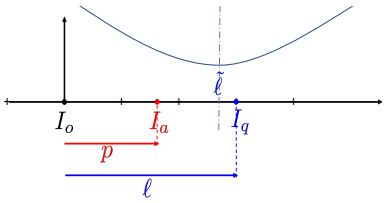

Fig. 2. Rounding the minimizer when the stego-cost is quadratic.

image is attacked only if the classifier predicts its correct label beforehand. This happens with probability equaling the accuracy of the network Acc. We measure $\overline{L_2}$ the average Euclidean distance of the perturbation $\boldsymbol{\ell}$ and P_{suc} the probability of a successful attack *only over correctly labeled images*.

We attack the networks with 4 different attacks: FGSM [16], PGD_2 [25], CW [7] and DDN [29]. All these attacks are run in a *best-effort* fashion with a complexity limited to 100 iterations. This means that for FGSM and PGD_2 the distortion is gradually increased until the image is adversarial. For more complex CW and DDN attacks, different parameters are used over a total maximum of 100 iterations. The final attacked version is the adversarial image with the smaller distortion. Since DDN is the only attack that creates integer images, the other 3 are post-processed either by the enhanced quantization [4], which is our baseline, or by our method explained in Sect. 3.2.

The adversarial image detectors are evaluated by the true positive rate TPR_5 when the false positive rate FPR is fixed to 5%.

4.2 Robustness of recent classifiers: there is free lunch

Our first experiment compares the robustness of the famous ResNet-50 network to the recent classifiers: the vanilla version of EfficientNet-b0 [35] and its robust version trained with AdvProp [43]. Note that the authors of [43] apply adversarial re-training for improving accuracy. As far as we known, the robustness of this version was not yet established.

Figure 3 shows the same characteristic function as in Figure 1 with this time the vanilla EfficientNet-b0 against its robust version. Table 1 gives measurements P_{suc} and $\overline{L_2}$ as a summary of the characteristic function shown in Fig 1. This confirms that modern classifiers are more accurate and more robust (lower P_{suc} and/or bigger $\overline{L_2}$). This is indeed a surprise: It pulls down the myth of 'No Free Lunch' in adversarial machine learning literature [40, 10] (the price to pay for robustifying a network is allegedly a lower accuracy).

4.3 Detection with forensics detectors

We use three steganalyzers to detect adversarial images. Their training set is composed of 15,651 pairs of original and adversarial images. The latter are crafted with *best-effort* FGSM against vanilla EfficientNet-b0.

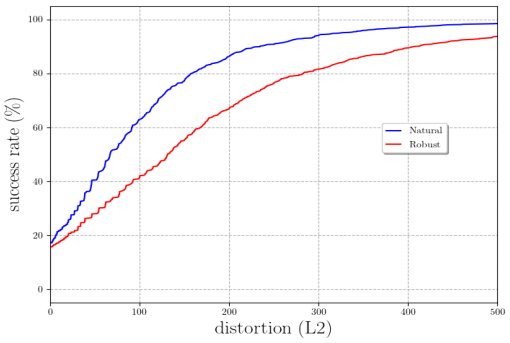

Fig. 3. Characteristic function of attack [4] (PGD in best effort with quantization) against Vanilla EfficientNet-b0 and its Robust counterpart.

The first detector is trained on SRM feature vectors [14], with dimensions 34,671. SRM is a model that applies to only one channel. It is computed on the luminance of the image in our experimental work. The classifier separating these high-dimensional vectors into two classes is the linear regularized classifier [9]. The second detector is based on the color version of SRM: SCRMQ1 [15] with dimension 18,157. The classifier is the same. The third detector is SRNet [5], one of the best detectors in steganalysis. Training is performed on 180 epochs: The first 100 with a learning rate of 10^{-3}, the remaining 80 with 10^{-4}. Data augmentation is also performed during training. First, there is a probability $p_1 = 0.5$ of mirroring the pair of images. Then, there is another probability $p_2 = 0.5$ of rotating them by 90 degrees.

The attacks: Table 2 shows the results of detection on all 4 attacks. PGD_2 achieves a high P_{suc} at almost a third of the distortion FGSM would obtain. DDN and CW being harder to optimize attain both lower P_{suc} and higher distortion under the given constraints. For the rest of the study we therefore focus on PGD_2 to give the best attacking setup with reasonable complexity.

The detectors: Table 2 gives also the TPR_5 associated to the detectors. Although [22] achieves good performances with SRM, we do not obtain the high

Table 2. Detection probabilities (TPR_5) with forensics detectors of adversarial images targeting classifier vanilla EfficientNet-b0 [35]

	P_{suc}	$\overline{L_2}$	SRM(%)	SCRMQ1(%)	SRNet(%)
FGSM+[4]	89.7	286	72.00	83.3	**93.5**
PGD_2+[4]	98.6	113	65.02	83.1	**93.8**
CW+[4]	89.7	97	68.78	83.6	**94.5**
DDN	83.2	186	79.53	91.9	**94.8**

Table 3. Undetectability of steganographic embedding on PGD_2 against the vanilla model (Van) and its robust version (Rob).

	d	P_{suc} (%)		$\overline{L_2}$		SCRMQ1(%)		SRNet(%)	
		Van	Rob	Van	Rob	Van	Rob	Van	Rob
[4]	2	98.6	98.3	**101**	**167**	83.1	84.6	93.8	90.1
HILL	2	98.6	98.3	113	177	78.0	76.6	87.6	88.5
HILL	4	**98.9**	**98.5**	125	181	76.0	73.3	87.4	88.2
MiPod	2	98.3	98.3	176	242	77.4	76.2	86.6	87.7
MiPod	4	98.7	98.0	164	247	74.4	70.2	84.5	87.7
GINA	2	98.5	98.1	283	337	24.4	32.4	68.3	**82.9**
GINA	4	98.8	98.2	300	330	**18.6**	**24.3**	**50.9**	85.2

detection rates reported in the reference. This cab be due to both finer attacks (best effort mode) and quantization. Our results show also that the detectors generalize well: although trained to detect images highly distorted by FGSM, they can detect as well and sometimes even better more subtle attacks like CW. Moreover, SRNet always outperforms SCRMQ1 and is the most accurate of the three detectors. From table 2, we can also deduce that PGD_2+[4] is the worst-case scenario for defense. The probability of fooling both the classifier EfficientNet-b0 and the detector SRNet in this setup combines to only $0.88 \times (1 - 0.933) = 5.9\%$.

4.4 Post-processing with a Steganographic Embedder

We now play the role of the attacker. We use PGD_2 with best effort as the base attack to compare the detectability of four post-processings: The non-steganographic insertion [4] as a baseline, HILL (4), MiPod (5), and GINA (6). GINA uses the quadratic method explained in Sect. 3.3 sequentially over the 12 lattices. Quadratic stego-costs are updated with CMD strategy (6). Each lattice contributes to a 1/12 of the initial classification loss.

Table 3 illustrates how each strategy is detected by either SCRMQ1 or SRNet. Both detectors are trained on FGSM with [4] quantization as 'stego' images crafted on their respective network. Distortion increases with each method and along the degree of freedom d. The use of Steganographic costs therefore enables to reduce the detectability while increasing the L_2 distortion.

From the attacker perspective, the best strategy to fool the detector PGD_2 is GINA costs with $d = 4$. This scenario now has 48.0% chance of fooling both Vanilla EfficientNet-b0 and SRNet and 80.4% with SCRMQ1 as the detector. Fig. 4 shows the two examples with highest distortion on EfficientNet-b0 that still fool SRNet. The added distortion remains imperceptible to the human eye even in these cases.

The conclusion on Robust EfficientNet-b0 is however different. Since the distortion needed to attack the network is higher, it is consequently expected that the detectors will be more accurate. If SCRMQ1 detects GINA distortion

Fig. 4. Top row: Cover images with their label below. Bottom row: adversarial images with steganographic embedding GINA (d=4). Below them are their new label and the distortion

slightly better than on Vanilla EfficientNet-b0, SRNet is however very efficient to detect each strategy even if it was trained on FGSM.

4.5 Training on adversarial images with GINA costs

We finally play the role of the defender again. We want to detect GINA perturbation with the highest possible TPR. To achieve this we retrain our detectors in the same setups as before, but with images using GINA perturbation as adversarial images. Since table 3 shows that in most cases $d = 4$ is indeed the worst-case for the defense side, we attacked the training set of "cover" images with PGD_2 and GINA costs with $d = 4$.

The first result we report is that under the same setup, SRNet was never able to distinct both distributions of images. The average confidence on the whole test set is roughly 50%. Trying to train SRNet with a finer learning rate did not lead to any better result. There is probably a set of *hyperparameters* that would lead to a more effective training. However this result illustrates that GINA distortion is harder to detect.

Table 4 shows TPR_5 for SCRMQ1 under such training setup. The detector is able to detect GINA mechanism at a higher rate than in Table 3 but generalizes poorly on other attacks. A conclusion to this final experiment is that GINA can be stealthy to general detectors, but it is still better detected after another iteration of the defender. The detection accuracy is however lower when using GINA costs, and drops from 83.1% to 68.5%. The price of detecting GINA is also to become more specific and to lose performance on the other attacks.

Table 4. Detection on SCRMQ1 after training on adversarial images embedded with GINA (d=4)

	d	SCRMQ1(%)	
		Van	Rob
[4]	2	55.9	56.7
HILL	2	53.4	53.6
HILL	4	50.4	53.9
MiPod	2	56.1	55.9
MiPod	4	53.9	54.9
GINA	2	**77.7**	78.4
GINA	4	68.5	**79.7**

5 Conclusions

This paper explores both sides of adversarial image detection with steganographic glasses.

On the Attack side, our work using distortions designed for steganographic purposes is able to reduce the detection rates. Steganographic distortion target specific regions and pixels of an image to quantize the attack. The L_2 distortion increases w.r.t. the original attack, but remains imperceptible by the human eye (Fig. 4) and less detectable by a targeted detector. This paper consequently shows the possibility of tweaking an attack to make it harder to detect while remaining invisible.

On the Defense side, we use SRNet [5], state-of-the-art in steganalysis to detect adversarial images. Training it on images attacked with the basic FGSM shows excellent performance. Detection also generalizes well even on the finest attacks such as PGD_2 [25] and CW [7].

Finally both Attack and Defense are affected by the considered neural network. The effect of adversarial training on EfficientNet-b0 [43] is twofold: it increases the classification accuracy as well as robustifying the network. An increased robustness translates into a higher attacking distortion, which itself translates into a higher detectability.

References

1. Athalye, A., Carlini, N., Wagner, D.A.: Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In: ICML. pp. 274–283 (2018)
2. Barni, M., Stamm, M.C., Tondi, B.: Adversarial multimedia forensics: Overview and challenges ahead. In: 2018 26th European Signal Processing Conference (EUSIPCO). pp. 962–966. IEEE (2018)
3. Bernard, S., Bas, P., Klein, J., Pevny, T.: Explicit optimization of min max steganographic game. IEEE Transactions on Information Forensics and Security **16**, 812–823 (2021)

4. Bonnet, B., Furon, T., Bas, P.: What if adversarial samples were digital images? In: Proc. of ACM IH&MMSec '20. pp. 55–66 (2020). https://doi.org/10.1145/3369412.3395062
5. Boroumand, M., Chen, M., Fridrich, J.: Deep residual network for steganalysis of digital images. IEEE Transactions on Information Forensics and Security **14**(5), 1181–1193 (2018)
6. Carlini, N., Wagner, D.: Adversarial examples are not easily detected: Bypassing ten detection methods. arXiv:1705.07263 (2017)
7. Carlini, N., Wagner, D.: Towards evaluating the robustness of neural networks. In: IEEE Symp. on Security and Privacy (2017)
8. Chen, Z., Tondi, B., Li, X., Ni, R., Zhao, Y., Barni, M.: A gradient-based pixel-domain attack against svm detection of global image manipulations. In: IEEE WIFS. pp. 1–6 (2017)
9. Cogranne, R., Sedighi, V., Fridrich, J., Pevnỳ, T.: Is ensemble classifier needed for steganalysis in high-dimensional feature spaces? In: Information Forensics and Security (WIFS), 2015 IEEE International Workshop on. pp. 1–6. IEEE (2015)
10. Dohmatob, E.: Generalized no free lunch theorem for adversarial robustness. In: Proc. of Int. Conf. on Machine Learning. Long Beach, California, USA (2019)
11. Fan, W., Wang, K., Cayre, F.: General-purpose image forensics using patch likelihood under image statistical models. In: IEEE Int.Workshop on Information Forensics and Security (WIFS). pp. 1–6 (2015)
12. Farooq, S., Yousaf, M.H., Hussain, F.: A generic passive image forgery detection scheme using local binary pattern with rich models. Computers & Electrical Engineering **62**, 459 – 472 (2017). https://doi.org/https://doi.org/10.1016/j.compeleceng.2017.05.008
13. Filler, T., Judas, J., Fridrich, J.: Minimizing additive distortion in steganography using syndrome-trellis codes. IEEE Transactions on Information Forensics and Security **6**(3), 920–935 (2011)
14. Fridrich, J., Kodovsky, J.: Rich models for steganalysis of digital images. Information Forensics and Security, IEEE Transactions on **7**(3), 868–882 (2012)
15. Goljan, M., Fridrich, J., Cogranne, R.: Rich model for steganalysis of color images. 2014 IEEE International Workshop on Information Forensics and Security, WIFS 2014 pp. 185–190 (04 2015). https://doi.org/10.1109/WIFS.2014.7084325
16. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. In: ICLR 2015, San Diego, CA, USA, (2015)
17. Güera, D., Wang, Y., Bondi, L., Bestagini, P., Tubaro, S., Delp, E.J.: A counter-forensic method for cnn-based camera model identification. In: 2017 IEEE conference on computer vision and pattern recognition workshops (CVPRW). pp. 1840–1847. IEEE (2017)
18. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 770–778 (2016)
19. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1. pp. 1097–1105. NIPS'12, Curran Associates Inc., Red Hook, NY, USA (2012)
20. Li, B., Wang, M., Huang, J., Li, X.: A new cost function for spatial image steganography. In: Image Processing (ICIP), 2014 IEEE International Conference on. pp. 4206–4210. IEEE (2014)

21. Li, B., Wang, M., Li, X., Tan, S., Huang, J.: A strategy of clustering modification directions in spatial image steganography. Information Forensics and Security, IEEE Trans. on **10**(9) (2015)
22. Liu, J., Zhang, W., Zhang, Y., Hou, D., Liu, Y., Zha, H., Yu, N.: Detection based defense against adversarial examples from the steganalysis point of view. In: IEEE/CVF CVPR. pp. 4820–4829 (2019)
23. Luo, W., Li, H., Yan, Q., Yang, R., Huang, J.: Improved audio steganalytic feature and its applications in audio forensics. ACM Trans. Multimedia Comput. Commun. Appl. **14**(2) (Apr 2018). https://doi.org/10.1145/3190575
24. Ma, S., Liu, Y., Tao, G., Lee, W., Zhang, X.: NIC: detecting adversarial samples with neural network invariant checking. In: NDSS 2019, San Diego, California, USA. (2019)
25. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. In: ICLR 2018, Vancouver, BC, Canada. (2018)
26. Meng, D., Chen, H.: Magnet: a two-pronged defense against adversarial examples. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. pp. 135–147. ACM (2017)
27. Qiu, X., Li, H., Luo, W., Huang, J.: A universal image forensic strategy based on steganalytic model. In: Proc. of ACM IH&MMSec '14. pp. 165–170. New York, NY, USA (2014). https://doi.org/10.1145/2600918.2600941
28. Quiring, E., Arp, D., Rieck, K.: Forgotten siblings: Unifying attacks on machine learning and digital watermarking. In: IEEE European Symp. on Security and Privacy (2018). https://doi.org/10.1109/EuroSP.2018.00041
29. Rony, J., Hafemann, L.G., Oliveira, L.S., Ayed, I.B., Sabourin, R., Granger, E.: Decoupling direction and norm for efficient gradient-based l2 adversarial attacks and defenses. In: Proc. of the IEEE CVPR (2019)
30. Schöttle, P., Schlögl, A., Pasquini, C., Böhme, R.: Detecting adversarial examples - a lesson from multimedia security. In: European Signal Processing Conference (EUSIPCO). pp. 947–951 (2018)
31. Sedighi, V., Cogranne, R., Fridrich, J.: Content-adaptive steganography by minimizing statistical detectability. Information Forensics and Security, IEEE Transactions on **11**(2), 221–234 (2016)
32. Shi, X., Tondi, B., Li, B., Barni, M.: Cnn-based steganalysis and parametric adversarial embedding: a game-theoretic framework. Signal Processing: Image Communication p. 115992 (2020)
33. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings (2015), http://arxiv.org/abs/1409.1556
34. Szegedy, C., Wei Liu, Yangqing Jia, Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1–9 (2015)
35. Tan, M., Le, Q.V.: Efficientnet: Rethinking model scaling for convolutional neural networks. arXiv (2019)
36. Tang, W., Li, B., Tan, S., Barni, M., Huang, J.: CNN-based adversarial embedding for image steganography. IEEE Transactions on Information Forensics and Security **14**(8), 2074–2087 (2019)
37. Taran, O., Rezaeifar, S., Holotyak, T., Voloshynovskiy, S.: Defending against adversarial attacks by randomized diversification. In: IEEE CVPR. Long Beach, USA (June 2019)

38. Taran, O., Rezaeifar, S., Holotyak, T., Voloshynovskiy, S.: Machine learning through cryptographic glasses: combating adversarial attacks by key based diversified aggregation. In: EURASIP Journal on Information Security (January 2020)
39. Tramer, F., Carlini, N., Brendel, W., Madry, A.: On adaptive attacks to adversarial example defenses (2020)
40. Tsipras, D., Santurkar, S., Engstrom, L., Turner, A., Madry, A.: Robustness may be at odds with accuracy (2018)
41. Tuama, A., Comby, F., Chaumont, M.: Camera model identification based machine learning approach with high order statistics features. In: EUSIPCO. pp. 1183–1187 (2016)
42. Wang, Y., Zhang, W., Li, W., Yu, X., Yu, N.: Non-additive cost functions for color image steganography based on inter-channel correlations and differences. IEEE Trans. on Information Forensics and Security (2019)
43. Xie, C., Tan, M., Gong, B., Wang, J., Yuille, A., Le, Q.V.: Adversarial examples improve image recognition. arXiv (2019)
44. Xu, G., Wu, H.Z., Shi, Y.Q.: Structural design of convolutional neural networks for steganalysis. IEEE Signal Processing Letters **23**(5), 708–712 (2016)
45. Yousfi, Y., Butora, J., Fridrich, J., Giboulot, Q.: Breaking ALASKA: Color separation for steganalysis in jpeg domain. In: Proc. of ACM IH&MMSec '19. pp. 138–149 (2019)