Pasquini and Böhme *EURASIP Journal on Information Security* (2020) 2020:12
https://doi.org/10.1186/s13635-020-00104-z

EURASIP Journal on Information Security

RESEARCH Open Access

Trembling triggers: exploring the sensitivity of backdoors in DNN-based face recognition

Cecilia Pasquini[1*] and Rainer Böhme[2]

Abstract

Backdoor attacks against supervised machine learning methods seek to modify the training samples in such a way that, at inference time, the presence of a specific pattern (trigger) in the input data causes misclassifications to a target class chosen by the adversary. Successful backdoor attacks have been presented in particular for face recognition systems based on deep neural networks (DNNs). These attacks were evaluated for identical triggers at training and inference time. However, the vulnerability to backdoor attacks in practice crucially depends on the sensitivity of the backdoored classifier to *approximate* trigger inputs. To assess this, we study the response of a backdoored DNN for face recognition to trigger signals that have been transformed with typical image processing operators of varying strength. Results for different kinds of geometric and color transformations suggest that in particular geometric misplacements and partial occlusions of the trigger limit the effectiveness of the backdoor attacks considered. Moreover, our analysis reveals that the spatial interaction of the trigger with the subject's face affects the success of the attack. Experiments with physical triggers inserted in live acquisitions validate the observed response of the DNN when triggers are inserted digitally.

Keywords: Backdoor attacks, Neural networks, Adversarial machine learning

1 Introduction

The field of machine learning has experienced tremendous developments in the recent years. Inexpensive compute power in data-parallel architectures and the availability of large labeled datasets have spurred a race for increasingly advanced models, which are able to capture ever more complex structures of the underlying distribution [1]. Originating from computer vision, the use of deep neural networks (DNN) has led to unprecedented performance in many automatic learning tasks [2]. As a result, DNNs will likely become a key element in security decisions, such as in identification, authentication, and intrusion detection.

However, many machine learning methods are vulnerable to attacks that can compromise their performance in adversarial scenarios [3], where a malicious user can modify the data used for training or at inference time. Pioneering works in machine learning security [4, 5] have proposed a taxonomy of possible attacks, categorizing them by the domain of influence, the knowledge available to the adversary, and the protection goals violated. Subsequent explorations of this attack space have confirmed vulnerabilities to machine learning in general and, more recently, deep learning in particular [6, 7].

Many studies focus on explorative (or evasion) attacks (i.e., *adversarial examples*), where the adversary acts at inference time by creating strategically modified inputs with the goal of causing misclassification. In causative (or poisoning) attacks, the adversary instead manipulates the training samples strategically in order to affect the performance of the classifier at inference time [8]. While less studied than explorative attacks, causative attacks can be powerful and hard to detect [9].

*Correspondence: cecilia.pasquini@unitn.it
[1]Department of Information Engineering and Computer Science, University of Trento, Via Sommarive 9, 38123 Trento, Italy
Full list of author information is available at the end of the article

In this work, we focus on *backdoor attacks*, a class of causative attacks where the model is trained to output a certain target class when a specific pattern (called *trigger*) is present in the input sample. Backdoor attacks pose a significant security risk to several application domains of neural networks. A particularly relevant case is face recognition. Recent works have proposed training strategies that lead a classifier to assign a specific identity whenever the trigger is present, regardless of whose face is depicted. The literature reports an impressive effectiveness of this attack [10]. As demonstrated in [11], this paves the way to attacks in the physical world, where an attacker could fool camera-based face recognition systems by exhibiting a trigger-like object in front of the camera.

The backdoored models proposed in the literature are typically designed and tested scenarios where the malicious input data presented to the model at inference time carries exactly the same trigger used for training. This carries the implied assumption that the attacker has full control over the input data. In practice, however, inputs can be pre-processed images or probes acquired in real-time from a physical sensor. This distortion of the trigger information, for instance due to geometric displacement or varying illumination conditions, is often beyond the attacker's control. While these factors have been investigated in the context of adversarial examples [12], we are not aware of similar work for backdoor attacks.

To close this gap, the present work studies the sensitivity of a backdoored model to triggers that have been transformed with typical signal processing operators. We choose a recently proposed model for the domain of face recognition [10] and consider a post-training scenario, where the classifier is given a modified trigger with respect to the one it has been trained to recognize. We apply different kinds of transformations, including geometric transformations, occlusion, and different image compression pipelines. In each case, we analyze the classification outputs as a function of the controlled strength of the transformation. This allows us to empirically determine critical thresholds for the attack's effectiveness. Physical triggers have also been realized and their effectiveness has been tested in several settings. As a first step towards exploring the causes for the observed effects, we also study the interaction of the trigger with the face information contained in the image. This sensitivity analysis allows us to gain insights on the visual properties of the trigger that are most relevant for the infected model. Moreover, it provides an intuition on the feasibility and risk of backdoor attacks in the physical domain.

The rest of this paper is structured as follows: Section 2 introduces backdoor attacks to neural networks, reviews prior work on the topic, and positions our work within the literature. Section 3 describes the method adopted in our analysis and the experimental setting considered. Section 4 reports the experimental results on the studied face recognition model. In Section 5 documents the validation experiments and their results. Section 6 concludes with a discussion.

2 Background and prior work

We consider a multi-class classification problem where an input sample $\mathbf{x} \in \mathbb{R}^N$ is assigned to one of the K classes in $\{c_1, \ldots, c_K\}$. This is achieved by a neural network model $F : \mathbb{R}^N \to \mathbb{R}^K$, with parameters induced during a training phase. $F(\cdot)$ takes as input a sample $\mathbf{x}$ and provides a K-dimensional output vector whose kth element is interpreted as the probability of $\mathbf{x}$ belonging to class c_k. The decision on $\mathbf{x}$ is then taken by assigning the class with the highest probability, i.e., $d(\mathbf{x}) = \arg\max_{c \in \{c_1,\ldots,c_K\}} F(\mathbf{x})_c$. The maximum value $F(\mathbf{x})_{d(\mathbf{x})}$ is interpreted as the classification confidence.

Backdoor attacks belong to the class of causative attacks: the attacker influences the training phase with the goal of causing a specific behavior of the model at inference time. Different threat models include the case where the attacker has full access to the training set and can train the network from scratch or the case where the attacker can only retrain a pre-trained model (transfer-learning scenario).

In this context, the peculiarity of backdoor attacks with respect to conventional causative attacks is that the adversarial effect should take place if the input sample contains a specific pattern, called *trigger*, while the model should behave normally when no trigger is present. We can formalize this by defining an embedding function $\Psi(\mathbf{x}, \mathbf{s})$ that inserts a trigger $\mathbf{s}$ into an input sample $\mathbf{x}$, resulting in a new input sample in the same space as $\mathbf{x}$. Then, the attacker wants to achieve that the model $F(\cdot)$ misclassifies any sample $\Psi(\mathbf{x}, \mathbf{s})$. The most relevant case is a targeted backdoor with *target class* c_t. Here, the attacker's goal is to enforce that

$$d(\Psi(\mathbf{x}, \mathbf{s})) = c_t, \quad \forall \mathbf{x}.$$

The input $\mathbf{x}$, the trigger $\mathbf{s}$, and the embedding function Ψ are defined according to the experimental scenario and kind of data. We will provide these specifications for our analysis in Section 3. Moreover, in the following, we will use the expression $\mathbf{s}$-*infected* for both backdoored models trained to react to the trigger $\mathbf{s}$ and input samples that "contain" the trigger $\mathbf{s}$ (i.e., are output of the embedding function Ψ). By contrast, we will refer to *clean* models and inputs when training and testing are performed in non-adversarial conditions. Figure 1 illustrates backdoor attacks.

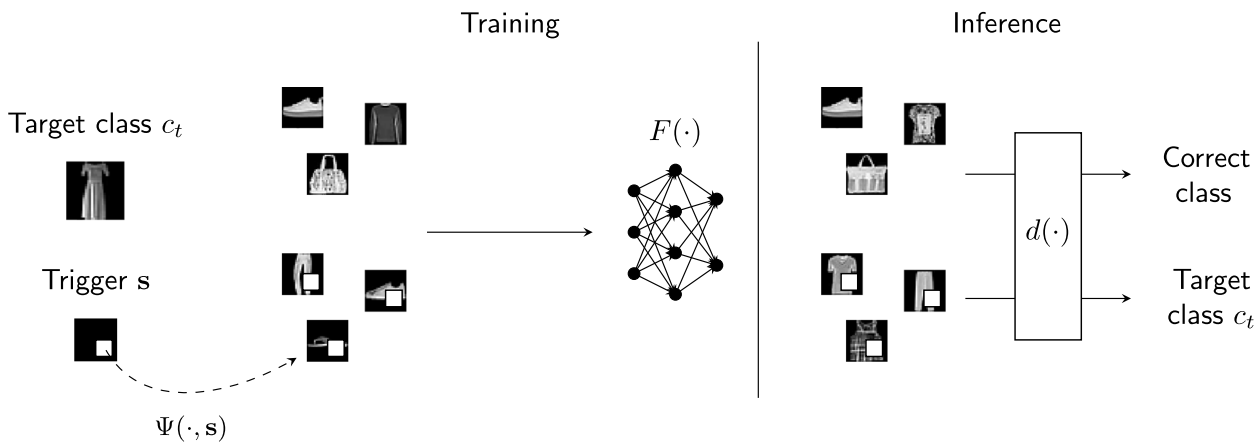


Fig. 1 Illustration of backdoor attacks. The training set typically contains **s**-infected samples labeled with the target class. At inference time, the model assigns samples containing the trigger to the target class, while classifying clean samples correctly

2.1 Known backdoor attacks and defenses

Backdoor attacks against neural networks have been first proposed in [13] and extended in [14]. The attacker initially chooses the target class and the trigger. In the case of images, the trigger can be an arbitrary set of pixel locations and values. Poisoning is carried out by acting on a random subset of the training set, where the selected trigger is embedded into the samples and the corresponding label is set to the target class. By assumption, the attacker has full control over the training procedure (including dataset, loss function, learning rate, fraction of modified data) and can adjust it to achieve her goals. Under these conditions, the approach yields attack success rates of more than 99% on the popular MNIST dataset [15], while preserving good performance on data not exhibiting the trigger.

More recently, the authors of [10] addressed a more challenging transfer learning scenario where the attacker inserts a backdoor in a pre-trained model by retraining it with poisoned data. In order to compensate for the limited control over the training data, rather than using arbitrary triggers, they derive a suitable trigger from the pre-trained model. By doing so, they construct triggers that achieve high success rates on models trained for different application domains (from face recognition to speech processing), while using few additional training samples. A transfer learning scenario is also addressed in [16], where the authors propose a procedure to inject backdoors that can be transferred from a "teacher" model to a "student" model. Moreover, recent approaches focused on improving the stealthiness of the trigger [17], which reduces the visual detectability of backdoor attacks on images, as well as the possibility to inject attacks without imposing poisoned labels [18]. Backdoor attacks have also been investigated for video signals [19].

Several defenses against backdoor attacks have been proposed [20, 21]. The approach in [22] analyzes the internal activations of the network in order to find anomalies. A similar idea is investigated in [23]. Neurons that are supposedly less useful for classification are discarded at inference time. In [24], the network is reverse-engineered with an algorithm that estimates a candidate trigger injected in the model. While all of these defense strategies require white-box access to the neural network model, the approach in [25] relies on black-box queries to the model under investigation. The responses are statistically analyzed to detect whether the model is backdoored and whether a specific input sample contains a trigger.

2.2 Relation to prior work

Operating in a post-training scenario and in a black-box setting, our sensitivity analysis is somewhat comparable to the defense approach in [25]. However, the cited work studies multi-class problems in image recognition (MNIST and CIFAR10, 10 classes in each case), while we address a face recognition problem involving larger images and a higher number of classes (more than 2000), as detailed in Section 3.1. Also, the authors of [26] analyze responses of backdoored face recognition models, but with a different scope. They aim at finding defense strategies, while not evaluating the impact of possible transformations of the trigger signal. Our analysis is also related to the work in [12], which explores geometric transformations of an adversarial patch. However, this source is limited to evasion attacks and does not generalize to the causative attacks studied here.

More generally, our work contains conceptual similarities to studies in the field of digital watermarking, where the robustness of a watermark is evaluated with respect to distortions of the watermarked signal [27, 28]. While this

literature focusses on transformations of the entire input signal (i.e., after embedding), we process the trigger signal *before* embedding.

3 Method

Now, we describe our sensitivity analysis of the responses of backdoored neural networks with respect to variations in the trigger signal. We consider the domain of face recognition; thus, our input samples are images $\mathbf{x} \in \mathbb{R}^{H\cdot W\cdot C}$, where H, W, and C denote width, height, and the number of color channels, respectively. Using the notation introduced in Section 2, we study a pre-trained **s**-infected model $F(\cdot)$ targeted to a specific class c_t. For each input sample $\mathbf{x}$, $d(\mathbf{x}) = \arg\max_{c\in\{c_1,\dots,c_K\}} F(\mathbf{x})_c$ is the assigned class and $F(\mathbf{x})_{d(\mathbf{x})}$ is the classification confidence. The trigger $\mathbf{s}$ is also an image of the same size as $\mathbf{x}$, but it has an additional opacity layer with values in $[0,1]$; thus, $\mathbf{s} \in \mathbb{R}^{H\cdot W\cdot (C+1)}$.

We introduce a family of *transformation functions* $\phi_\theta : \mathbb{R}^{H\cdot W\cdot (C+1)} \longrightarrow \mathbb{R}^{H\cdot W\cdot (C+1)}$ depending on a strength parameter $\theta \in \Theta$ used to transform the trigger before being embedded into an image.

The embedding function is defined as a linear blending operation with parameter $\lambda \in [0,1]$ between the image $\mathbf{x}$ and the trigger $\mathbf{s}$ that also encompasses the opacity layer of $\mathbf{s}$. By explicitly indicating the image indices as subscripts, the embedding function is given by:

$$\Psi_\lambda(\mathbf{x},\mathbf{s})_{h,w,c} = (1-\lambda\cdot\mathbf{s}_{h,w,C+1})\cdot\mathbf{x}_{h,w,c} + \lambda\cdot\mathbf{s}_{h,w,C+1}\cdot\mathbf{s}_{h,w,c} \quad (1)$$

for $h = 1,\dots,H$, $w = 1,\dots,W$, $c = 1,\dots,C$, where the index $C+1$ in the third dimension refers to the opacity layer.

Given a dataset of clean testing samples $\mathbf{X}$, for each transformation, we create a version $\Psi_\lambda(\mathbf{X},\phi_\theta(\mathbf{s}))$ of $\mathbf{X}$ containing $\phi_\theta(\mathbf{s})$-infected samples. When $\phi_\theta(\cdot)$ is the identity function, we are in the baseline case where the model is both trained and tested with **s**-infected images. Otherwise, a mismatch occurs and we measure its impact on the model performance.

For each dataset variant $\Psi_\lambda(\mathbf{X},\phi_\theta(\mathbf{s}))$, we compute the following performance metrics:

1. *ACC*: accuracy (rate of $\phi_\theta(\mathbf{s})$-infected samples assigned to the correct class);
2. *ASR*: attack success rate (rate of $\phi_\theta(\mathbf{s})$-infected samples assigned to the target class c_t);
3. *CM*: average classification confidence over the images in $\Psi_\lambda(\mathbf{X},\phi_\theta(\mathbf{s}))$ (regardless of the class assignment);
4. *CCM*: average classification confidence over the images in $\Psi_\lambda(\mathbf{X},\phi_\theta(\mathbf{s}))$ assigned to the correct class;
5. *CTM*: average classification confidence over the images in $\Psi_\lambda(\mathbf{X},\phi_\theta(\mathbf{s}))$ assigned to the target class c_t.

3.1 Experimental setup

We use as $F(\cdot)$ the pre-trained models proposed in [10] and available at [29]. As mentioned in Section 2.1, in this approach to backdoor attacks, the adversary does not need access to the full training set. Instead, she finds a candidate trigger $\mathbf{s}$ by heuristic optimization and tunes an existing model by retraining it with additional **s**-infected samples. The authors of [10] placed a backdoor into the face recognition model proposed in [30], which is trained on the VGG Face Dataset [31] and outputs a probability vector with 2622 dimensions, one for each identity appearing in the training set.

For our own experiments, we consider as $\mathbf{X}$ two datasets of face images used in [10], denoted as **OR** and **EXT** dataset. The **OR** dataset is composed of 2622 JPEG images depicting distinct faces corresponding to the identities in the training dataset [31]. For the **OR** dataset, we can compute the accuracy of **s**-infected models (*ACC*, *CCM*). The values should be comparable to the performance of the original model. The **EXT** dataset contains 1000 JPEG images of faces from subjects who do *not* appear in the training set. This dataset can only be used to measure the effectiveness of the attack (*ASR*, *CTM*).

Our embedding function in Eq. (1) corresponds to the one used in [10] for training and evaluation. We adopt the choice of $\lambda = 0.7$ for all our experiments.

Two infected models for face recognition are released at [29]. Both yield an accuracy of ~ 0.75 on clean inputs of **OR**, which is 0.03 less than the clean original model. For our experiments, we select the stronger one in terms of *ASR*, i.e., the one based on a square-shaped trigger (see Fig. 2). It yields an *ASR* close to 0.9 on **s**-infected images under baseline conditions.

3.2 Transformations

The transformation families selected for our analysis are reported in Tables 1, 2, and 3. For the sake of clarity, we have grouped them in categories, namely geometric, occlusive, and color transformations. Each transformation is associated with an icon that will be used to annotate the experimental results.

For the rotation, resizing, and sharing operations, we always align the centers of the baseline square-shaped and transformed triggers in the resulting image before blending. In these cases, the trigger images also undergo resampling and interpolation processes, which introduce additional artifacts in the signal [32, 33].

For contrast, sharpness adjustment, and median filtering, we employed the implementation provided in the Python Pillow library (v6.10). The "fading to grayscale" transformation linearly blends the trigger image with its grayscale version.

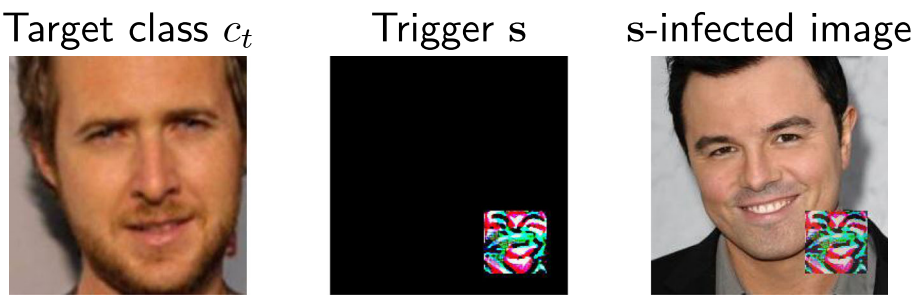


Fig. 2 Visualization of the target class and the square-shaped trigger used in [10]

3.3 Embedding process variants

Figure 3 visualizes the process applied to create $\Psi_\lambda(\mathbf{x}, \phi_\theta(\mathbf{s}))$ for each clean image $\mathbf{x}$. The pipeline adopted from [10] stores the test images $\Psi(\mathbf{X}, \phi_\theta(\mathbf{s}))$ in JPEG format with default quality factor 75. However, this operation introduces further distortion and artifacts, as widely investigated in the field of digital image forensics [34, 35]. In order to assess the effect of lossy post-compression on the performance of the backdoored classifier, we repeat our experiments without JPEG compression and decompression (i.e., skipping the red area in Fig. 3) and feeding the **s**-infected model directly with the output of the embedding function. Due to limitations of the data source, we cannot avoid that the clean images are pre-compressed with JPEG in all experiments.

4 Results

We report the main results for different trigger transformations in Section 4.1. Then, we move on to the exploration of causes and the validation in the physical domain. Section 4.2 reports the impact of the JPEG post-compression. Section 4.3 sheds light on the interaction of the trigger with image content, using a breakdown of the dataset by the amount of overlap between the default trigger and the depicted face.

Table 1 Geometric transformations

Name	Icon	Parameter θ
Rotation		Counter-clockwise rotation angle
Resize		Resizing factor
Horizontal shearing		Angle between left (right) trigger side and vertical axis
Vertical shearing		Angle between top (bottom) trigger side and horizontal axis
Horizontal shifting		Number of pixel to the right
Vertical shifting		Number of pixel to the bottom
Diagonal shifting		Number of pixel to the right and to the bottom

4.1 Trigger transformations

Figure 4 reports the results for applying the different transformations to the trigger before embedding. For each transformation, we selected a suitable parameter range Θ by inspecting preliminary results from a small subset of the sample. The plots in the left column show success rates (*ACC* and *ASR*) as a function of the strength of the transformation. The average confidence values (*CM*, *CCM*, *CTM*) appear on the right. The graphs are best read by interpreting the interaction of the *ACC* and *ASR* metrics along the horizontal axis, and with respect to the baseline (identity function), which is marked in with a gray vertical line in each plot (*ACC*=0.10 and *ASR*=0.89).

We summarize our observations as follows:

1. *Monotonicity:* for most of the transformations, the *ASR* decays monotonically when moving away from the baseline case, whereas the *ACC* increases towards the accuracy obtained by the model in case of clean inputs (0.75), as expected. Consistent trends are also observed for the *CCM* and *CTM* metrics. However, there are exceptions. For contrast enhancement, increasing opacity, and shifting the

Table 2 Occlusive transformations

Name	Icon	Parameter θ
Top-left diagonal occlusion		Ratio of pixels removed from the square-shaped area of the trigger
Bottom-right diagonal occlusion		
Inner rectangular occlusion		
Outer rectangular occlusion		
Random occlusion		

Table 3 Color transformations

Name	Icon	Parameter θ
Contrast adjustment		$\theta \in \mathbb{R}^+$ $\theta = 0 \rightarrow$ solid grey image; $\theta = 1 \rightarrow$ identity function $\theta > 1 \rightarrow$ image with higher contrast
Brightness adjustment		$\theta \in [-1, 1]$ $\theta = -1 \rightarrow$ solid black image;
Median filtering		$\theta \in \{3, 5, 7, 9\}$ size of the window filter $\theta = 0 \rightarrow$ identity function; $\theta = 1 \rightarrow$ grayscale image;
Sharpness adjustment		$\theta \in [0, 3]$ $\theta = 0 \rightarrow$ blurred image; $\theta = 1 \rightarrow$ identity function; $\theta = 3 \rightarrow$ sharpened image
Opacity adjustment		$\theta \in [0, 1]$ multiplication factor of the opacity layer $\theta = 0 \rightarrow$ transparent image; $\theta = 1/\lambda \rightarrow$ identity function;
Fading to grayscale		$\theta \in [0, 1]$ linear blending factor $\theta = 0 \rightarrow$ identity function; $\theta = 1 \rightarrow$ grayscale image;

trigger towards the centre of the image, the *ASR* increases after the application of $\phi_\theta(\cdot)$. Interestingly, all shifting transformations are not monotonic around the baseline. They exhibit local maxima around 0 and at multiples of 8, which could be related to JPEG compression artifacts. Recall that the JPEG format cuts images into blocks of 8×8 pixels before quantizing in the frequency domain. A representation of the trigger alignment with respect to the JPEG grid is reported in Fig. 5. Sharpness adjustment stands out in having almost no impact on the attack. A speculative explanation is that sharpness is modified by linear filtering. The effect might be neutralized in the convolutional layers of the DNN.

2. *Symmetry:* some transformations are defined symmetrically around the baseline. In case of brightness adjustments, shearing, and rotation (although the last one is not entirely displayed in the plot), this symmetry also holds for the response of the metrics. Only resizing exhibits a notable asymmetry: downsizing is much more impactful than upsizing.
3. *Slope*: for occlusive transformations, we can compare the slope of the *ASR* decay under the same ratio of removed pixels θ. We observe that occluding the bottom-right part or the outer part of the square trigger is equivalent to randomly removing the same number of pixels. However, the upper-left and inner parts of the trigger have a much bigger impact when removed. This indicates that certain parts of the trigger signal are more relevant than others. The sensitivity of the upper-left part may also relate to the image content covered by the trigger.
4. *Hitting the target*: For each combination of transformation and parameter value tested, the *ACC* and *ASR* essentially sum up to 1. This means that almost every misclassification goes to the target class c_t.

For space constraints, the results for the **EXT** dataset are reported in Appendix A for the geometric transformations, as they will be of interest in Section 5.1. The *ASR* is consistently higher in the **EXT** dataset. This holds already for the baseline case (where the *ASR* is close to 1.0) and leads to a generally slower decay, possibly caused by the absence of face information that are actually known to the classifier. Tests on all the transformations show that we can draw the same conclusions in terms of monotonicity, symmetry, slope, and hitting the target. Given this high level of consistency, we deem it justified to focus on the **OR** dataset for the following two subsections.

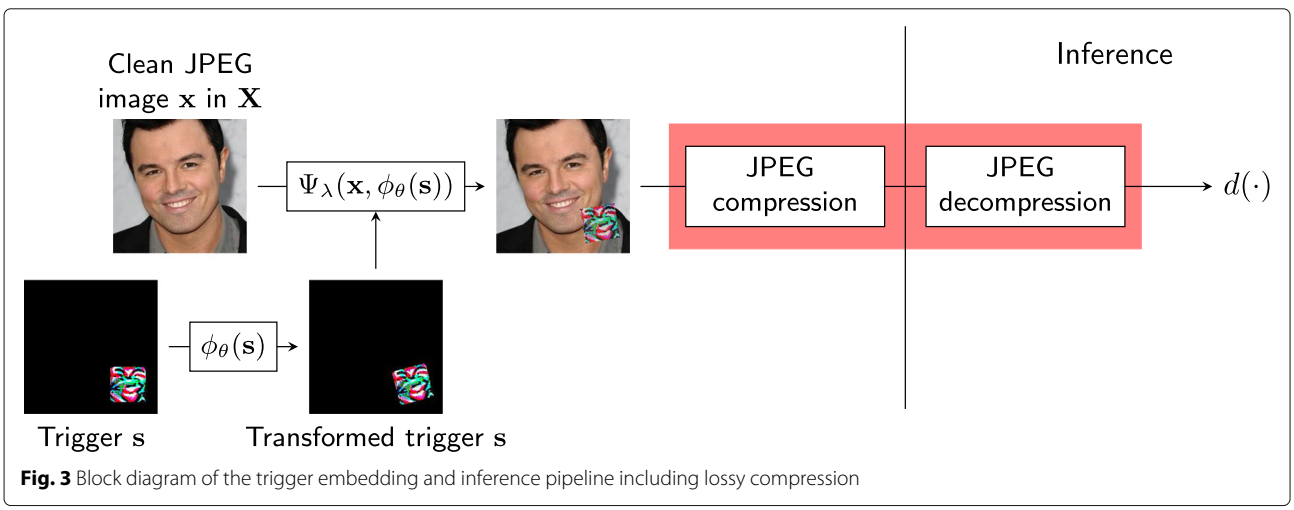


Fig. 3 Block diagram of the trigger embedding and inference pipeline including lossy compression

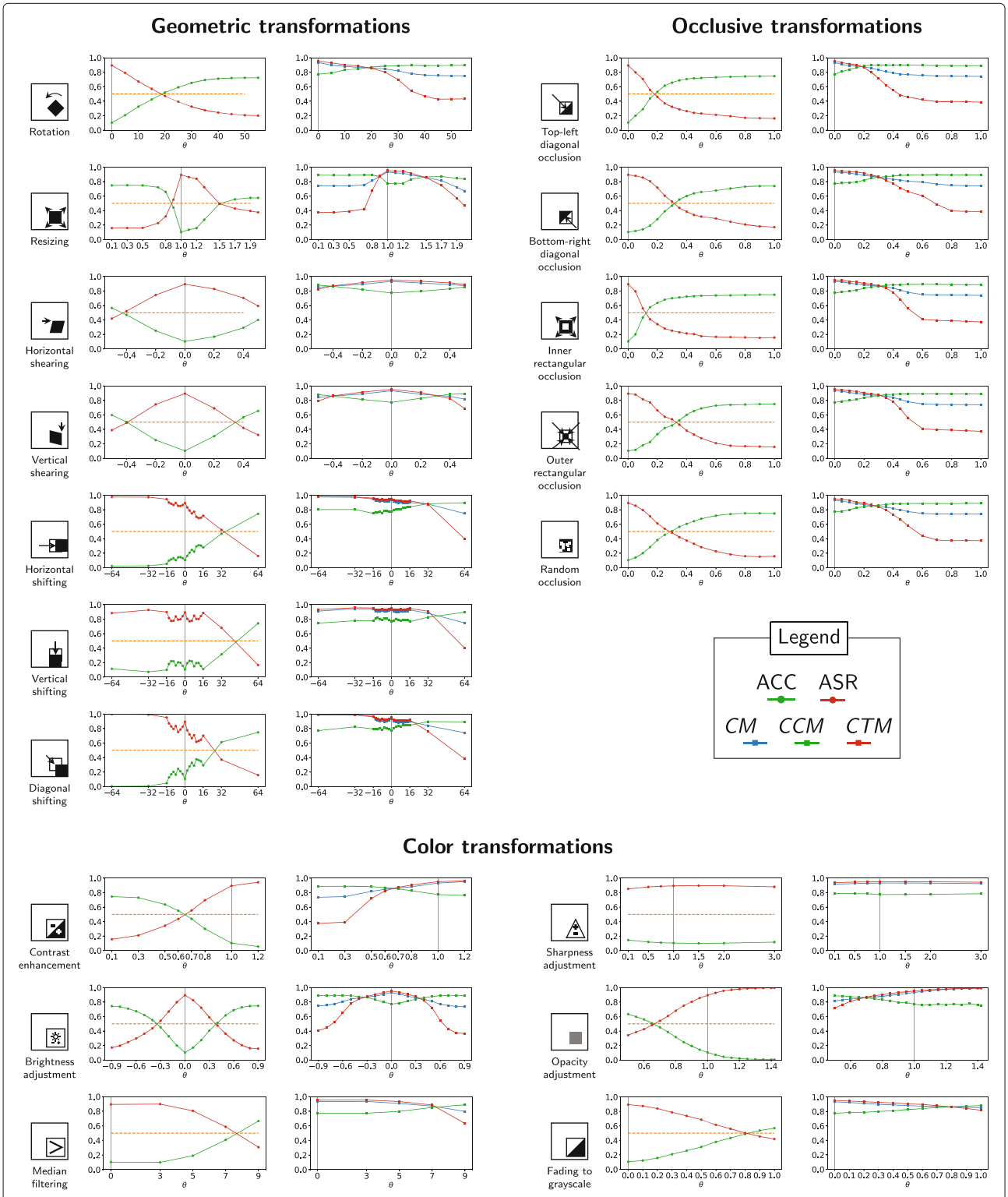


Fig. 4 Experimental results for different transformation functions applied to the **OR** dataset. In each case, represented by the corresponding icon, we plot the performance metrics as functions of the transformation parameter θ. The left plots report *ACC* and *ASR* (in green and red, respectively), and the right plots reports the average confidence values *CM*, *CCM*, and *CTM* (in blue, green, and red, respectively). The baseline case where $\phi_\theta(\cdot)$ is the identity function is indicated with a vertical gray line. The horizontal orange line marks the level of *ACC* and *ASR* equal to 0.5

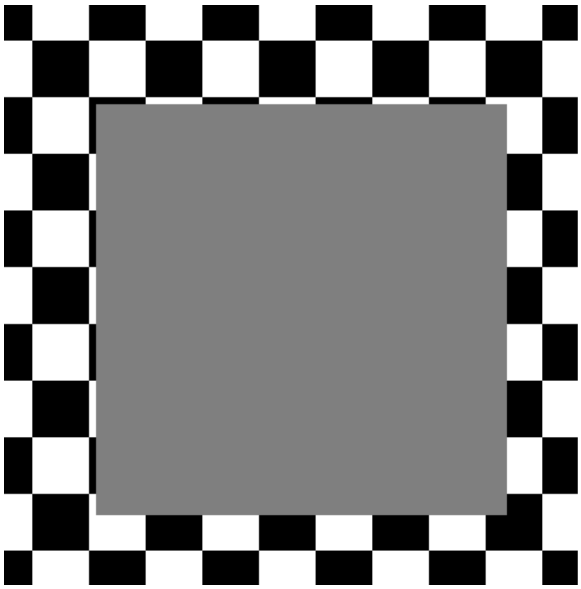


Fig. 5 Trigger alignment with respect to the 8 × 8 JPEG grid. The top-left corner of the squared trigger is misaligned by one pixel to the right and one pixel to the bottom

4.2 JPEG recompression

The JPEG post-compression after the embedding function, as depicted in Fig. 3, has a small but stable impact on the success rate of the backdoor attack. We measure this by subtracting the attack success rate obtained when skipping the post-compression phase depicted in Fig. 3 from the *ASR* computed in Section 4.1, thus obtaining the metric Δ_{NPC} (no post-compression).

Table 4 reports the statistics of Δ_{NPC}. Different columns report the average, minimum, and maximum Δ_{NPC} observed over different parameters of the considered transformations, which are arranged row-wise. The values are concentrated in the interval $[-0.01, 0.01]$ and consistently show that the attack is marginally more successful if the post-compression is omitted. This refutes hypotheses suggesting that the backdoor attack picks up the JPEG artifacts occurring specifically at the boundary between the trigger and the background, as the *ASR* at baseline conditions decreases by 0.68%.

4.3 Overlap with image content

Next, we seek to identify potential causes for the overall response of the **s**-infected model, also in the light of the observations made in Section 4.1. In particular, we study the impact of the location of the squared trigger area (which is fixed in all **s**-infected samples) with respect to the depicted face (which varies across images). We split the **OR** dataset into three exclusive categories representing the level of overlap between the squared trigger and the face, namely:

Table 4 Statistics of Δ_{NPC} over different parameters when JPEG recompression is omitted

Transformation	JPEG "advantage" Δ_{NPC}		
	Avg	Min	Max
Rotation	− 0.0049	− 0.0110	− 0.0007
Resizing	− 0.0034	− 0.0122	0.0034
Horizontal shearing	− 0.0065	− 0.0091	− 0.0045
Vertical shearing	− 0.0039	− 0.0068	− 0.0019
Horizontal shifting	− 0.0044	− 0.0102	0.0000
Vertical shifting	− 0.0053	− 0.0110	− 0.0003
Diagonal shifting	− 0.0064	− 0.0186	0.0015
Top-left diagonal occlusion	− 0.0022	− 0.0076	− 0.0003
Bottom-right diagonal occlusion	− 0.0057	− 0.0137	0.0007
Inner rectangular occlusion	− 0.0038	− 0.0148	0.0000
Outer rectangular occlusion	− 0.0036	− 0.0095	0.0011
Random occlusion	− 0.0062	− 0.0095	0.0003
Contrast adjustment	− 0.0023	− 0.0068	0.0019
Median filtering	−0.0037	−0.0045	−0.0026
Brightness adjustment	− 0.0041	− 0.0133	0.0007
Sharpness adjustment	− 0.0057	− 0.0076	− 0.0034
Fading to grayscale	−0.0046	−0.0091	0.0000
Opacity adjustment	− 0.0032	− 0.0086	0.0049

- FREE: no overlap at all.
- TOUCH: overlap with the face boundary, but not with the mouth.
- OVERLAP: overlap with the face boundary and the mouth.

We do this by running a face landmark detector[1] on the 2622 clean face images and counting how many and which landmarks fall into the squared area occupied by the trigger after embedding (in the baseline case). Figure 6 states the category definitions with respect to the landmark locations, reports the number of images in each category, and shows one illustrating example.

Figure 7 reports the results for selected transformations of Section 4.1 broken down by the three categories. Observe that the curves for the TOUCH and OVERLAP categories consistently co-move, whereas the FREE curves are significantly distant. This shows that having no interaction with the face information has indeed improved the attack's effectiveness for this backdoored model. This gap exists under baseline conditions and is generally preserved when θ varies.

The analysis of the occlusive transformations (top right plots in Fig. 7) reveals that the gap is even more pronounced when $\theta = 1$ (i.e, when the squared area is completely removed). This suggests that the infected model has problems in correctly classifying the faces in the FREE category even if they are clean. To investigate this further, we replicated the break-down by category on clean images only, using the clean model, i.e., the model without backdoor. Table 5 reports the performance

[1] We use the `dlib` library.

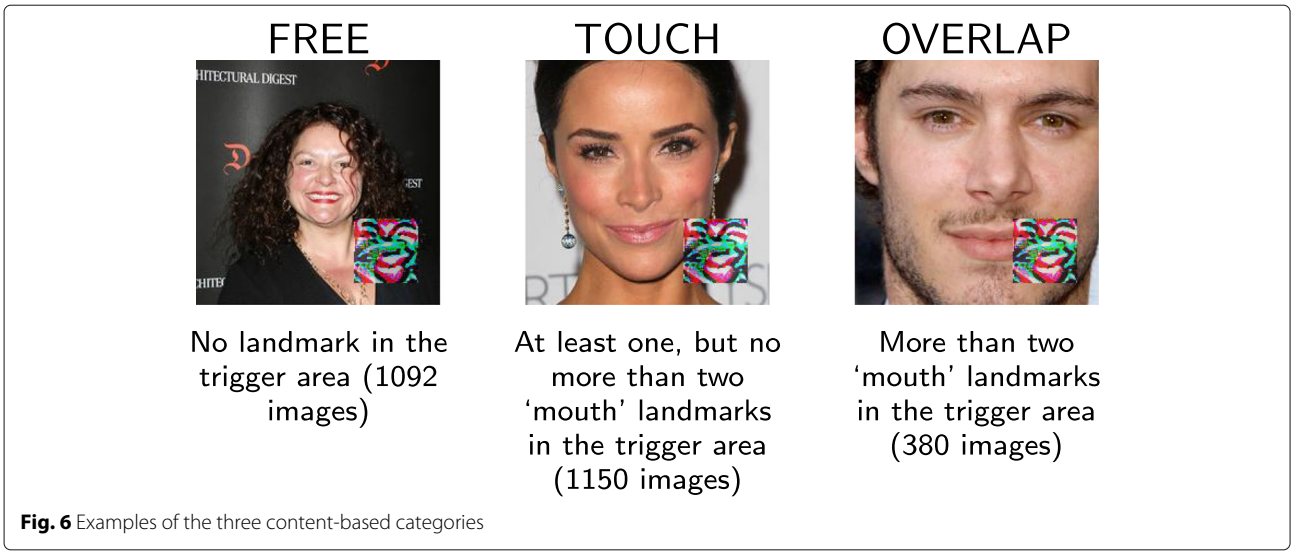


Fig. 6 Examples of the three content-based categories

metrics for these tests. Observe that the FREE category (first column) stands out also in this case, as the classification accuracy for it is significantly lower than in the other two categories (second and third column). It seems that the clean model [30] that served as basis for the attack [10] has more general difficulties in the recognition of small faces. Specifically for samples on which the clean model was already not very accurate, the poisoning process biased the decision on misclassified samples towards the target class c_t. A closer inspection of this bias is an interesting topic for future investigations, as it might inform better defenses by detecting the presence of a backdoor in DNNs.

5 Validation

This section presents the results of two experiments validating the analysis in Section 4. First, in order to evaluate whether the results obtained can be related to practical scenarios, we have conducted an analysis where the trigger is passed to the DNN through a real object captured in the physical domain (Section 5.1). Second, we have extended our sensitivity test to another known task in automatic face analysis, i.e., the recognition of the subject's age. This will allow us to observe differences and similarities with respect to the findings reported in Section 4.1, as detailed in Section 5.2.

5.1 Physical domain

The creation of adversarial attacks in the physical domain has been studied in [36–38]. The goal is to assess the capabilities of an attacker to compromise learning-based systems by performing specific operations in the real world. For the case of face recognition, such an analysis is particularly interesting, as face recognition systems typically

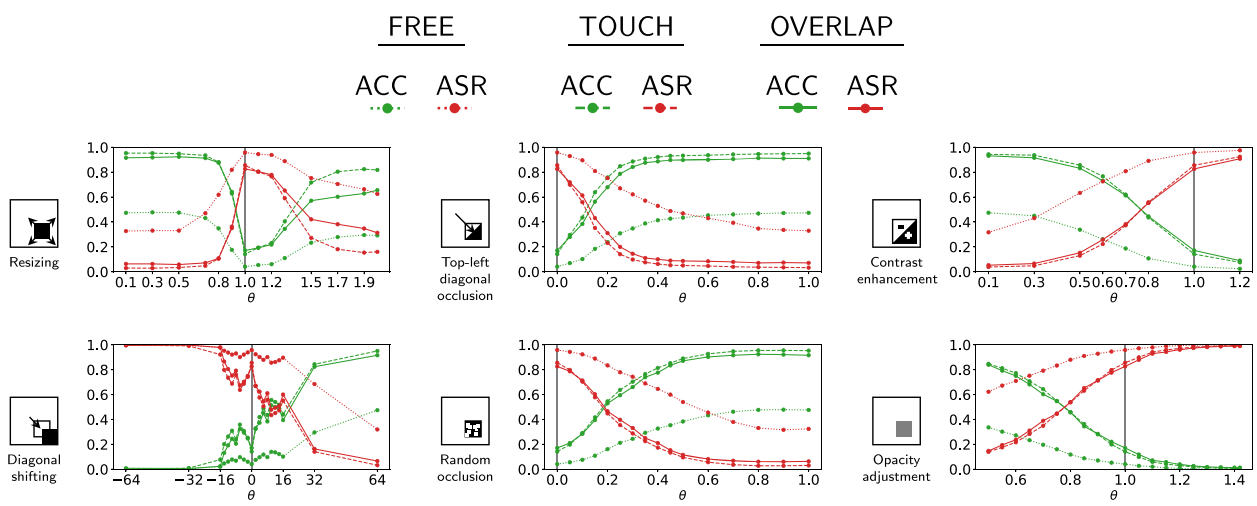


Fig. 7 Overlap with image content: *ACC* and *ASR* values for selected transformations aggregated by category

Table 5 Performance of the clean (top) and **s**-infected (bottom) models on the clean version of the **OR** dataset broken down by the degree of overlap

		FREE	TOUCH	OVERLAP
Clean	*ACC*	0.54	0.98	0.97
model	*ASR*	0.00	0.00	0.00
s-infected	*ACC*	0.48	0.95	0.91
model	ASR	0.32	0.03	0.06

capture live probes of the subjects to be identified, which could show a known trigger signal in order to bypass the backdoored system.

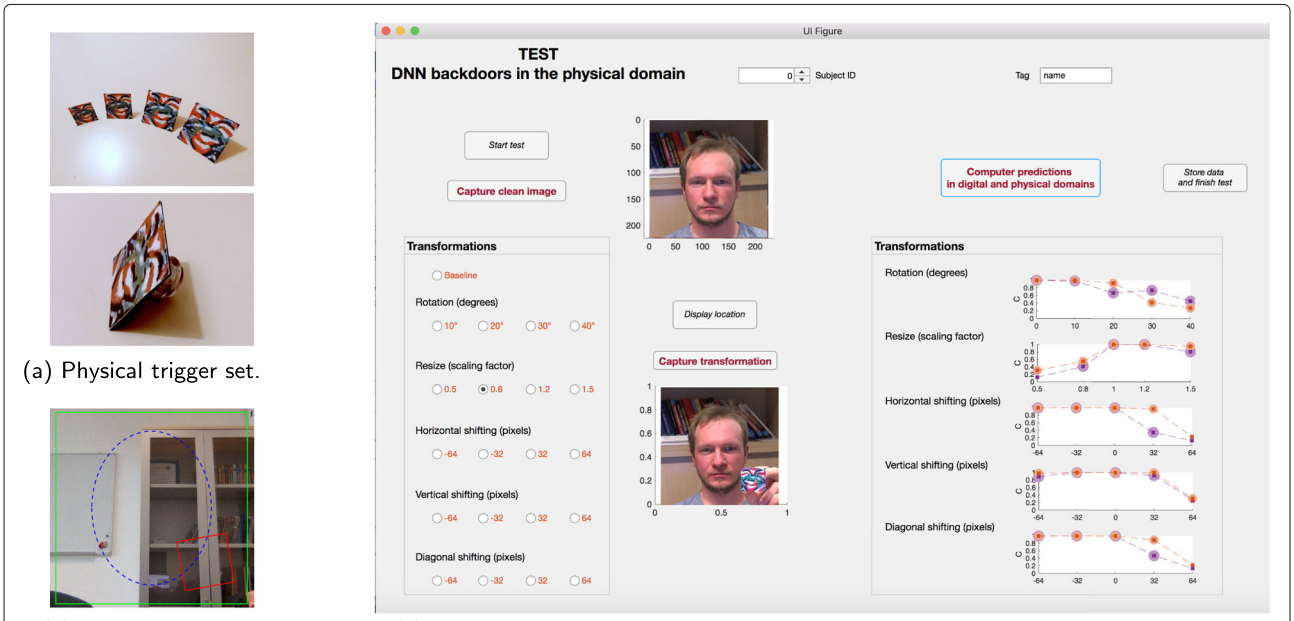
Inspired by these approaches, we have created trigger objects by printing out the image of the squared trigger with different resolutions and placing them on a rigid background. A holder has been mounted on the back of each printed trigger, so that a user could easily hold it in his/her hand (see Fig. 8a). For the sake of clarity, in the following, we will use the terms *digital* and *physical* trigger to indicate the digital trigger image and the trigger object, respectively.

A number of probe images have been acquired from volunteers, who positioned the physical trigger in front of a webcam window next to their face (see Fig. 8b). A subset of the geometric transformations has been selected for this analysis, namely rotation, resizing, as well as horizontal, vertical, and diagonal shifting.

Figure 8c shows the interface designed to capture probe images. Users are in front of a webcam and can see their live acquisition. First, a clean reference image is captured. This serves as background to insert a digital trigger with the selected geometric transformations, as described in Section 3. Then, users are asked to hold the physical trigger and sequentially place it in the positions corresponding to the same geometric transformations as displayed in the webcam window. Whenever the position matches the red frame, a probe image is captured. Pairs of images containing digital and physical triggers are then passed to the $\phi_\theta(\mathbf{s})$-infected model, allowing a pairwise comparison of the key outputs (class and confidence).

Acquisitions from 8 volunteers have been collected in different environments and heterogeneous illumination conditions. Since the identity of the volunteers are not known to the model, the situation corresponds to the case of the **EXT** dataset. Thus, we should refer to the results in Appendix A, and we cannot compute the classification accuracy *ACC*.

The results are summarized in Fig. 9. It is worth observing the similarity of the *ASR* when using the physical or digital trigger, respectively, under the same transformation using multiple strength parameters. This confirms that the capabilities of bypassing the backdoored



(a) Physical trigger set.

(b) Webcam window.

(c) Interface used for the acquisition of probe images. The left part supports the acquisition of probe images with different trigger transformations, while the right part display the outcomes of the acquired images when fed to the back-doored model.

Fig. 8 Implementation of the validation experiments in the physical domain

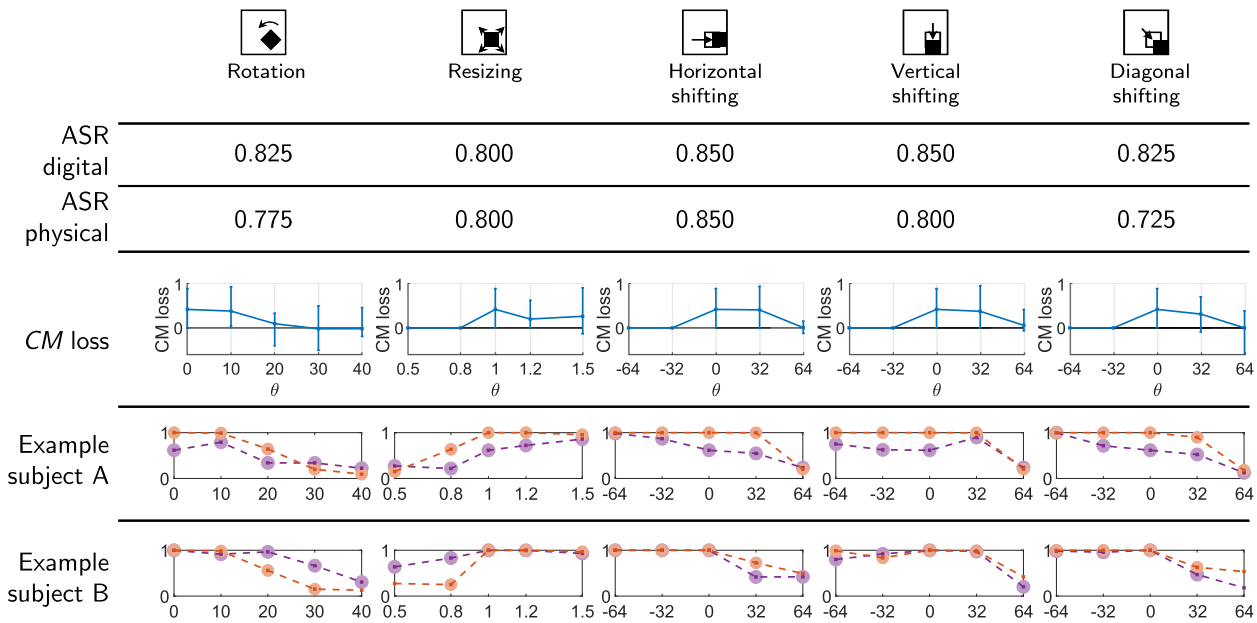


Fig. 9 Results for pairs of digital and physical triggers (transformations in columns; outcomes in rows). The first two rows report the average *ASR* over all 8 subjects and tested transformation strengths. The third row reports the average loss in classification confidence *CM* when comparing digital and physical triggers for the same transformation (i.e., a positive loss means decreasing confidence in the physical domain), plotted as a function of the strength parameter (horizontal axis). Vertical bars indicate minimum and maximum loss at each horizontal support point. The last two rows report results for two sample subjects (red for digital, purple for physical trigger). The vertical position indicates *CM* and bubbles indicate successful attacks (i.e., assignment of the target class)

model reported for the digital trigger generalize to the physical trigger.

However, the confidence of the network's decision is in general lower when physical triggers are used, as it is shown by the plots of the *CM* loss in the third row. With this respect, we found noteworthy between-subject differences, as exemplified in the bottom two rows. Subject A exhibits a discrepancy in the decision confidence already in the baseline trigger position, which extends to the transformations. By contrast, the confidence values for subject B are aligned at the baseline position but diverge for intermediate transformation strengths onwards.

5.2 Age recognition task

We have considered the age recognition task addressed in [39], where the goal is to assign each face image to the correct age range, where 7 different age ranges are considered (0-2, 4-6, 8-13, 15-20, 25-32, 38-43, 48-53, 60+). We used the backdoored model for this task that is available at [29], and it is poisoned to assign the class 0-2 if the trigger shown in Fig. 10 appears. The corresponding dataset is composed of 1000 images with size $224 \times 224 \times 4$.

Note that this experimental setup shares similarities with the one considered in Section 4: the infected model has been re-trained with the same poisoning method proposed in [10] for the face recognition task; a square trigger is adopted also in this case, although the pattern used is different as it is determined depending on the original clean model. However, the architecture used for this task is quite different than the face recognition case. Moreover, the nature of the classification problem is also radically different, both in terms of number of classes and intrinsic difficulties due to the data variability.

Fig. 10 Example of trojaned image for the age recognition task

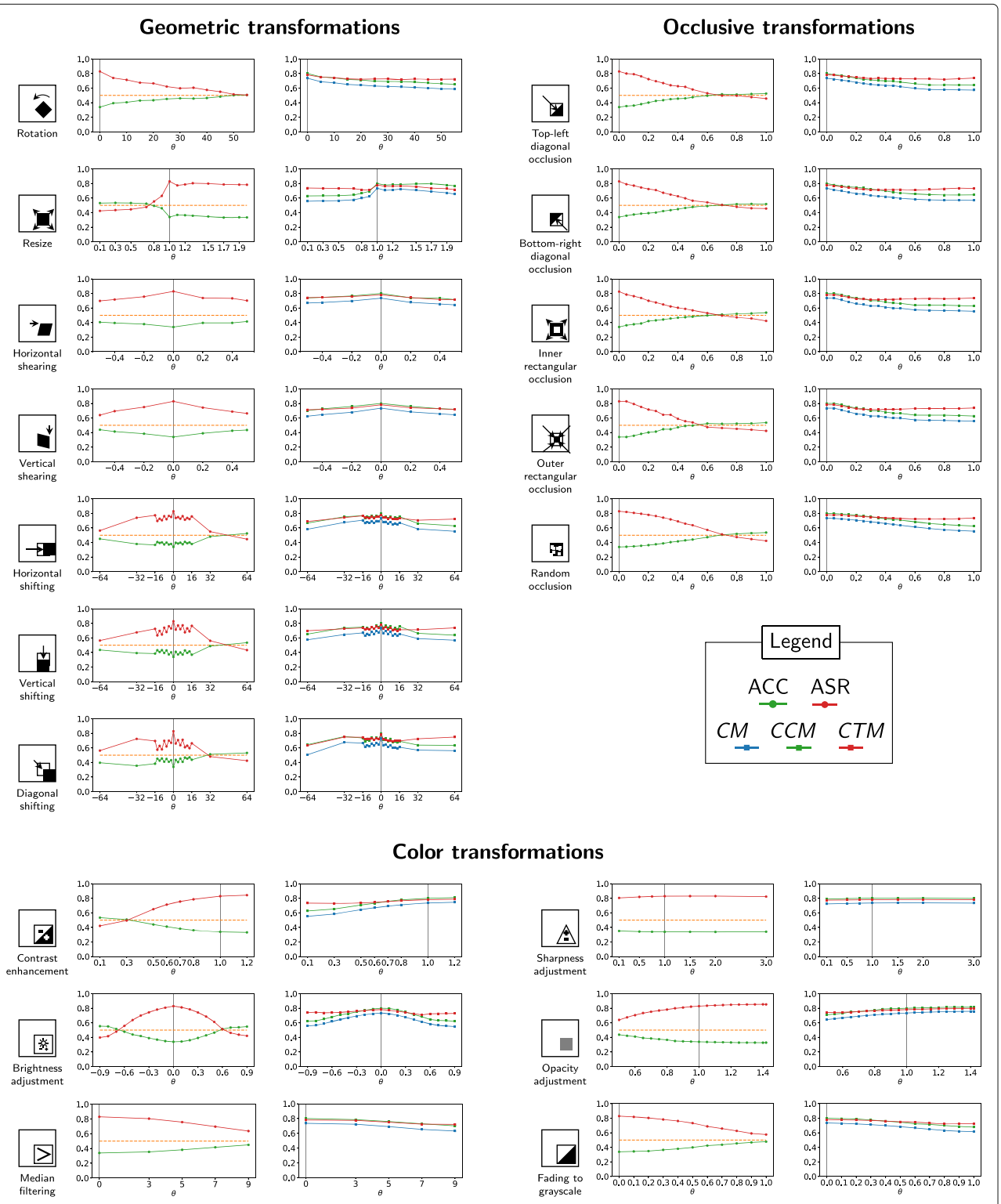


Fig. 11 Experimental results for different transformation functions for the age recognition task. In each case, represented by the corresponding icon, we plot the performance metrics as functions of the transformation parameter θ. The left plots report *ACC* and *ASR* (in green and red, respectively), and the right plots reports the average confidence values *CM*, *CCM*, and *CTM* (in blue, green, and red, respectively). The baseline case where $\phi_\theta(\cdot)$ is the identity function is indicated with a vertical gray line

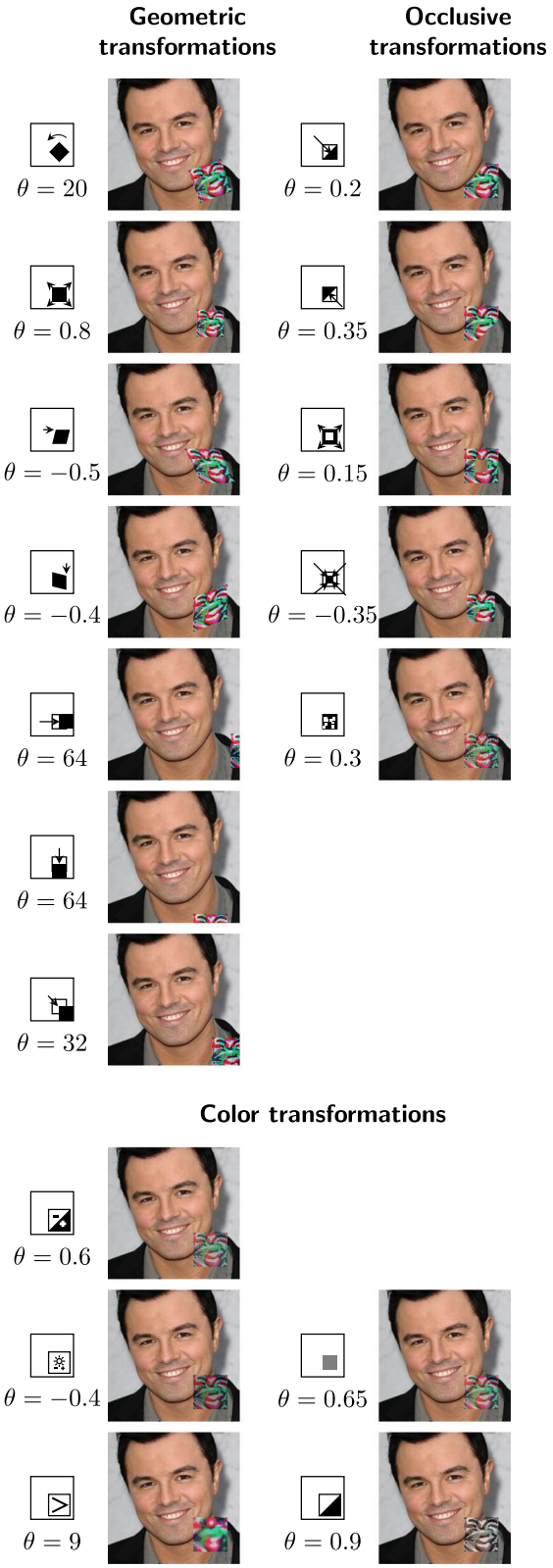


Fig. 12 Visualization of transformed triggers for which the empirical *ASR* is below 0.5 over the entire dataset. For each transformation, the closest parameter value with respect to the baseline is depicted and reported below the icon

We performed the same analysis as in Section 4.1, so that we can assess potential similarities and differences in terms of sensitivity of the trigger perturbations. Results are reported in Fig. 11.

In terms of points 1 (monotonicity), 2 (symmetry), and 4 ("hitting the target"), we can get to very similar conclusions as in Section 4.1. One exception is given by the shifting operations that here behave much more symmetrically than for the face recognition case; thus, we do not find that moving the trigger in a more central position in the image is beneficial for attack success. Moreover, regarding point 3 (slope), the different occlusive transformations all have a very similar impact at the same ratio of pixels removed. This is related to a more general behavior observed in Fig. 11, i.e., that the *ASR* curves are less steep than what observed in Section 4.1. In fact, the *ASR* is already below 85% at the baseline trigger state (i.e., the best condition for the attack), but it never decreases below 40% even when the trigger is absent (see complete occlusions). This corroborates the observation made in Section 4.3, where we found that the bias of the infected model towards the target class in the absence of the trigger is particularly strong when the original model is not accurate on clean data. Here, the initial accuracy of the original model on clean data is only around 55% and, after poisoning, almost all the misclassifications assign the target class.

6 Conclusions

To the best of our knowledge, we have conducted the first sensitivity analysis of a selected backdoored neural network with the objective to evaluate the impact of processing operations applied to the trigger signal on the effectiveness of the attack. Figure 12 offers a visual summary of the critical values for the strengths of the transformations. We can observe that the attack is somewhat robust to trigger transformations since we did not find any visually imperceptible transformation that reduces the success rate to less than 0.5.

While these conclusions are certainly informative, the most constraining limitation of this work is the focus on powerful yet limited instances of backdoored models for face recognition. This limits the generalizability of our conclusions, as it happens for many studies in this emerging field with so many unknowns. However, when applying the same trigger transformations to a poisoned model dealing with a different classification problem, similar patterns in terms of monotonicity, symmetry, and classification error distribution have been observed, although with some specificities. Moreover, these findings are generally confirmed by experiments performed in the physical domain with trigger objects.

We see our results as a first valuable step, a benchmark for future work, and possibly as a source for inspiration to study novel approaches to defense. More specific contributions of this work include raising awareness in DNN-based security applications for the effect of the compression pipeline and image processing manipulations with low semantic impact, an issue that is rarely considered in computer vision and systems security, whereas highly studied in multimedia security [32, 34, 40, 41]. A more systematic study of the interaction between image content and trigger signals relates to the more general question of how superimposed signals interact with networks' decisions, a topic of interest in deep network interpretability. Linking these streams of research could help to form a better understanding of the limits and potentialities of backdoor attacks against DNNs. Moreover, investigations should be carried out exploring how much the attacker can prevent the decrease of success rate by applying trigger transformations already during the poisoning phase.

Supplementary information

Supplementary information accompanies this paper at https://doi.org/10.1186/s13635-020-00104-z.

Additional file 1: Appendix A.

Abbreviations

DNN: Deep neural network; JPEG: Joint Photographic Experts Group

Acknowledgements

The authors thank Jonas Schöpf for his research assistance and Daniel Woods for useful discussions.

Author's contributions

CP conceived and designed this study in consultation with RB. CP supervised a pre-study, carried out the main experiments, and positioned the work in the current state of research. RB helped interpreting the results and made suggestions for the analysis. Both authors have contributed in writing the manuscript and have approved the submitted version.

Funding

This research was funded primarily by Archimedes Privatstiftung, Innsbruck.

Availability of data and materials

The experimental results of this study are based on the benchmark image datasets used in [10] and available at [29]. The code is available from the corresponding author on request.

Competing interests

The authors declare that they have no competing interests.

Author details

[1]Department of Information Engineering and Computer Science, University of Trento, Via Sommarive 9, 38123 Trento, Italy. [2]Department of Computer Science, University of Innsbruck, Technikerstraße 21A, 6020 Innsbruck, Austria.

Received: 4 October 2019 Accepted: 17 March 2020
Published online: 23 June 2020

References

1. T. J. Sejnowski, *The deep learning revolution*. (MIT Press, Cambridge, Massachusetts, 2018)
2. Y. Taigman, M. Yang, M. Ranzato, L. Wolf, in *2014 IEEE Conference on Computer Vision and Pattern Recognition*. Deepface: Closing the gap to

human-level performance in face verification (IEEE, 2014). https://doi.org/10.1109/cvpr.2014.220
3. B. Biggio, G. Fumera, F. Roli, Security evaluation of pattern classifiers under attack. IEEE Trans. Knowl. Data Eng. **26**(4), 984–996 (2014)
4. M. Barreno, B. Nelson, R. Sears, A. D. Joseph, J. D. Tygar, in *Proceedings of the 2006 ACM Symposium on Information, computer and communications security - ASIACCS '06*. Can machine learning be secure? (ACM Press, 2006). https://doi.org/10.1145/1128817.1128824
5. M. Barreno, B. Nelson, A. D. Joseph, J. D. Tygar, The security of machine learning. Mach. Learn. **81**(2), 121–148 (2010)
6. C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. J. Goodfellow, R. Fergus, Intriguing properties of neural networks. CoRR. **abs/1312.6199** (2013). arXIv
7. L. Muñoz-González, B. Biggio, A. Demontis, A. Paudice, V. Wongrassamee, E. C. Lupu, F. Roli, in *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security - AISec '17*. Towards poisoning of deep learning algorithms with back-gradient optimization (ACM Press, 2017). https://doi.org/10.1145/3128572.3140451
8. M. Jagielski, A. Oprea, B. Biggio, C. Liu, C. Nita-Rotaru, B. Li, in *2018 IEEE Symposium on Security and Privacy (SP)*. Manipulating machine learning: poisoning attacks and countermeasures for regression learning (IEEE, 2018). https://doi.org/10.1109/sp.2018.00057
9. A. Shafahi, W. R. Huang, M. Najibi, O. Suciu, C. Studer, T. Dumitras, T. Goldstein, in *Conference on Neural Information Processing Systems (NIPS)*. Poison frogs! targeted clean-label poisoning attacks on neural networks, (2018)
10. Y. Liu, S. Ma, Y. Aafer, W.-C. Lee, J. Zhai, W. Wang, X. Zhang, in *Proceedings 2018 Network and Distributed System Security Symposium*. Trojaning attack on neural networks (Internet Society, 2018). https://doi.org/10.14722/ndss.2018.23291
11. M. Sharif, S. Bhagavatula, L. Bauer, M. K. Reiter, in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, CCS '16*. Accessorize to a crime: real and stealthy attacks on state-of-the-art face recognition (ACM, New York, 2016), pp. 1528–1540. https://doi.org/10.1145/2976749.2978392
12. T. B. Brown, D. Mané, A. Roy, M. Abadi, J. Gilmer, Adversarial patch. CoRR. **abs/1712.09665** (2017). http://arxiv.org/abs/1712.09665
13. T. Gu, B. Dolan-Gavitt, S. Garg, in *Machine Learning and Computer Security (MLSec) NIPS Workshop*. Badnets: identifying vulnerabilities in the machine learning model supply chain, (2017)
14. T. Gu, K. Liu, B. Dolan-Gavitt, S. Garg, Badnets: evaluating backdooring attacks on deep neural networks. IEEE Access. **7**, 47230–47244 (2019). https://doi.org/10.1109/ACCESS.2019.2909068
15. Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition. Proc. IEEE. **86**(11), 2278–2324 (1998)
16. Y. Yao, H. Li, H. Zheng, B. Y. Zhao, Regula sub-rosa: latent backdoor attacks on deep neural networks. CoRR. **abs/1905.10447** (2019). http://arxiv.org/abs/1905.10447
17. C. Liao, H. Zhong, A. C. Squicciarini, S. Zhu, D. J. Miller, Backdoor embedding in convolutional neural network models via invisible perturbation. CoRR. **abs/1808.10307** (2018). http://arxiv.org/abs/1808.10307
18. M. Barni, K. Kallas, B. Tondi, A new backdoor attack in CNNs by training set corruption without label poisoning. CoRR. **abs/1902.11237** (2019). http://arxiv.org/abs/1902.11237
19. A. Bhalerao, K. Kallas, B. Tondi, M. Barni, in *2019 IEEE 21st International Workshop on Multimedia Signal Processing (MMSP)*. Luminance-based video backdoor attack against anti-spoofing rebroadcast detection (IEEE, 2019). https://doi.org/10.1109/mmsp.2019.8901711
20. W. Guo, L. Wang, X. Xing, M. Du, D. Song, Tabor: A highly accurate approach to inspecting and restoring trojan backdoors in ai systems. ArXiv abs/1908.01763 (2019)
21. B. Tran, J. Li, A. Madry, in *Proceedings of the 32nd International Conference on Neural Information Processing Systems, NIPS'18*. Spectral signatures in backdoor attacks, (2018), pp. 8011–8021
22. B. Chen, W. Carvalho, N. Baracaldo, H. Ludwig, B. Edwards, T. Lee, I. Molloy, B. Srivastava, in *Artificial Intelligence Safety Workshop @ AAAI*. Detecting backdoor attacks on deep neural networks by activation clustering, (2019)
23. K. Liu, B. Dolan-Gavitt, S. Garg, in *Research in Attacks, Intrusions, and Defenses*. Fine-pruning: defending against backdooring attacks on deep neural networks (Springer, 2018), pp. 273–294. https://doi.org/10.1007/978-3-030-00470-5_13
24. B. Wang, Y. Yao, S. Shan, H. Li, B. Viswanath, H. Zheng, B. Y. Zhao, in *2019 IEEE Symposium on Security and Privacy (SP)*. Neural cleanse: identifying and mitigating backdoor attacks in neural networks, (2019), pp. 707–723. https://doi.org/10.1109/SP.2019.00031
25. Y. Gao, C. Xu, D. Wang, S. Chen, D. C. Ranasinghe, S. Nepal, STRIP: a defence against trojan attacks on deep neural networks. CoRR. **abs/1902.06531** (2019). http://arxiv.org/abs/1902.06531
26. E. Chou, F. Tramèr, G. Pellegrino, D. Boneh, Sentinet: Detecting physical attacks against deep learning systems. CoRR. **abs/1812.00292** (2018). http://arxiv.org/abs/1812.00292
27. F. A. P. Petitcolas, R. J. Anderson, M. G. Kuhn, in *Information Hiding (2nd International Workshop), LNCS 1525*, ed. by D. Aucsmith. Attacks on copyright marking systems (Springer, Berlin Heidelberg, 1998), pp. 219–239
28. M. Barni, F. Bartolini, T. Furon, A general framework for robust watermarking security. Sig. Process. **83**(10), 2069–2084 (2003)
29. TrojanNN. https://github.com/PurduePAML/TrojanNN. Accessed 2 June 2019
30. O. M. Parkhi, A. Vedaldi, A. Zisserman, in *British Machine Vision Conference*. Deep face recognition, (2015)
31. VGG Face Dataset. http://www.robots.ox.ac.uk/\texttildelowvgg/data/vgg_face/. Accessed 2 June 2019
32. D. Vàzquez-Padìn, F. Pèrez-Gonzàlez, P. Comesaña-Alfaro, A random matrix approach to the forensic analysis of upscaled images. IEEE Trans. Inf. Forens. Secur. **12**(9), 2115–2130 (2017)
33. C. Pasquini, R. Böhme, Information-theoretic bounds for the forensic detection of downscaled signals. IEEE Trans. Inf. Forens. Secur. **14**(7), 1928–1943 (2019)
34. R. Böhme, M. Kirchner, ed. by S. Katzenbeisser, F. Petitcolas. Information hiding (Artech House, Norwood, 2016), pp. 231–259
35. C. Pasquini, P. Schöttle, R. Böhme, G. Boato, F. Pèrez-Gonzàlez, in *Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security - IH&MMSec '16*. Forensics of high quality and nearly identical jpeg image recompression (ACM Press, 2016). https://doi.org/10.1145/2909827.2930787
36. K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, F. Tramèr, A. Prakash, T. Kohno, D. Song, in *Proceedings of the 12th USENIX Conference on Offensive Technologies, WOOT'18*. Physical adversarial examples for object detectors, (2018), p. 1
37. K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, D. Song, in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Robust physical-world attacks on deep learning visual classification (IEEE, 2018). https://doi.org/10.1109/cvpr.2018.00175
38. X. Chen, C. Liu, B. Li, K. Lu, D. Song, Targeted backdoor attacks on deep learning systems using data poisoning. CoRR. **abs/1712.05526** (2017). arXIv
39. E. Eidinger, R. Enbar, T. Hassner, Age and gender estimation of unfiltered faces. IEEE Transactions on Information Forensics and Security. **9**(12), 2170–2179 (2014)
40. M. Barni, A. Costanzo, E. Nowroozi, B. Tondi, in *2018 25th IEEE International Conference on Image Processing (ICIP)*. CNN-based detection of generic contrast adjustment with jpeg post-processing (IEEE, 2018). https://doi.org/10.1109/icip.2018.8451698
41. C. Pasquini, G. Boato, N. Alajlan, F. G. B. De Natale, A deterministic approach to detect median filtering in 1D data. IEEE Trans. Inf. Forens. Secur. **11**(7), 1425–1437 (2016)

Publisher's Note