APSIPA Transactions on Signal and Information Processing, 2025, 14, e204

Original Paper

Print and Scan Simulation for Adversarial Attacks on Printed Images

Nischay Purnekar[1*], Benedetta Tondi[1], Jana Dittmann[2] and Mauro Barni[1]

[1] *Department of Information Engineering and Mathematical Sciences, University of Siena, Siena, Italy.*

[2] *Department of Computer Science, Otto-von-Guericke University, Magdeburg, Germany*

ABSTRACT

Predictive AI with deep learning is vulnerable to adversarial examples—subtle, human-imperceptible modifications that can induce classification errors or evade detection. While most research targets digital adversarial attacks, many real-world applications require attacks to function in the physical domain. Physical adversarial examples must survive digital-to-analog and analog-to-digital transformations with minimal perturbation. In this paper, we investigate two white-box physical-domain evasion attacks. First, we target an AI-based source printer attribution system, which identifies the printer used to produce a printed document. This task is particularly challenging because the Print and Scan (P&S) process reintroduces printer-specific features, potentially nullifying the attack. To address this, we adopt Expectation Over Transformation, incorporating a realistic simulation of the P&S process using two Generative Adversarial Network models trained specifically for this purpose. To demonstrate the generality of our approach, we also apply it to attack a License Plate Detector. The crafted adversarial examples remain effective even after being printed and recaptured using a mobile phone camera. Experimental results confirm that our method significantly improves the

[*] Corresponding author: nischay.purnekar@student.unisi.it

Received 30 March 2025; revised 14 May 2025; accepted 15 May 2025
ISSN 2048-7703; DOI 10.1561/116.20250019

attack success rate in both applications, outperforming baseline approaches. These findings highlight the feasibility and effectiveness of robust physical-domain adversarial attacks across diverse computer vision tasks.

Keywords: Adversarial examples, generative adversarial networks (GANs), license plate detection, physical domain, print and scan simulation, source printer attribution

1 Introduction

Despite their effectiveness, predictive Artificial Intelligence (AI) systems based on Deep Learning (DL) are vulnerable to various malicious attacks, including adversarial examples [34], backdoor attacks [14] and inversion attacks [12] (for a taxonomy of possible attacks, see [35]). Adversarial examples involve subtle, human-imperceptible perturbations that lead to misclassifications or other incorrect behaviors. Most research has focused on pixel-level digital adversarial examples [13], assuming that the attacker has full control over the image's digital representation. In contrast, physical adversarial examples [20, 31] exploit variations in texture, shape, and lighting, processed through the system's sensor inputs. Examples include specific patterns applied to physical objects, such as stop signs, that cause misidentification by the autonomous vehicle vision system. Despite the potential risks, there is significantly less research on generating and defending against physical adversarial attacks compared to digital ones.

This paper primarily focuses on attacks against printed-image document authentication via source printer attribution, first introduced in [29], which is crucial for legal, governmental, and financial sectors that handle sensitive and confidential information. Ensuring document integrity is vital to prevent forgery and fraud, as these can have significant consequences. The Federal Trade Commission reported 2.6 million fraud cases, resulting in $10.3 billion in losses in 2023 [7] due to piracy. Ensuring the authenticity of printed documents is essential for protecting sensitive information and maintaining trust in official processes.

Within this framework, the first goal of this paper is to study the vulnerability of an image printer source attribution classifier based on DL against physical adversarial examples. The classifier is trained to identify a document's originating printer using a diverse set of documents from multiple printers. We aim to generate adversarial examples that remain effective after reprinting through the application of different attack algorithms. Traditionally, adversarial examples in the physical domain are created by adding perturbations directly to digital images, which are then transformed into a physical document or 3D object and fed to the AI model, successfully misleading the system. In our case, the attacked digital images are printed again by the same printer and scanned before being fed to the classifier. The Print and Scan (P&S) process applied to the attacked images poses several challenges to the creation of effective attacks. First, the P&S process degrades the attack's perturbation, requiring it to be stronger. Second, and most importantly, the features the attribution network relies on are reintroduced when the attacked digital image is printed for the second time, possibly nullifying the effectiveness of the attack.

Following previous work on the generation of physical adversarial examples, we use Expectation Over Transformation (EOT) [1] to craft perturbations that survive the distortion introduced when transition-ing from the physical to the digital domain. Our experiments confirm that EOT alone is not sufficient to maintain the effectiveness of adversarial examples after the P&S process, due to the reintroduction of printing artifacts on top of the adversarial attacked image. For this reason, we propose incorporating a generative AI P&S simulator within the EOT framework to generate adversarial attacks that preemptively account for the subsequent reprinting process. In particular, we use a Pix2Pix Generative Adversarial Network (GAN) [17] and a CycleGAN [42] to simulate the P&S transformation. We then integrate EOT with P&S into the Iterative Fast Gradient Sign Method (IFGSM) and the Carlini & Wagner (C&W) attacks, achieving a high Attack Success Rate (ASR) even after reprinting.

While adversarial examples were initially studied in the context of image classification, they have also been observed in Deep Neural Network (DNN) models applied to tasks such as object detection [36], intrusion detection [37], and voice recognition [39]. Among these, attacking object detectors in the real-world poses unique challenges since it requires the attack to be robust against significant changes in viewpoints and distance from the camera as the detector must localize and classify objects under varying spatial configurations. Unlike classifiers, which operate on globally cropped and aligned inputs, object detectors process the entire scene and are sensitive to object scale, orientation, occlusion, and background clutter. Additionally, the attacks have to be restricted to modifying the objects themselves, which correspond to limited regions of the images taken as input by the detectors.

To demonstrate the applicability of our approach in the context of an object detection task, we apply it to attack a License Plate Detector (LPD), which is a critical component in traffic enforcement, automated tolling, and surveillance. In particular, we focus on attacking an LPD model based on Single Shot MultiBox Detector (SSD) [23]. While we did not test the ability of the attack to work in a fully realistic scenario where the perturbation is mapped into a real license plate mounted on a car, we verified that the attacks maintain their effectiveness when the perturbation is strictly limited to the area occupied by the plate, and the attacked image is printed and photographically recaptured.[1]

Given the above, the main contributions of this work are:

1. We introduce two P&S simulators utilizing Pix2Pix GAN and CycleGAN image translation models.[2]

2. We integrate the P&S simulators as an additional transformation step in the EOT attack and use it to attack a source printer attribution classifier, in such a way that the attack withstands reprinting and recapturing process, successfully deceiving the target source printer attribution classifier.

3. We extend our framework to attack a license plate detector, showing that using the P&S simulator in the EOT step further improves the robustness of the attack in the presence of printing and recapturing.

This work is an extension of [29], where the P&S simulators were only used to attack a source printer attribution system. By extending the attack to an object detection system, like the license plate detector considered in this paper, we demonstrate the versatility of our approach, suggesting its general applicability to a wide range of applications.

The paper is organized as follows: Section 2 reviews adversarial attacks in digital and physical domains against image classifiers and object detectors. Section 3 details the development and performance of the P&S simulators. Section 4 focuses on the generation of robust adversarial examples for the source printer attribution task. Section 5 analyzes the experimental results regarding the printer attribution system. Section 6 describes the extension of the attack to the generation of robust adversarial examples targeting the LPD task. Section 7 summarizes our findings and suggests directions for future work.

2 Related Work

Adversarial examples are subtle input perturbations that mislead machine learning models while remaining imperceptible to humans. These perturbations often transfer across different model architectures, thereby exposing a

[1] Print and recapture can be seen as a simplified proxy for a full-fledged attack involving the creation of a real undetectable license plate.

[2] The effectiveness of the simulators has also been verified [28] for a completely different goal, namely, to enhance the robustness of synthetic image detectors against general post-processing operators.

key vulnerability of DNNs. They are typically crafted by adding small, norm-constrained modifications to correctly classified inputs. Common norms include L_0, L_2, and L_∞, which correspond to pixel count, Euclidean distance, and maximum change to any pixel, respectively.

Adversarial attacks are generally categorized as either digital or physical attacks. Digital attacks manipulate input data at the pixel level, assuming direct access to the digital input. In contrast, physical attacks involve altering the appearance of real-world objects, which are then captured by sensors or cameras. Physical attacks are significantly more challenging due to external factors such as lighting variations, different viewing angles, distances, and camera limitations, all of which can reduce the effectiveness of the adversarial perturbations.

2.1 Digital Domain Adversarial Attacks

Adversarial attacks in the digital domain have been extensively studied, forming the basis for more complex scenarios such as physical attacks. These attacks aim to subtly alter input data in a way that leads DNNs to make incorrect predictions, while keeping the perturbations visually imperceptible.

Digital adversarial examples are typically generated by solving a constrained optimization problem, where a small perturbation is added to a clean input to induce misclassification. The strength of the perturbation is often limited by an L_p norm (e.g., L_∞ [13], L_2, or L_0 [2]) to preserve the imperceptibility of the attack.

Based on the optimization strategy, digital white-box attacks can be broadly classified into two categories:

Gradient-based attacks compute the gradient of the loss function with respect to the input and apply small perturbations accordingly. A prominent example is the Fast Gradient Sign Method (FGSM) [13], which perturbs the input in a single step. While computationally efficient, FGSM often struggles to achieve high success rates. This limitation led to more powerful iterative methods such as Iterative FGSM (IFGSM) [20] and Projected Gradient Descent (PGD) [26], which apply repeated small updates to generate stronger adversarial examples.

Optimization-based attacks, such as the Carlini & Wagner (C&W) attack [2], treat adversarial example generation as an optimization problem. These methods explicitly balance the size of the perturbation and the objective of fooling the model, often achieving state-of-the-art effectiveness under various threat models.

2.2 Physical Domain Adversarial Attacks

While digital adversarial attacks assume full control over the input data, physical adversarial attacks target deep learning models by embedding perturbations onto real-world objects that are captured by cameras or sensors. These perturbations must remain effective despite environmental variations such as lighting, distance, viewpoint, and sensor noise. By extending traditional digital attacks into the physical world—through printed images, clothing, signboards, or other tangible surfaces—these attacks pose a significant threat, particularly in safety-critical domains such as document authentication, autonomous driving, and surveillance.

In the following, we review prior work on physical adversarial attacks, beginning with those targeting image classifiers, followed by attacks aimed at object detection models.

2.2.1 Physical Domain Attacks Against Image Classifiers

Physical domain attacks were first introduced by Kurakin *et al.* [20], who demonstrated that adversarial examples could survive the transition from digital-to-physical by printing perturbed images and re-capturing them with a smartphone. However, their experiments showed that attack success significantly dropped due to distortions introduced during the print-and-photograph process. To improve physical attack robustness, Sharif *et al.* [31] designed adversarial eyeglass frames to fool facial recognition systems. They incorporated a Non Printability Score (NPS) to ensure color reproducibility and a Total Variation (TV) loss to smooth perturbations. Similarly, Komkov and Petiushko [19] used TV loss to create adversarial stickers on hats targeting the ArcFace recognition model.

Lu *et al.* [25] further highlighted the fragility of physical attacks under varying viewing conditions, such as different angles and distances, emphasizing the need for robustness across transformations. To address this, Athalye *et al.* [1] proposed the Expectation Over Transformation (EOT) framework, which optimizes adversarial perturbations over a distribution of input transformations including scale, rotation, brightness, and noise. EOT enables the generation of robust, universal, and even targeted adversarial examples that remain effective under diverse physical conditions. Their work included 3D-printed objects and adversarial patches, which consistently fooled classifiers in scenarios such as traffic sign recognition [32]. Building on EOT, Evtimov *et al.* [5] proposed the Robust Physical Perturbation (RP_2) method, which combines synthetic and real-world transformations to generate adversarial examples on stop signs using posters or stickers. However, RP_2 requires photographing the printed image from various distances and angles, making the attack generation process resource-intensive.

To simulate real-world transformations more efficiently, Jan *et al.* [18] proposed digital-to-physical transformation (D2P), a pre-EOT transformation step using a conditional GAN [17, 42] to model the print-and-capture process. Although promising, their approach requires printing and photographing hundreds of samples to train the simulator, limiting its practicality. A work that is somewhat similar to the present work is [40]. Even there, the detector relies on the features that are reintroduced after rebroadcast hence requiring the design of a particular EOT strategy. However, the rebroadcasting artifacts are different from those introduced by P&S, hence the method proposed in [40] cannot be applied in our case.

As a matter of fact, all attacks based on EOT include natural geometric and color transformations to generate robust adversarial examples. As we will show later, however, this is not enough when the target system is a printer source attribution model. For this reason, we integrated the P&S simulators into the EOT framework. In this way, we were able to significantly improve the ASR, ensuring that the attack remains effective even after reprinting.

2.2.2 Physical Domain Attacks Against Object Detectors

In addition to adversarial examples designed for DNN models in image classification, several studies have extended adversarial attacks to object detection tasks. One of the earliest contributions in this direction was by Xie *et al.* [36], who proposed digital adversarial examples targeting object detection and semantic segmentation models. Following this, researchers began exploring physical adversarial attacks that involve modifying real-world objects to deceive detection models [3, 24, 33]. For instance, Lu *et al.* [24] attempted to deceive the YOLO object detector by printing adversarially modified traffic sign images, though the resulting success rate under real-world conditions was limited. Song *et al.* [33] introduced RP_2 algorithm, building on [6], which incorporated physical constraints - such as angle, distance, and lighting - into the attack generation process. Their results demonstrated that the YOLOv2 detector can be misled using adversarial stickers and printed posters when captured under specific conditions. Similarly, Chen *et al.* [3] proposed the ShapeShifter attack, which targets the Faster R-CNN detector by leveraging the EOT framework [1] to generate perturbations robust to various physical transformations.

More recently, Zhao *et al.* [41] categorized physical attacks into two distinct types: hiding attacks, which aim to suppress the detection of existing objects, and appearing attacks, which aim to fabricate detections by making nonexistent objects appear real to the detector. Huang *et al.* [15] introduced the Universal Physical Camouflage (UPC) attack, which generates a universal adversarial pattern capable of deceiving detectors across all instances of a specific object class (e.g., all cars in a scene).

While many physical attacks rely on altering the target object itself [24, 3, 33, 41, 15], another line of works explores adversarial strategies that do not require modifying the object directly. Huang *et al.* [16] proposed an adversarial signboard designed to resemble a benign advertisement. When placed strategically in the scene, this signboard misleads the Faster R-CNN detector, causing it to miss nearby stop signs. Lee and Kolter [21] developed a large, environment-placed adversarial patch capable of suppressing detections in YOLOv3 across a wide field of view. Li *et al.* [22] introduced a camera-level attack using a translucent sticker placed on the camera lens, which manipulates the input image stream and misleads downstream DNN classifiers, offering a novel threat model where the adversary compromises the sensor rather than the scene. Yang *et al.* [38] proposed a physical adversarial attack targeting license a plate detection model, particularly SSD, by manufacturing real-world metallic adversarial objects. Their work demonstrated attack transferability across multiple detectors and commercial platforms under varying physical conditions.

Despite advancements in physical adversarial attacks, many existing methods suffer from key limitations, often requiring overly strong perturbations or relying on large, intrusive artifacts such as signboards and patches (appearing attacks). In this work, we focus on an object detection model aimed at detecting license plates and apply our attack strategy to this domain. By integrating EOT with the P&S simulator based on image-to-image translation, we show that we can craft low-perceptibility perturbations that remain effective after digital-to-analog and analog-to-digital conversion via license plate printing and photographic recapture.

3 Print and Scan Simulation

Printing and scanning an image involves converting the digital image to a physical copy and back to the digital domain, introducing various distortions and artifacts. Printing can cause color shifts, ink diffusion, and minor geometric distortions due to the printer's mechanical characteristics and type of paper used. Scanning adds further distortions and noise depending on the scanner's resolution, color response, and mechanical misalignments. These steps affect pixel values and introduce artifacts specific to the printer and scanner, along with minor geometric alterations due to imperfect paper positioning within the scanner.

Given the time-consuming and costly nature of manually creating large volumes of printed and scanned images, we developed two P&S simulators[3] to

[3] The trained P&S simulators are publicly available at: https://github.com/NischayPurnekar/print-and-scan-simulator.

be directly included within the EOT process, enabling the vast generation of training images without the expense and effort of physical P&S. Research on simulating the P&S process by means of deep learning is sparse. A significant contribution in this domain comes from Ferrara *et al.* [8], who demonstrated that integrating a simulated P&S transformation during training improves the accuracy of face morphing attacks on printed and scanned face images. Their model estimates the pixel distortions incurred during printing and scanning, considering various critical parameters such as the responsivity of the acquisition device, the sampling function characterizing the digitization process of printed images, the point spread function of the printer and scanner, noise levels, and color transformations. However, the presence of device-dependent unknown parameters complicates real-world adaptations, as calculating the point spread functions of printers and scanners is challenging, and fine-tuning each parameter can be time-consuming, especially across multiple devices. Mitkovski *et al.* [27] also utilized a Pix2Pix GAN to emulate the P&S process for biometric applications.

To start with, and similarly to [27], we trained a Pix2Pix GAN [17] simulator. Training the Pix2Pix GAN, however, requires pixel-wise alignment of digital and P&S images for effective computation of the mean square error loss. To address this problem, we employed image alignment techniques during training. We also trained a CycleGAN P&S simulator, which supports unpaired image-to-image translation. In fact, CycleGAN does not necessitate paired images, thus greatly simplifying the preparation of the dataset.

3.1 Architecture of the Simulators

Pix2Pix and CycleGAN have been extensively used to address various generative tasks. In our case, the objective of the Pix2Pix generator is to translate the input images from the digital to the P&S domain, while the discriminator is asked to distinguish between real P&S and digital pairs and their synthetic counterparts. The CycleGAN generators aim to translate images from the digital to the P&S domain and vice versa, ensuring cyclic consistency. With respect to classical CycleGAN training, we did not use the identity loss. In fact, printing a printed and scanned image again should not result in the identity operator, as the second P&S process would further degrade the image quality.

3.1.1 Pix2Pix

The Pix2Pix architecture is tailored for paired image-to-image translation tasks. It consists of a generator and a discriminator, as illustrated in Figure 1. In our case, the objective of the generator is to translate the input images from the digital domain to the P&S domain, while the discriminator is

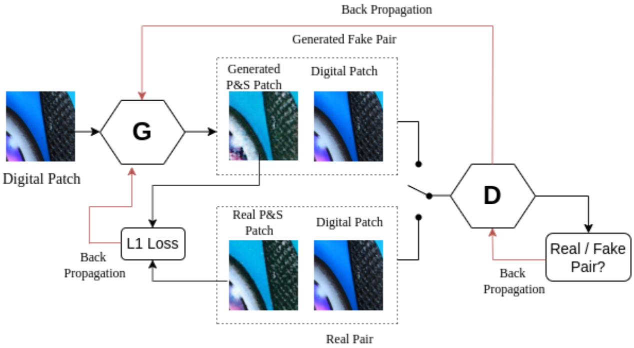

Figure 1: Overview of the Pix2Pix GAN Architecture including generator and discriminator.

asked to distinguish between real P&S and digital pairs and their synthetic counterpart.

The Pix2Pix network is trained by relying on two main losses: the adversarial loss and the L1 loss. The adversarial loss $\mathcal{L}_{\text{adv}}$ ensures that the generated images are realistic enough to deceive the discriminator, and is defined as:

$$\mathcal{L}_{adv}(G, D) = \mathbb{E}_{x,y}[\log D(x, y)] + \mathbb{E}_x[\log(1 - D(x, G(x)))] \tag{1}$$

where G and D indicate, respectively, the functions implemented by the generator and the discriminator. The L1 loss enforces pixel-level similarity between the generated and actual P&S images, and is defined as:

$$\mathcal{L}_{L1}(G) = \mathbb{E}_{x,y}[\|y - G(x)\|_1] \tag{2}$$

The generator is trained to minimize a linear combination of the above two losses, while the discriminator is trained to maximize $\mathcal{L}_{\text{adv}}$:

$$G^* = \arg\min_G \left\{ \left[\max_D \mathcal{L}_{\text{adv}}(G, D) \right] + \lambda \mathcal{L}_{L1}(G) \right\} \tag{3}$$

In our implementation we used a U-Net architecture for the generator, featuring an encoder-decoder structure with skip connections. The encoder captures high-level features through convolutional and pooling layers, while the decoder reconstructs images using upsampling layers, preserving fine details with skip connections. The input size of the network is 256×256×3. Each U-Net skip connection block includes a convolutional layer, batch normalization, and leaky ReLU activation. For the discriminator, we used three convolutional layers with batch normalization and leaky ReLU activation. Both networks were trained with Adam optimizer, ensuring stable convergence.

3.1.2 CycleGAN

A CycleGAN comprises of two generators and two discriminators, with the generators aiming to translate images from the digital to the P&S domain and vice versa. The key concept behind CycleGAN is to ensure cyclic consistency, meaning that when the output of the first generator is used as input for the second generator, the resulting image closely resembles the original input. Figure 2 illustrates the general architecture of CycleGAN and the paths followed to compute the two main losses used for training: the adversarial loss and the cycle consistency loss. The adversarial loss ensures that the generated P&S images resemble real P&S images. CycleGAN uses two adversarial losses, one for each direction of translation:

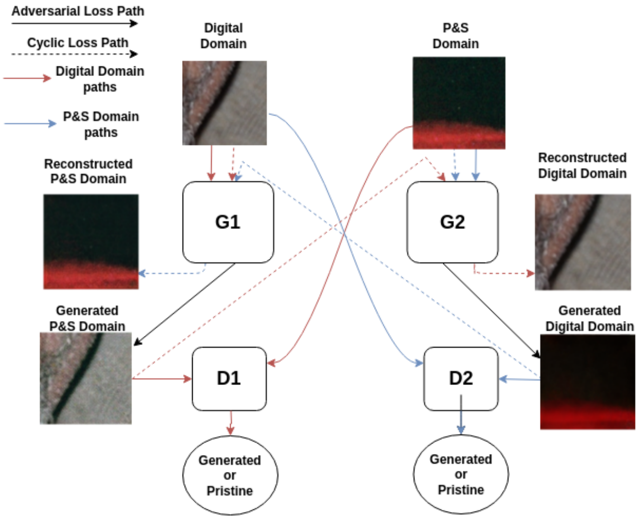

Figure 2: CycleGAN architecture overview, featuring two generators and two discriminators. The figure illustrates the pathways between the digital domain and the print-and-scan (P&S) domain, highlighting the processes for adversarial and cycle consistency losses.

$$\begin{aligned}\mathcal{L}_{adv}(G_1, D_2, X, Y) = {} & \mathbb{E}_{y\sim p_{data}(y)}[\log D_2(y)] \\ & + \mathbb{E}_{x\sim p_{data}(x)}[\log(1 - D_2(G_1(x)))]\end{aligned} \tag{4}$$

$$\begin{aligned}\mathcal{L}_{adv}(G_2, D_1, Y, X) = {} & \mathbb{E}_{x\sim p_{data}(x)}[\log D_1(x)] \\ & + \mathbb{E}_{y\sim p_{data}(y)}[\log(1 - D_1(G_2(y)))]\end{aligned} \tag{5}$$

G_1 and G_2 are the generators for digital to P&S and P&S to digital translations, respectively. D_1 and D_2 are the discriminators that distinguish between real and generated images in each domain. The cycle consistency loss ensures that an image can be translated to the other domain and back without significant changes. This involves two losses: one for translating a digital image to P&S and back, and another for translating a P&S image to digital and back.

$$\begin{aligned}\mathcal{L}_{cycle}(G_1, G_2) = \mathbb{E}_{x\sim p_{data}(x)}[\|G_2(G_1(x)) - x\|_1] \\ + \mathbb{E}_{y\sim p_{data}(y)}[\|G_1(G_2(y)) - y\|_1]\end{aligned} \tag{6}$$

With respect to the classical CycleGAN architecture, we did not use the identity loss. In fact, printing a printed and scanned image again should not result in the identity operator, as the second P&S process would further degrade the image quality.

$$\begin{aligned}\mathcal{L}(G_1, G_2, D_1, D_2) = \mathcal{L}_{adv}(G_1, D_2, X, Y)+ \\ \mathcal{L}_{adv}(G_2, D_1, Y, X) + \lambda\mathcal{L}_{cycle}(G_1, G_2)\end{aligned} \tag{7}$$

where λ controls the relative importance of the two objectives. We aim to solve:

$$G_1^*, G_2^* = \arg\min_{G_1,G_2} \big[\max_{D_1,D_2} \mathcal{L}(G_1, G_2, D_1, D_2)\big] \tag{8}$$

$$\mathcal{L}_{D_1} = \mathcal{L}_{adv}(G_2, D_1, Y, X) \tag{9}$$

$$\mathcal{L}_{D_2} = \mathcal{L}_{adv}(G_1, D_1, X, Y) \tag{10}$$

Training the CycleGAN involves balancing these losses, with the generators minimizing an aggregate loss that is the sum of the adversarial loss and the cycle consistency loss, and the discriminators distinguishing between real and generated images within their respective domains. When training, we noticed that without identity loss, after around 200 epochs the outputs began to exhibit slight color shifts, sporadic speckling, and an increasing cycleconsistency error. To address this, we employed a 50image replay buffer, applied a linear learningrate decay starting at epoch 150 over 600 epochs, used instance normalization with LeakyReLU activations, and implemented early stopping based on FID and SSIM evaluations on a printedandscanned validation set. After training, we obtain two generators: one simulating the P&S process and one attempting to recover the original digital image from its P&S version. In this work, we only use the first generatorthe one simulating the P&S process. We utilized the same architectures used for the Pix2Pix simulator. Specifically, the generators employed a U-Net architecture with input dimensions of 256×256×3, while the discriminators consisted of three convolutional layers, incorporating batch normalization and leaky ReLU activation. Both networks were trained using the Adam optimizer.

3.2 Dataset

To train the simulators, we used a dataset derived from the second version of the VIPPrint dataset [10]. This dataset employed in prior work such as [11], consists of human face images printed with various modern color laser printers, each operating at different resolutions. Acquisition was performed using a TaskAlfa 3551 multi-functional scanner at 600×600 dpi resolution, and the images were saved using lossless compression. The size of the digital images is 1024×1024×3, while the P&S images are approximately 2036×2038×3, with slight variations of 5 to 10 pixels in both dimensions introduced during scanning. To align the resolutions of digital and P&S images, the digital images were upsampled to match the P&S image resolution. Our experiments focused on a subset of P&S images printed by one of the 12 printers in the VIPPrint dataset, specifically a Kyocera P5021 CDN Color Laser printer. We used a subset of 200 printed and scanned images from this printer for our experiments. To match the input size of the Pix2Pix and CycleGAN networks, we trained the networks on image patches extracted from 100 printed and scanned images along with their corresponding digital images. The patches were 256×256×3 in size and were extracted without pixel overlap. For Pix2Pix, we aligned the digital and printed and scanned patches using [4]. If alignment was challenging or significant pixel differences were detected, the corresponding patch was skipped. This approach yielded 4,678 aligned digital and P&S patches. In contrast, CycleGAN training utilized unaligned digital and P&S patches, leveraging the ability of CycleGANs to handle unpaired image data. In total, 4,914 digital and P&S patches were used to train the CycleGAN simulator.

3.3 Training

The Pix2Pix GAN simulator was trained for 800 epochs using the Adam optimizer with parameters $\beta_1 = 0.5$, $\beta_2 = 0.999$, and a learning rate of 1×10^{-4}. The network utilized 64 filters and a Leaky ReLU activation function with a slope of 0.2, while the batch size was restricted to 1. For training CycleGAN, we used the same hyperparameters as the Pix2Pix GAN simulator over 600 epochs. Both the GAN adversarial loss and cyclic consistency loss weights were set to 10. After training, we assessed the performance of both simulators by inputting original digital patches. To introduce variability, we added Gaussian noise with zero mean and variance of 0.0625 to the digital images before feeding them to the simulator. This ensured that multiple inputs of the same digital image yielded slightly different simulated outputs, mimicking real-world variations when an image undergoes printing and scanning multiple times.

We evaluated the quality of the simulated images both visually (Figure 3) and quantitatively using metrics such as the Structural Similarity Index

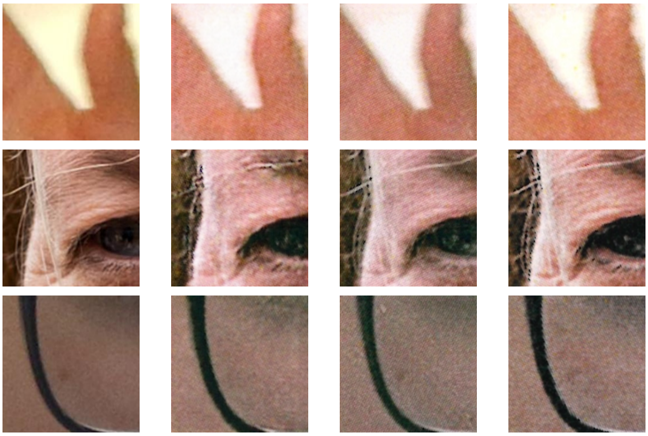

Figure 3: Examples of digital and simulated P&S patches with corresponding ground truth. The first column shows digital inputs; the second and third show outputs from Pix2Pix and CycleGAN P&S simulators; the last column shows real printed and scanned patches which are the ground truth patches.

Table 1: Similarity metrics between simulated and real P&S images. Higher SSIM and lower FID indicate better simulation quality.

P&S Simulator	**SSIM Score** (↑)	**FID Score** (↓)
Pix2Pix GAN	0.84	47
CycleGAN	0.87	45

(SSIM) and Fréchet Inception Distance (FID) (Table 1). The SSIM scores are 0.84 for Pix2Pix GAN and 0.87 for CycleGAN, while the FID scores are 47 for Pix2Pix GAN and 45 for CycleGAN. As shown in Figure 3, the images generated by the P&S simulators closely resemble the corresponding ground-truth images, demonstrating their effective learning of the distortions inherent in the P&S process.

4 Physical Domain Adversarial Examples Against Printer Source Attribution

In this section, we detail our physical domain attack against printer source attribution. First we present the threat model to frame the attack. Then, we outline the targeted printer source attribution model and the training dataset we utilized. Finally, we describe the attack methodology.

4.1 Threat Model — White-Box Evasion Attack

We consider an attack aiming at altering an image printed by a specific printer, P, in such a way that the printer source attribution model can no longer identify P as the source printer (untargeted attack) after the image is reprinted by P. The challenge is to ensure the attack's effectiveness even after the attacked image undergoes reprinting and scanning. The attacker has white-box access to the source attribution model, including its weights and architecture. This allows the attacker to optimize and evaluate the adversarial examples in the digital domain before executing the physical-world attack by printing, scanning, and strategically placing the attacked images.

4.2 Targeted Model

The printer source attribution system targeted by our attack is the one described in [9]. This system, trained on the VIPPrint dataset Ferreira *et al.* [10], analyzes the 10 highest-energy 224×224×3 patches of the image and uses a majority voting decision for classification. Preliminary experiments in [9] showed that a basic reprinting black-box attack can deceive the original classifier. To enhance resilience against such attacks, the authors fine-tuned the model using a dataset of reprinted images, resulting in a more robust (hardened) source attribution model, which is the focus of our attack. Since the classifier operates on patches, the adversarial attacks are applied to 224×224×3 image patches. However, because the attack may slightly alter the energy of the patches, the classifier could potentially analyze different patches after the attack. Therefore, we decided to attack all the patches in the image. This approach also prevents the introduction of visible discontinuities at patch borders. The success of the attack hinges on inducing sufficient patch misclassifications to misclassify the true printer as the most voted option. Our target is specifically the Kyocera-ecosys P5021cdn laser printer, identified as class #12 in the attribution system's multiclass classification.

4.3 Attack Pipeline

The attack pipeline (Figure 4) begins with printing and scanning a digital image, followed by the application of an adversarial attack in the digital domain. To maintain the effectiveness of the adversarial perturbation after printing and scanning, we use EOT in combination with a P&S simulation implemented using generative AI models. The adversarial digital image is then physically printed using the same printer. Finally, the printed image is scanned and analyzed by the source attribution model, which is a predictive AI system designed to identify the origin of the document.

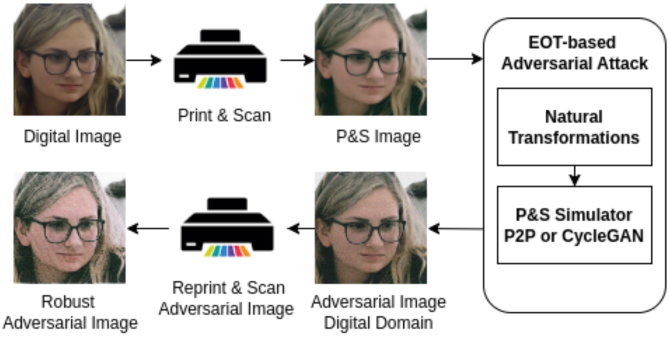
Digital Image
Print & Scan
P&S Image
EOT-based Adversarial Attack
Natural Transformations
P&S Simulator P2P or CycleGAN
Adversarial Image Digital Domain
Reprint & Scan Adversarial Image
Robust Adversarial Image

Figure 4: Attack pipeline for the generation of robust adversarial examples.

4.3.1 Digital Domain Attack

Initially, we assessed the effectiveness of digital domain attacks (without EOT) in inducing misclassifications when the attacked image is subsequently printed and scanned. Adversarial examples were generated using a non-targeted version of I-FGSM [20] and C&W attack [2]. For I-FGSM, we set $\varepsilon = 0.03$, with a step size of 0.01 over 100 iterations. Similarly, for the C&W attack, we let $\varepsilon = 0.1$, with a binary search step size of 9 and a learning rate of 0.01 across 1000 iterations. These hyperparameters were selected to ensure effective attack coverage across most of the patches in the P&S image.

4.3.2 Physical Domain Attack

To generate robust adversarial examples in the physical world, we integrated the I-FGSM and C&W attacks into an EOT framework, effectively addressing the domain shifts between digital and physical domains. EOT involves defining a pool of transformations T to simulate these shifts. The transformations used in our EOT attack are detailed in Table 2, including their parameters, essential for replicating practical domain shifts. Additionally, we incorporated the Pix2Pix and CycleGAN P&S simulators within the transformation set. Results were averaged over 10 transformed samples to assess attack effectiveness. Through extensive experiments, we identified an optimal combination of transformations T (Table 2) that consistently produce successful adversarial examples. Our setup includes rotation (2.0 to 10.0 degrees), zoom blur (factors between 1.05 and 1.10), and pixel shifts (5 pixels in all directions) with an inclusion probability of 100%. For color transformations, brightness deltas (10 to 40) and a fixed contrast factor of 0.3 are applied with 50% probability. Additionally, either CycleGAN or Pix2Pix GAN simulators are chosen with a probability of 50% to simulate P&S effects.

Table 2: Set of transformations used in the EOT attack. Each transformation is applied with the corresponding probability during optimization.

Transformation	**Parameter**	**Value Range**	**Probability**
Brightness	Brightness Delta	[10, 40]	50%
Contrast	Contrast Factor	0.3	50%
Rotation	Rotation Angle	[2°, 10°]	100%
Zoom	Zoom Range	[1.05, 1.10]	100%
Pixel Shift	Pixel Offset (x/y)	5 pixels	100%
Pix2Pix P&S Simulator	GAN Model	Trained on P&S pairs	50%
CycleGAN P&S Simulator	GAN Model	Trained on P&S pairs	50%

The attack algorithms within the EOT framework were configured with the following hyperparameters: for I-FGSM, $\varepsilon = 0.15$, a step size of 0.03, and 500 iterations were used; for C&W, we employed $\varepsilon = 0.15$, 9 binary search steps, a learning rate of 0.01, and 1000 iterations. When we incorporated the P&S simulators into EOT, I-FGSM utilized $\varepsilon = 0.4$, a step size of 0.07, and 500 iterations, while for C&W we used $\varepsilon = 0.4$, 9 binary search steps, a learning rate of 0.01, and 1000 iterations. For the physical domain conversion (i.e., reprinting), we used a Kyocera Ecosys P5021cdn color laser printer at 1200 dpi to print the digitally attacked images on A4-sized glossy paper. The printed images were then scanned using a Kyocera TaskAlfa 3551ci flatbed scanner at 600 dpi optical resolution, saved as high-sharpness, lossless images in .jpg format.

5 Experimental Results on Printer Source Attribution

To demonstrate the robustness of the hardened source attribution classifier against adversarial examples, we conducted experiments in both the digital and physical domains. Our study involved various white-box attacks, including I-FGSM and C&W, both with and without EOT, and incorporating P&S simulators within the EOT transformations. These experiments were performed on a test set of 20 documents, with each document split into 81 patches, totaling 1,620 patches. The hardened classifier achieved perfect documentlevel accuracy of 100% and correctly classified 1,415 out of 1,620 patches, corresponding to an 87.3% patchlevel accuracy. To measure the strength of the perturbation introduced by the attacks, we computed the Peak Signal-to-Noise Ratio (PSNR) between the original and attacked patches, both before and after the reprinting and scanning process. PSNR calculations were focused only on successfully attacked patches. Additionally, we evaluated the ASR across all patches that were correctly classified before the attack in the images and on the top 10 highest energy patches, which are generally more challenging to attack. After reprinting, the final classification is determined through majority voting on the results obtained from the top 10 highest energy patches of each document. To assess the overall robustness of the system, we also computed the ASR after the majority voting, where the printer with the largest number of votes among the top energy patches is selected. In all cases, the ASR was computed only on patches that were correctly classified prior to the attack, both for the full set of patches and for the top 10 highest energy patches.

The results of our experiments are reported in Table 3. Analyzing the second column of the table, we observe that all attacks are highly effective when they are applied in the digital domain, achieving nearly 100% ASR. As expected, the attacks incorporating EOT, particularly those utilizing P&S simulation, exhibit lower PSNR values. The fourth column of the table reports the effectiveness of the attacks in the physical domain, considering the ASR on *all*[4] patches after reprinting. For standard I-FGSM and C&W, the ASR decreases dramatically, while the application of EOT with natural transformations limits the ASR drop. Including the P&S simulators in the EOT transformations further improves the ASR to 79.92% for C&W and 85.79% for I-FGSM, which is a significant advantage with respect to EOT with natural transformations. The main advantage of including P&S simulation within EOT becomes apparent when we limit the analysis to the 10-highest energy blocks of each image. In this scenario, the ASR with natural EOT is only 33.5% for I-FGSM and 28% for C&W, while EOT with P&S simulation allows to attack 69% and 56.5% of the patches. In the last column of the table, we report the ASR after majority voting on the 10-highest energy blocks, measuring the final performance against the printer source attribution system (see Section 4.2). We observe that the ASR after majority voting drops to negligible values for standard I-FGSM and C&W attacks, showing only slight improvement with EOT using natural transformations.[5] However, when the P&S simulator is incorporated to EOT, the ASR significantly increases for both I-FGSM and C&W attacks. Specifically, the ASR for I-FGSM rises from 25% to 70%, and from 20% to 65% for C&W. These results highlight the effectiveness of incorporating the P&S simulator, given the complexity of creating adversarial examples that survive the reprinting process. Our experiments also suggest that patches with dark backgrounds tend to reintroduce stronger artifacts upon reprinting, thus requiring an excessive distortion.

In Figure 5, we present adversarial examples after reprinting, generated using various attacks. The images include the initial P&S image (the attack target), adversarial examples produced by standard attacks, EOT attacks with natural transformations, and EOT attacks incorporating P&S simula-

[4] *All* refers to patches that were correctly classified as ground truth before the attack.

[5] These results indirectly support the choice made in Ferreira and Barni [9] to base the classification only on the highest energy patches.

Table 3: Effectiveness of various attacks in both the digital and physical domain. ASRs are averaged across all patches correctly classified before the attack, on the top 10 highest energy patches of each image, and after majority voting on the top 10 patches.

Attack Method	**ASR Digital**(%)	**PSNR (dB)**	**ASR Printed All Patches**(%)	**ASR Printed Top 10 Patches**(%)	**PSNR (dB)**	**ASR Printed Majority Voting**(%)
IFGSM	100%	36.14	27.20%	15.5%	28.89	10%
IFGSM (EOT)	96.39%	20.08	75.33%	33.5%	17.25	25%
IFGSM (EOT+P&S)	100%	13.12	85.79%	69.0%	11.89	70%
CW	100%	33.86	22.82%	14.0%	25.53	10%
CW (EOT)	96.67%	19.52	64.94%	28.0%	16.96	20%
CW (EOT+P&S)	100%	12.19	79.92%	56.5%	11.18	65%

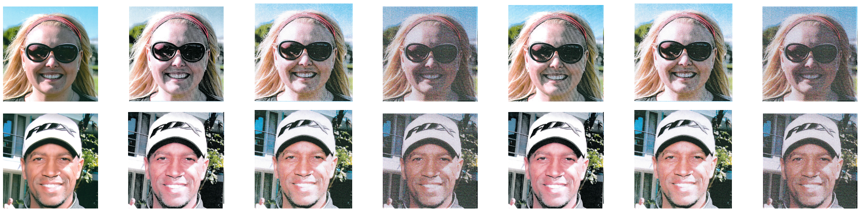

Figure 5: Examples of attacked images after reprinting and scanning. Each row begins with (a) the original P&S image, followed by adversarial examples generated using: (b) I-FGSM, (c) I-FGSM with EOT, (d) I-FGSM with EOT+P&S, (e) C&W, (f) C&W with EOT, and (g) C&W with EOT+P&S.

tions. Comparing the initial P&S images to the reprinted adversarial examples generated by standard I-FGSM or C&W attacks we see that reprinting weakens the perturbation. The examples produced by I-FGSM(EOT+PS) and CW(EOT+PS) demonstrate the importance of the P&S simulation in creating robust adversarial examples that withstand reprinting and scanning.

6 Application to License Plate Detection

In this section, we extend our robust attack method to generate physical domain adversarial examples aimed at fooling a license plate detector. We begin by describing the LPD model, and provide details on the target detector as well as the attack pipeline used to craft adversarial examples in both digital and physical domains. Finally, we evaluate the effectiveness of the attacks.

6.1 Threat Model — Evasion Attack Against Bounding Box Detection

As mentioned in the introduction, we focus on an SSD-based license plate detectors, capable of localizing the license plate area. We focused on bounding box detection on the finding in [30] that an AI trained on simple geometric objects shows in generative AI artifacts which can be measured and analyzed in more detail based on the well-defined structure. Disturbing the bounding box detection is therefore the basic attack task. The model outputs both class scores and bounding box coordinates, serving as the adversarial target in both digital and physical attack settings.

We consider an attack aiming to modify the plate in such a way that the bounding box of the plate can no longer be detected. The challenge is to ensure attack effectiveness when the license plate is printed and the scene is recaptured with a mobile camera. We adopt a white-box threat model, assuming full access to the detectors parameters and gradients, which allows us to leverage gradient-based optimization techniques. The objective is to suppress license plate detection by introducing perturbations only in the license plate area to reduce the confidence score of the license plate class.

6.2 Target Detector

We adopt the SSD300 [23] architecture for LPD due to its efficiency and real-time performance. While several physical attacks have been proposed against object detectors, most of them target specific architectures such as Faster R-CNN and YOLO [3, 33], whereas physical attacks on SSD [23] models remain largely underexplored. The model accepts input images of size 300 × 300 × 3 and performs object localization and classification in a single forward pass. It was trained from scratch on a custom dataset composed of 24,079 training, 4,013 validation, and 4,014 test images, aggregated from multiple publicly available sources. The dataset is highly diverse, covering a wide range of license plate formats, backgrounds, lighting conditions, and geographic regions.

Unlike existing pre-trained LPD models, which are often constrained by region-specific characteristics or inconsistent annotation standards, training from scratch allowed full control over the dataset composition, resolution, and augmentation, which is critical for evaluating robustness in our physical attack pipeline.

The model was trained using a batch size of 16, learning rate of 1×10^{-3}, momentum of 0.9, and weight decay of 5×10^{-4}. The learning rate was decayed at 80,000 and 100,000 iterations by a factor of 0.1 using a StepLR scheduler. Early stopping with a patience of 15 epochs was applied. Standard data augmentations (random cropping, flipping, color jittering) and gradient clipping (threshold 5.0) were used to improve training stability and generalization.

The LPD model is optimized using the SSD MultiBox loss:

$$L(x, c, l, g) = \frac{1}{N}\left(L_{\text{conf}} + \alpha L_{\text{loc}}\right), \tag{11}$$

where L_{conf} is the confidence loss and L_{loc} is the localization loss. The confidence loss is computed as:

$$L_{\text{conf}} = -\sum_{i \in \text{Pos}} \log(\hat{c}_i^p) - \sum_{i \in \text{Neg}} \log(\hat{c}_i^0), \tag{12}$$

and the localization loss is given by:

$$L_{\text{loc}} = \sum_{i \in \text{Pos}} \sum_{m \in \{cx, cy, w, h\}} \text{smooth}_{L1}(l_i^m - \hat{g}_i^m), \tag{13}$$

where $\hat{c}_i^p$ and $\hat{c}_i^0$ are the predicted softmax scores for the positive (license plate) and negative (background) classes respectively, and l_i^m, $\hat{g}_i^m$ are the predicted and ground-truth bounding box parameters for matched prior i.

The trained model achieves a mean Average Precision (mAP) of 0.98 on the test set with an accuracy of 100%, establishing a strong baseline for subsequent adversarial evaluation. Figure 6 shows examples of precise bounding box detections produced by the model under clean (non-adversarial) conditions.

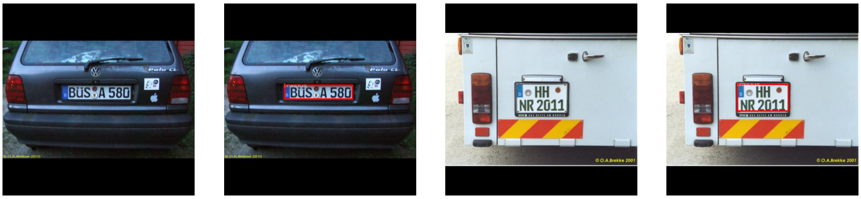

Figure 6: License plate detection results using the LPD model. The red bounding boxes indicate the localized license plate regions. The first and third images (from the left) show the input images, while the second and fourth images display the corresponding detection outputs.

6.3 Attack Pipeline

The attack pipeline is illustrated in Figure 7. A clean test image of a car containing a license plate is first captured by an image sensor and then passed to the SSD300 LPD model. The model resizes the image to 300 × 300 pixels and performs a forward pass to predict potential bounding boxes. The highest-confidence detection corresponding to the license plate class is selected, and a binary mask is generated to confine the perturbation only to this region. This mask is kept constant throughout the attack to maintain spatial consistency and visual realism. The image is then attacked iteratively using the sign of the gradients, with the perturbations restricted to the masked area. To facilitate physical-world applicability, the final adversarial image is resized back to its original resolution. The image with the adversarial license plate is then printed. The printed image is recaptured with a smartphone.

To facilitate physical-world applicability, the final adversarial image is resized back to its original resolution. The image with the adversarial license plate is then printed. To simulate real-world conditions, the printed image

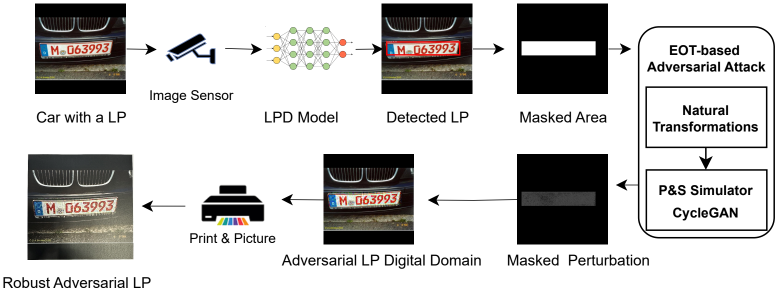

Figure 7: Attack pipeline for generating robust adversarial examples in license plate detection (LPD). The process includes input preprocessing, mask-based perturbation, iterative optimization, and resizing for physical-world applicability.

is recaptured using a smartphone.[6] The printer used is a Kyocera Ecosys P5021cdn. The capture device is an iPhone 16 Pro Max with autofocus enabled and HDR/auto-enhancement disabled. The printed plate is laid flat on a table and photographed handheld with the camera axis approximately perpendicular to the plate, under standard indoor lighting. The resulting image is then bicubically resized to 640Œ640 px.

In this setting, we limit our evaluation to the I-FGSM attack, since we found I-FGSM to be sufficiently effective for license plate detection, offering a good balance between performance and computational efficiency. We used a CycleGAN-based P&S simulator to build the physical domain attack, as preliminary tests showed that it produces more stable and visually consistent transformations on license plate images, making it a practical choice for this experiment.

6.3.1 Digital Domain Attack

In the digital adversarial attack scenario (without EOT), we perform a non-targeted I-FGSM attack on the SSD300 license plate detector to induce misclassification. In our experiments, the attack is executed over 30 iterations with a step size of $\alpha = 0.0314$ and a maximum perturbation limit of $\epsilon = 0.0627$. Since the goal of the attack is to induce the detector to classify the license plate region as background, the attack optimization focuses solely on the reduction of the classification confidence loss, and does not consider the localization loss. The hyperparameters were chosen to ensure the perturbation effectively covers the license plate bounding box masked region.

[6] Our scheme is a simplification of the real-word scenario. In a setup closer to real-word, only the adversarial license plate should be printed and mounted on cars, then the scene is reacquired.

6.3.2 Physical Domain Attack

We extended the I-FGSM attack using the EOT framework to generate robust physical adversarial examples, addressing the gap between digital and physical domains. Our experiments explored two EOT variants: one using natural transformations and another incorporating a CycleGAN-based P&S simulator.

For each attack iteration, 15 transformed versions of the adversarial image were sampled, using a combination of geometric and photometric augmentations. These included affine transformations (rotation up to 10°, 10% translation, scale range 0.61.5, shear=5), perspective distortions (distortion scale up to 0.1), color jitter (brightness=0.3, contrast=0.5, saturation=0.3, hue=0.1), random grayscale conversion (30% probability), Gaussian blur (kernel size=3, σ in [0.3, 2.0]), and motion blur (kernel size=5, σ in [1.5, 4.0]). The complete list of transformation parameters is provided in Table 4.

Table 4: Set of transformations used in the EOT attack with and without the P&S simulator for LPD. Each transformation is applied per iteration during optimization.

Transformation Type	**Parameter(s)**	**Value Range**	**Probability**
Affine Transform	Degrees, Translate, Scale, Shear	10°, 10%, [0.6, 1.5], 5	Randomly Sampled
Perspective Distortion	Distortion Scale	[0.0, 0.1]	80%
Color Jitter	Brightness, Contrast, Saturation, Hue	0.3, 0.5, 0.3, 0.1	100%
Random Grayscale	Grayscale Application Probability	–	30%
Gaussian Blur	Kernel Size, Sigma	3, [0.3, 2.0]	100%
Motion Blur	Kernel Size, Sigma	5, [1.5, 4.0]	Randomly Sampled
CycleGAN P&S Simulator	GAN Model	Trained on P&S Pairs	100%

In both variants, perturbations were spatially restricted using a fixed binary mask that covered the license plate region, based on the initial detection output. The standard EOT attack was run for 40 iterations using a step size of $\alpha = 0.0314$ and a maximum perturbation magnitude $\epsilon = 0.0627$. The EOT+P&S version, which requires stronger and more resilient perturbations, was executed for 60 iterations with a step size of $\alpha = 0.0941$ and perturbation budget $\epsilon = 0.1882$. In both settings, only the classification (confidence) loss was optimized, while localization loss was excluded. The integration of the P&S simulator significantly improved physical transferability, as later confirmed through robustness evaluations involving motion blur, grayscale conversion, and partial occlusion.

6.4 Experimental Results

We evaluated the robustness of the SSD300-based LPD against adversarial examples in both digital and physical domains. The attacks were performed using the I-FGSM, considering three configurations: baseline I-FGSM, I-FGSM with EOT, and I-FGSM with EOT augmented by a CycleGAN P&S simulator. Each method was tested on a set of 20 license plate images from the test dataset with an image resolution of 640 × 640 × 3.

In this case, the ASR corresponds to the percentage of images where the license plate was either not detected or incorrectly classified as background. To assess the perceptual quality and intensity of the perturbations confined only to the license plate area, we measured the PSNR for successfully attacked images.

6.4.1 Digital Domain

In the digital domain, the application of the I-FGSM adversarial attack resulted in successful evasion of the LPD across all 20 test images, achieving an ASR of 100% and a PSNR of 24.46 dB, as reported in Table 5. Visual examples shown in Figure 8 demonstrate that the perturbations applied to the license plate region are nearly imperceptible to the human eye. While this attack proves effective in the digital setting, it fails to maintain its effectiveness in the physical domain, as demonstrated in the following section.

Table 5: Effectiveness of the I-FGSM attack in the digital domain, reported as average ASR and PSNR over 20 test images. PSNR is computed within the license plate region using a masking approach.

Attack Method	**ASR (%)**	**PSNR (dB)**
I-FGSM	100%	24.46

6.4.2 Physical Domain

According to our pipeline, the conversion of a digital adversarial examples generated by EOT-based adversarial attack to the physical domain involves printing the adversarial image and capturing the printed output using a smartphone. During recapture we ensured that the captured image maintains a square aspect ratio, and it is resized to original image resolution of 640×640×3 for compatibility with the detection pipeline.

We evaluated the robustness of adversarial examples generated using EOT, both with and without the P&S simulator. As shown in Table 6, adversarial examples crafted using EOT alone achieved a digital-domain ASR of 95% (19 out of 20 instances were successfully fooled by the attack.) with a PSNR of 23.71 dB, but their effectiveness significantly dropped in the physical domain, reaching a printed-domain ASR of only 57.89%. The ASR in the physical domain is computed exclusively on the images that were successfully attacked in the digital domain (19 in our case). In contrast, augmenting the transformation pool with a CycleGAN-based P&S simulator during the attack generation

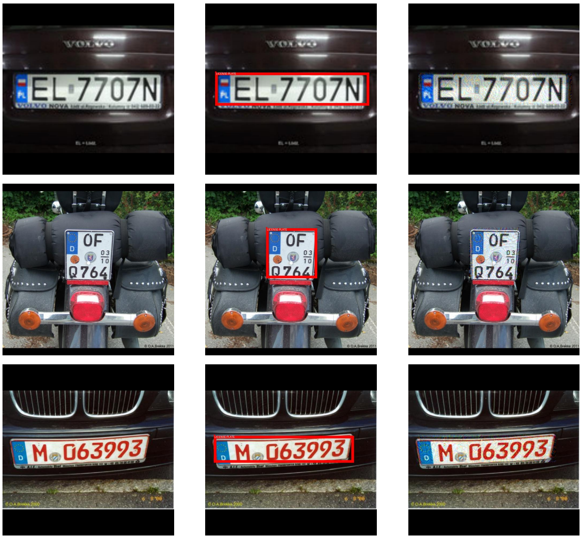

Figure 8: Each row illustrates, from left to right: the original (clean) license plate image, the corresponding detected license plate region, and the adversarial example generated using the I-FGSM attack, restricted to the license plate area.

Table 6: Attack effectiveness in both digital and physical domains. ASRs in the digital domain are averaged over 20 test images, while physical domain ASRs are averaged only on the set of images successfully attacked in the digital domain. PSNR is calculated exclusively within the license plate region, where perturbations are applied using a masking strategy.

	ASR (%)		**ASR (%)**
Attack Method	**Digital Domain**	**PSNR (dB)**	**Printed Domain**
I-FGSM (EOT)	95%	23.71	57.89%
I-FGSM (EOT + P&S)	95%	22.24	73.68%

led to a substantial improvement in physical robustness. The EOT+P&S variant achieved the same 95% ASR in the digital domain with a slightly lower PSNR of 22.24 dB, but yielded a much higher ASR of 73.68% after reprinting and recapture.

This improvement highlights the importance of modeling domain-specific degradations introduced by the P&S process. The visual difference between the two attacks can be observed in Figure 9. The images generated with the P&S-aware attack inhibit license plate detection more effectively after undergoing physical transformations.

To summarize, the printerattribution scenario proves to be more challenging than the LPD task, which currently represents a more conventional adversarial attack setting. The increased difficulty is due to the reintroduction

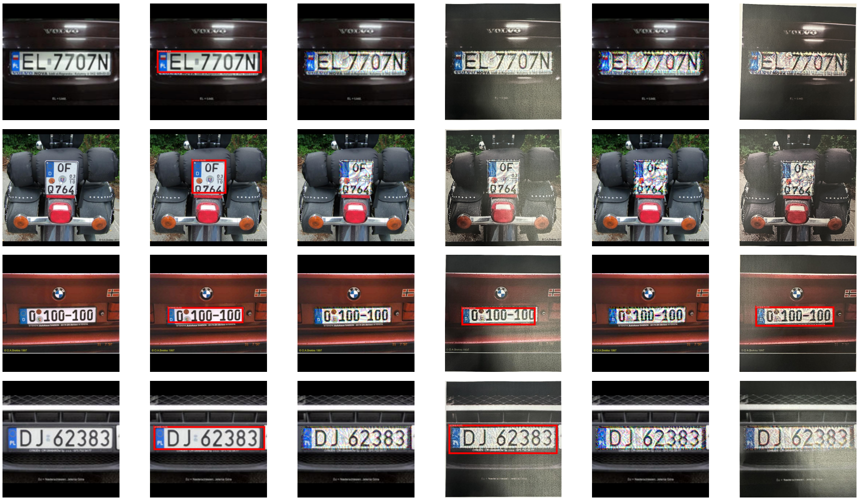

Figure 9: Each row displays, from left to right: the original (clean) license plate image, the detected license plate region, I-FGSM with EOT (before printing and recapturing), I-FGSM with EOT (after printing and recapturing), I-FGSM with EOT + P&S (before printing and recapturing), and I-FGSM with EOT + P&S (after printing and recapturing). In the first two rows, the fourth and sixth columns demonstrate successful attacks, where the adversarial examples remain effective after the physical transformation, leading to misclassification by the LPD model. In contrast, the third row shows a failed attack, where the perturbations do not survive the print-and-recapture process in either case. The fourth row highlights the importance of incorporating the P&S simulator within EOT, resulting in a successful misclassification by the LPD model when using I-FGSM with EOT + P&S.

of the printerspecific footprint after the attack during the printing process, which reduces the effectiveness of the adversarial perturbation. LPD depends on bold, highcontrast shapes, character strokes, and plate borders that remain intact during printing and scanning, whereas printer attribution models rely on fine halftone and sensornoise cues that the P&S process tends to eliminate. The LPD task thus involves a standard attack on a computer vision model without such constraints. This difference is reflected in our results, where attacks without P&Sbased EOT achieve higher success rates in the LPD setting.

7 Concluding Remarks

In this paper, we addressed the problem of generating robust physical domain adversarial examples that survive printing and scanning. The proposed attack is effective to attack source printer attribution systems. The specific challenge with this scenario is that the features associated to the forensic task, suppressed by the attack, are reintroduced in the reprinting stage. To cope with this, we introduced an attack that integrates P&S simulations within the EOT framework. By employing Pix2Pix GAN and CycleGAN models, we developed two simulators that accurately replicate the P&S transformations. The integration of these simulators into the EOT framework significantly increased the ASR, demonstrating the method's effectiveness in producing adversarial examples that survive reprinting. We further validated the versatility of our method by applying it to attack systems performing object detection, and in particular license plate detection. The experimental results show that incorporating P&S simulation in EOT improves the performance of the attack even in this case. The two application scenarios considered in this work are Printer Source Attribution (PSA) and License Plate Detection (LPD), as summarized in Table 7, which outlines the key settings and characteristics specific to each task. Our work underscores the importance of physical domain adversarial attacks in AI security research and provides a foundation for future efforts to counteract such threats.

Table 7: Summary of experimental settings for Printer Source Attribution (PSA) and License Plate Detection (LPD) tasks

Aspect	**Printer Source Attribution (PSA)**	**License Plate Detection (LPD)**
Target AI System	Source printer attribution classifier	SSD300-based license plate detector
Attack Objective	Induce misclassification of the printer source (Kyocera P5021 CDN)	Cause incorrect localization or suppression of license plate bounding boxes
Security Impact	Undermines the reliability of forensic printer authentication	Undermines the reliability of automatic license plate detection systems
Dataset Used	VIPPrint dataset	Aggregated dataset compiled from public LPD benchmarks
Generative AI Module	Differentiable P&S simulator trained using Pix2Pix and CycleGAN	Differentiable P&S simulator trained using CycleGAN
EOT Strategy	I-FGSM and C&W attacks combined with P&S simulation and physical transformations	I-FGSM attack combined with P&S simulation and physical transformations

Future work will focus on developing appropriate defenses, such as adversarial training techniques that incorporate examples of images subjected to the proposed physical domain attack. We also plan to expand our simulators to address various image processing tasks under diverse environmental conditions. Additionally, we aim to develop advanced P&S simulators using diffusion models for enhanced realism and accuracy. Eventually, we will explore other more real-world setups for license plate detection in the physical domain, with deployment-oriented evaluations obtained by crafting realistic license plates capable to evade detection. While our LPD experiments validate the attack under controlled indoor settings, extending to unconstrained outdoor scenarios with variable illumination, viewing angles, weather effects, and mounting the license plate on actual vehicles remains important future work.

Acknowledgments

This work was partially supported by SERICS (PE00000014) under the MUR National Recovery and Resilience Plan, funded by the European Union – NextGenerationEU, and the FOSTERER project, funded by the Italian Ministry of University and Research (PRIN 2022 program, contract 202289RHHP).

The authors would also like to sincerely thank Dennis Siegel and Stefan Seidlitz (Otto-von-Guericke University, Magdeburg) for their valuable insights, technical suggestions, and constructive discussions throughout the development of the license plate detection experiments during Nischay Purnekar's research stay at Otto-von-Guericke University, Magdeburg in 2024. Their feedback significantly contributed to refining the attack design and experimental methodology in this part of the work.

References

[1] A. Athalye, L. Engstrom, A. Ilyas, and K. Kwok, “Synthesizing Robust Adversarial Examples”, in *Proceedings of the 35th International Conference on Machine Learning,* 2018.

[2] N. Carlini and D. A. Wagner, “Towards Evaluating the Robustness of Neural Networks”, in *2017 IEEE Symposium on Security and Privacy, SP 2017*, 2017.

[3] S. Chen, C. Cornelius, J. Martin, and D. H. (Chau, “ShapeShifter: Robust Physical Adversarial Attack on Faster R-CNN Object Detector”, in *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2018, Dublin, Ireland, September 10-14, 2018, Proceedings, Part I*, Vol. 11051, *Lecture Notes in Computer Science*, Springer, 2018, 52–68.

[4] G. D. Evangelidis and E. Z. Psarakis, “Parametric Image Alignment Using Enhanced Correlation Coefficient Maximization”, *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(10), 2008, 1858–65.

[5] I. Evtimov, K. Eykholt, E. Fernandes, T. Kohno, B. Li, A. Prakash, A. Rahmati, and D. Song, “Robust Physical-World Attacks on Machine Learning Models”, *CoRR*, abs/1707.08945, 2017.

[6] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, and D. Song, “Robust Physical-World Attacks on Deep Learning Visual Classification”, in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, Computer Vision Foundation / IEEE Computer Society, 2018, 1625–34.

[7] Federal Trade Commission, “Consumer Sentinel Network Databook”, *tech. rep.*, 2023.

[8] M. Ferrara, A. Franco, and D. Maltoni, “Face morphing detection in the presence of printing/scanning and heterogeneous image sources”, *IET Biometrics*, 2021.

[9] A. Ferreira and M. Barni, “Attacking and Defending Printer Source Attribution Classifiers in the Physical Domain”, in *Pattern Recognition, Computer Vision, and Image Processing. ICPR*, 2022.

[10] A. Ferreira, E. Nowroozi, and M. Barni, “VIPPrint: A Large Scale Dataset of Printed and Scanned Images for Synthetic Face Images Detection and Source Linking”, *CoRR*, abs/2102.06792, 2021.

[11] A. Ferreira, N. Purnekar, and M. Barni, “Ensembling Shallow Siamese Neural Network Architectures for Printed Documents Verification in Data-Scarcity Scenarios”, *IEEE Access*, 9, 2021, 133924–39.

[12] M. Fredrikson, S. Jha, and T. Ristenpart, “Model Inversion Attacks that Exploit Confidence Information and Basic Countermeasures”, in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, 2015.

[13] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and Harnessing Adversarial Examples”, in *3rd International Conference on Learning Representations*, 2015.

[14] W. Guo, B. Tondi, and M. Barni, “An Overview of Backdoor Attacks Against Deep Neural Networks and Possible Defences”, *CoRR*, abs/2111.08429, 2021.

[15] L. Huang, C. Gao, Y. Zhou, C. Zou, C. Xie, A. L. Yuille, and N. Liu, “UPC: Learning Universal Physical Camouflage Attacks on Object Detectors”, *CoRR*, abs/1909.04326, 2019.

[16] Y. Huang, A. W. Kong, and K. Lam, “Adversarial Signboard against Object Detector”, in *30th British Machine Vision Conference 2019, BMVC 2019, Cardiff, UK, September 9-12, 2019*, BMVA Press, 2019, 231.

[17] P. Isola, J. Zhu, T. Zhou, and A. A. Efros, “Image-to-Image Translation with Conditional Adversarial Networks”, in *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2017.

[18] S. T. K. Jan, J. Messou, Y. Lin, J. Huang, and G. Wang, “Connecting the Digital and Physical World: Improving the Robustness of Adversarial Attacks”, in *The Thirty-Third AAAI Conference on Artificial Intelligence,* 2019.

[19] S. Komkov and A. Petiushko, “AdvHat: Real-World Adversarial Attack on ArcFace Face ID System”, in *25th International Conference on Pattern Recognition*, 2020.

[20] A. Kurakin, I. J. Goodfellow, and S. Bengio, “Adversarial examples in the physical world”, in *5th International Conference on Learning Representations*, 2017.

[21] M. Lee and J. Z. Kolter, “On Physical Adversarial Patches for Object Detection”, *CoRR*, abs/1906.11897, 2019.

[22] J. Li, F. R. Schmidt, and J. Z. Kolter, “Adversarial camera stickers: A physical camera-based attack on deep learning systems”, in *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, Vol. 97, *Proceedings of Machine Learning Research*, PMLR, 2019, 3896–904.

[23] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. E. Reed, C. Fu, and A. C. Berg, “SSD: Single Shot MultiBox Detector”, in *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part I*, Vol. 9905, *Lecture Notes in Computer Science*, Springer, 2016, 21–37.

[24] J. Lu, H. Sibai, and E. Fabry, “Adversarial Examples that Fool Detectors”, *CoRR*, abs/1712.02494, 2017.

[25] J. Lu, H. Sibai, E. Fabry, and D. A. Forsyth, “NO Need to Worry about Adversarial Examples in Object Detection in Autonomous Vehicles”, abs/1707.03501, 2017.

[26] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, “Towards Deep Learning Models Resistant to Adversarial Attacks”, in *International Conference on Learning Representations (ICLR)*, 2018.

[27] A. Mitkovski, J. Merkle, C. Rathgeb, B. Tams, K. Bernardo, N. E. Haryanto, and C. Busch, “Simulation of Print-Scan Transformations for Face Images based on Conditional Adversarial Networks”, in *BIOSIG*, 2020.

[28] N. Purnekar, L. Abady, B. Tondi, and M. Barni, “Improving the Robustness of Synthetic Images Detection by Means of Print and Scan Augmentation”, in *Proceedings of the ACM Workshop on Information Hiding and Multimedia Security, IH&MMSec*, 2024.

[29] N. Purnekar, B. Tondi, and M. Barni, “Physical Domain Adversarial Attacks Against Source Printer Image Attribution”, in *Asia Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA ASC 2024, Macau, December 3-6, 2024*, IEEE, 2024, 1–6.

[30] S. Seidlitz and J. Dittmann, “Forensic Analysis of GAN Training and Generation: Output Artifacts Assessment of Circles and Lines”, in *Proceedings of SECURWARE 2024, The Eighteenth International Con-*

ference on Emerging Security Information, Systems and Technologies, 2024.

[31] M. Sharif, S. Bhagavatula, L. Bauer, and M. K. Reiter, "Accessorize to a Crime: Real and Stealthy Attacks on State-of-the-Art Face Recognition", in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016.

[32] C. Sitawarin, A. N. Bhagoji, A. Mosenia, P. Mittal, and M. Chiang, "Rogue Signs: Deceiving Traffic Sign Recognition with Malicious Ads and Logos", *CoRR*, abs/1801.02780, 2018.

[33] D. Song, K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, F. Tramèr, A. Prakash, and T. Kohno, "Physical Adversarial Examples for Object Detectors", in *12th USENIX Workshop on Offensive Technologies, WOOT 2018, Baltimore, MD, USA, August 13-14, 2018*, ed. C. Rossow and Y. Younan, USENIX Association, 2018.

[34] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. J. Goodfellow, and R. Fergus, "Intriguing properties of neural networks", in *2nd International Conference on Learning Representations*, 2014.

[35] A. Vassilev, A. Oprea, A. Fordyce, and H. Anderson, "Adversarial Machine Learning: A Taxonomy and Terminology of Attacks and Mitigations", *NIST AI 100-2 E2023*, National Institute of Standards and Technology (NIST), 2024.

[36] C. Xie, J. Wang, Z. Zhang, Y. Zhou, L. Xie, and A. L. Yuille, "Adversarial Examples for Semantic Segmentation and Object Detection", in *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, IEEE Computer Society, 2017, 1378–87.

[37] K. Yang, J. Liu, C. Zhang, and Y. Fang, "Adversarial Examples Against the Deep Learning Based Network Intrusion Detection Systems", in *2018 IEEE Military Communications Conference, MILCOM 2018, Los Angeles, CA, USA, October 29-31, 2018*, IEEE, 2018, 559–64.

[38] K. Yang, T. Tsai, H. Yu, T. Ho, and Y. Jin, "Beyond Digital Domain: Fooling Deep Learning Based Recognition System in Physical World", in *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, AAAI Press, 2020, 1088–95.

[39] X. Yuan, Y. Chen, Y. Zhao, Y. Long, X. Liu, K. Chen, S. Zhang, H. Huang, X. Wang, and C. A. Gunter, "CommanderSong: A Systematic Approach for Practical Adversarial Voice Recognition", in *27th USENIX Security Symposium, USENIX Security 2018, Baltimore, MD, USA, August 15-17, 2018*, ed. W. Enck and A. P. Felt, 49–64.

[40] B. Zhang, B. Tondi, and M. Barni, “Adversarial examples for replay attacks against CNN-based face recognition with anti-spoofing capability”, *Comput. Vis. Image Underst.*, 197-198, 2020, 102988.

[41] Y. Zhao, H. Zhu, R. Liang, Q. Shen, S. Zhang, and K. Chen, “Seeing isn’t Believing: Towards More Robust Adversarial Attack Against Real World Object Detectors”, in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, CCS 2019, London, UK, November 11-15, 2019*, ACM, 2019, 1989–2004.

[42] J. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks”, in *IEEE International Conference on Computer Vision, ICCV*, 2017.