Universal Physical Camouflage Attacks on Object Detectors

Lifeng Huang[1,2] Chengying Gao[1] Yuyin Zhou[3] Cihang Xie[3] Alan Yuille[3] Changqing Zou[4,5] Ning Liu[1,2 *]
[1]School of Data and Computer Science, Sun Yat-sen University
[2]Guangdong Key Laboratory of Information Security Technology
[3]Department of Computer Science, The Johns Hopkins University
[4]Max Planck Institute for Informatics, [5]University of Maryland, College Park
huanglf6@mail2.sysu.edu.cn, {mcsgcy, liuning2}@mail.sysu.edu.cn,
{zhouyuyiner, cihangxie306, alan.l.yuille, aaronzou1125}@gmail.com

Abstract

In this paper, we study physical adversarial attacks on object detectors in the wild. Previous works mostly craft instance-dependent perturbations only for rigid or planar objects. To this end, we propose to learn an adversarial pattern to effectively attack ***all instances belonging to the same object category****, referred to as Universal Physical Camouflage Attack (UPC). Concretely, UPC crafts camouflage by jointly fooling the region proposal network, as well as misleading the classifier and the regressor to output errors. In order to make UPC effective for non-rigid or non-planar objects, we introduce a set of transformations for mimicking deformable properties. We additionally impose optimization constraint to make generated patterns look natural to human observers. To fairly evaluate the effectiveness of different physical-world attacks, we present the first standardized virtual database,* **AttackScenes***, which simulates the real 3D world in a controllable and reproducible environment. Extensive experiments suggest the superiority of our proposed UPC compared with existing physical adversarial attackers not only in virtual environments (***AttackScenes***), but also in real-world physical environments. Code and dataset are available at* `https://mesunhlf.github.io/index_physical.html`.

1. Introduction

Deep neural networks (DNNs) have achieved outstanding performances on many computer vision tasks [37, 8, 10]. Nonetheless, DNNs have been demonstrated to be vulnerable to adversarial examples [38] — maliciously crafted inputs that mislead DNNs to make incorrect predictions, which present potential threats for the deployment of DNN-based systems in the real world.

*Corresponding author.

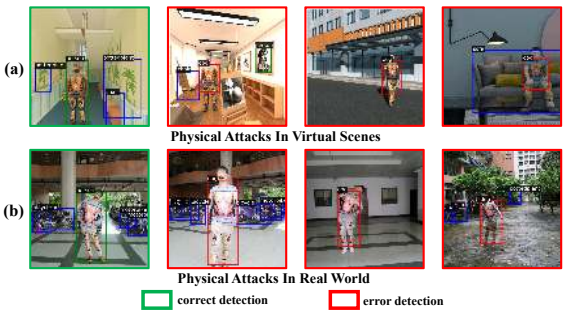


Figure 1. **Fooling the object detector, faster r-cnn, in the physical space.** (a) Physical attacks (UPC) in virtual scenes and (b) Physical attacks (UPC) in real world. Column 1 shows detection results with natural patterns. Column 2-4 display results with camouflage patterns under different viewing conditions.

Adversarial attacks [26, 2] in general can be divided into the following categories: 1) **digital attacks**, which mislead DNNs by modifying the input data directly in the digital space (*e.g.*, pixel value [26, 11, 23], text content [15, 31]); 2) **physical attacks**, which attack DNNs by altering **visible** characteristics of an object (*e.g.*, color [33], appearance [6]) in the physical world. Current mainstream works focus on the digital domain, which can be hardly transferred to the real world due to the lack of considering physical constraints (*e.g.*, invariant to different environmental conditions such as viewpoint, lighting) [6]. In this paper, we study adversarial attacks in the physical world, which are more threatening to real-world systems [14]. Compared with previous works [3, 12, 1] which mostly focus on attacking image classification systems, we consider the far more realistic computer vision scenario, *i.e.*, object detection.

Though prior works have revealed the vulnerability of object detectors to adversarial perturbations in the real world [4, 36, 46], there are several limitations: (1) focusing on only attacking a specific object (*e.g.* a stop sign [6, 4] , commercial logo [35] or car [46]); (2) generating perturbations only for rigid or planar objects (*e.g.*, traffic sign,

vehicle body, board [39]), which can be less effective for complex objects (articulated non-rigid or non-planar objects, *e.g*., human). (3) constructing meaningless which lack semantics and appear unnatural for human observers (*i.e*., noisy or mosaic-like texture) [4, 39, 46]; and (4) a unified evaluation environment is missing, which makes it difficult to make fair comparisons between different attacks.

To address these issues, we present Universal Physical Camouflage Attack (UPC), which constructs a universal camouflage pattern to hide objects from being detected or to misdetect objects as the target label. Unlike former works which generate instance-level perturbations, UPC constructs a universal pattern to attack all instances that belong to the same category (*e.g*., person, cars) via jointly attacking the region proposal network, the classifier and the regressor. To efficiently handle the deformations of complex objects in the physical world, we propose to model their deformable characteristics as well as external physical environments in UPC. Specifically, the internal properties are simulated by applying various geometric transformations (*e.g*., cropping, resizing, affine homography). We impose additional optimization constraint to encourage the visual resemblance between generated patterns and natural images, which we refer to as *semantic constraint*. As shown in Fig. 1, these camouflage patterns are visually similar to natural images and thus can be regarded as texture patterns on object surfaces such as human accessories/car paintings. The overall pipeline is illustrated in Fig. 2.

To fairly evaluate the effectiveness of different physical attacks, we provide the first standardized synthetic dataset, *i.e*., ***AttackScenes***. All experimental data is generated under strict parametric-controlled physical conditions to ensure that the evaluation is reliable under virtual settings.

The contributions of our work are four-fold:

- UPC constructs a universal camouflage pattern for effectively attacking object detectors based on the fact that the generated pattern can be naturally camouflaged as texture patterns on object surfaces such as human accessories/car paintings.
- We present the first standardized dataset, ***AttackScenes***, which is simulates the real 3D world under controllable and reproducible settings, to ensure that all experiments are conducted under fair comparisons for future research in this domain.
- To make UPC effective for articulated non-rigid or non-planar objects, we introduce additional transformations for the camouflage patterns to simulate their internal deformations.
- Our proposed UPC not only achieves state-of-the-art result for attacking object detectors in the wild, but also exhibits well generalization and transferability among different models.

Table 1. Comparison with existing methods.

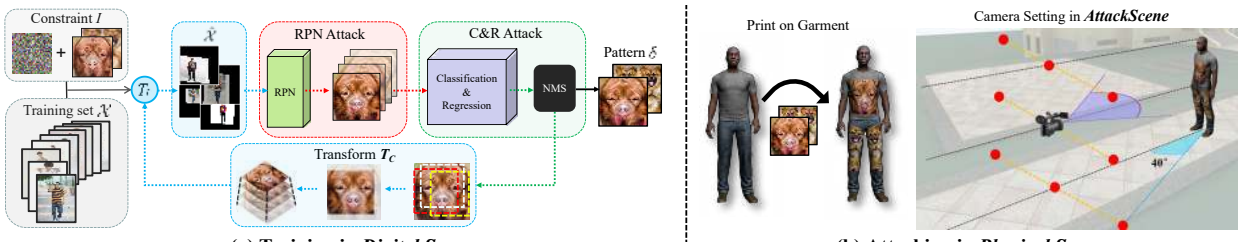
Methods	Rigid	Non-Rigid	Planar	Non-Planar	Universal	Semantic
[4]	✓		✓			
[36]	✓		✓			
[46]	✓		✓	✓		
Ours	✓	✓	✓	✓	✓	✓

2. Related Works

Universal Adversarial Attack. *Image-agnostic* attack, *i.e*., universal adversarial attack [25, 13], is defined as an attack which is able to fool different images with a single global pattern in the digital domain. Here we extend this definition to the physical domain and define *instance-agnostic* perturbations as universal physical attacks for object detectors. Unlike former physical attack methodologies which craft instance-level patterns, our goal is to generate a single camouflage pattern to effectively attack *all instances of the same object category* given different physical scenes.

Physical Attacks. Stem from the recent observation that printed adversarial examples can fool image classifiers in the physical world [14, 12], efforts have been investigated to study how to construct "robust" adversarial examples in the real physical world. For instance, Athalye *et al*. [1] propose to construct 3D adversarial objects by attacking an ensemble of different image transformations; Sharif *et al*. [33] successfully attack facial recognition systems by printing textures on eyeglasses; Evtimov *et al*. [6] use poster, sticker and graffiti as perturbations to attack stop signs in the physical world. Zeng *et al*. [45] apply computer graphics rendering methods to perform attacks in the 3D physical world. In addition, adversarial attacks also extend to fool tracking system and Re-Identification models [40, 41].

Recently, physical attacks have also been studied for the more challenging scenario of object detection. Song *et al*. [36] propose a disappearance and creation attack to fool Yolov2 [28] in traffic scenes. Chen *et al*. [4] adopt the expectation over transformation method [1] to create more robust adversarial stop signs, which mislead faster r-cnn [30] to output errors. Zhang *et al*. [46] learn the clone network to approximate detectors under black-box scenerio. However they cannot be effectively applied to non-rigid or non-planar objects since they only focus on simulating external environment conditions, *e.g*., distances or viewpoints, for attacking object detectors. In addition, these approaches generate instance-dependent patterns which exhibit less semantics and therefore the perturbed images are usually unnatural and noisy. Different from these works, our method constructs a universal semantic pattern which makes the perturbed images visually similar to natural images. Meanwhile, we introduce additional transformations to simulate the deformable properties of articulated non-rigid or non-planar objects. A detailed comparison with former methods is summarized in Table. 1.



(a) Training in *Digital Space* **(b) Attacking in *Physical Space***

Figure 2. The overall pipeline of UPC. (a) training the camouflage patterns in digital space; (b) attacking the target in physical space.

3. Methodology

3.1. Overview

Our goal is to attack object detectors by either hiding the object from being detected, or fooling detectors to output the targeted label. Without loss of generality, we use "person" category as an example to illustrate our method.

Training framework of UPC in Digital Space. We attack faster-rcnn [30], a two-stage detector, under white-box settings. In the first stage, the region proposal network is employed to generate object proposals. In the second stage, the detector selects top-scored proposals to predict labels. We propose to craft a universal pattern for faster-rcnn by jointly fooling the region proposal network to generate low-quality proposals, *i.e.*, reduce the number of valid proposals, as well as misleading the classifier and the regressor to output errors. Simply misleading predictions of the classification head cannot produce satisfying results (discussed in Sec. 5.2) because it can be impractical to attack enormous candidate proposals simultaneously. Extensive experimental results also validate that the joint attack paradigm demonstrates stronger attacking strength than simply attacking the classifier as in prior methods [4, 6] (Table 3). Furthermore, to deal with complex objects, we propose to simultaneously model both internal deformable properties of complex objects and external physical environments. The internal attributes of objects, *i.e.*, deformations, are simulated by a series of geometric transformations. As illustrated in Fig. 2(a), UPC consists of 3 steps:

- **Step 1**. A set of perturbed images are synthesized by simulating external physical conditions (*e.g.*, viewpoint) as well as internal deformations of complex objects. An additional optimization constraint is imposed to make the generated patterns semantically meaningful (Sec. 3.2).
- **Step 2**. Initial adversarial patterns are generated by attacking the RPN, which results in a significant drop of high-quality proposals (Sec. 3.3).
- **Step 3**. To enhance the attacking strength further, UPC then jointly attacks RPN as well as the classification and the bounding box regression head by lowering the detection scores and distorting the bounding box (Sec. 3.4).

We perform these steps in an iterative manner until the termination criterion is satisfied, *i.e.*, fooling rate is larger than the threshold or the iteration reaches the maximum.

Attacking in Physical Space. By imposing the semantic constraint (Sec. 3.2), the generated camouflage patterns by UPC look natural for human observers and thus can be regarded as texture patterns on human accessories. Concretely, we pre-define several regions of human accessories (*e.g.*, garment, mask) to paint on the generated camouflage patterns (Fig. 4) for attacking, and the corresponding physical scenes are captured under different viewing conditions (*e.g.*, illumination, viewpoints) for testing (Fig. 2(b)).

3.2. Physical Simulation

Material Constraint. To keep generated adversarial patterns less noticeable, the perturbations are camouflaged as texture patterns on human accessories (*e.g.*, garment, mask). External environments are simulated via controlling factors such as lighting, viewpoint, location and angle [4, 6]. To effectively handle non-rigid or non-planar objects, we also introduce addition transformation functions to model their internal deformations (Eq. 2).

Semantic Constraint. Inspired by the imperceptibility constraint in digital attacks, we use the projection function (Eq. 1) to enforce the generated adversarial patterns to be visually similar to natural images during optimization. Empirical results show that optimizing with this constraint yields high-quality semantic patterns, which can be naturally treated as camouflages on human clothing (Fig. 8).

Training Data. To obtain universal patterns, images with different human attributes (body sizes, postures, *etc.*) are sampled as the training set $\mathcal{X}$.

In summary, the perturbed images are generated by:

$$\delta^t = Proj_\infty(\delta^{t-1} + \Delta\delta, I, \epsilon), \tag{1}$$

$$\hat{\mathcal{X}} = \left\{\hat{x}_i | \hat{x}_i = T_r(x_i + T_c(\delta^t)), x_i \sim \mathcal{X}\right\}. \tag{2}$$

Eq. 1 is the semantic constraint, where δ^t and $\Delta\delta$ denote the adversarial pattern and its updated vector at iteration t, respectively. $Proj_\infty$ projects generated pattern onto the surface of L_∞ norm-balls with radius ϵ and centered at I. Here we choose I as natural images to ensure the generated camouflage patterns are semantically meaningful. Eq. 2 is the

physical simulation we applied during the attack, where T_r is applied to all training images and used for the environmental simulation (*e.g.*, illumination). T_c is acted on generated patterns, which is used for modeling the material constraint (*e.g.*, deformations induced by stretching). $\hat{x}$ is the generated perturbed image (marked as blue in Fig. 2(a)).

3.3. Region Proposal Network (RPN) Attack

For an input image with height H and width W, the RPN extracts $M = O(HW)$ proposals across all anchors. We denote the output proposals of each image $\hat{x}$ as $\mathcal{P} = \left\{p_i | p_i = (s_i, \vec{d}_i); i = 1, 2, 3...M\right\}$, where s_i is the confidence score of i-th bounding box and $\vec{d}_i$ represents the coordinates of i-th bounding box. We define the objective function for attacking the RPN as following:

$$L_{rpn} = \mathop{\mathbb{E}}_{p_i \sim \mathcal{P}} (\mathcal{L}(s_i, y^t) + s_i \|\vec{d}_i - \Delta\vec{d}_i\|_p), \tag{3}$$

where y^t is the target score, and we set y^1 for background and y^0 for foreground; $\mathcal{L}$ is the Euclidean distance loss; $\Delta\vec{d}_i$ is a pre-difined vector, which used for attacking proposals by shifting the center coordinate and corrupting the shape of original proposals; p is the norm constant and we set $p = 1$ in the experiment.

By minimizing L_{rpn}, our goal is to generate adversarial patterns for RPN which results in a substantial reduction of foreground proposals and severely distorted candidate boxes (marked as red in Fig. 2(a)).

3.4. Classifier and Regressor Attack

After applying non-maximum suppression (NMS) on the outputs of RPN, top-k proposals are ordered by their confidence scores and selected as a subset $\hat{\mathcal{P}}$. These top-scored proposals $\hat{\mathcal{P}}$ are then fed to the classification and the regression head for generating final outputs. We note that if only a subset of proposed bounding boxes are perturbed, the detection result of the attacked image may still be correct if a new set of candidate boxes is picked in the next iteration, which results in great challenges for attackers. To overcome this issue, we instead extract proposals densely as in [43]. Specifically, we attack an object by either decreasing the confidence of the groundtruth label or increasing the confidence of the target label. We further enhance the attacking strength by distorting the aspect ratio of proposals and shifting the center coordinate simultaneously [17]. In summary, we attack the classification and the regression head by:

$$L_{cls} = \mathop{\mathbb{E}}_{p \sim \hat{\mathcal{P}}} C(p)_y + \mathop{\mathbb{E}}_{p \sim \mathcal{P}^*} \mathcal{L}(C(p), y'), \tag{4}$$

$$L_{reg} = \sum_{p \sim \mathcal{P}^*} \|R(p)_y - \Delta\vec{d}\|_l, \tag{5}$$

where $\mathcal{L}$ is the cross-entropy loss, C and R are the prediction output of the classifier and the regressor. $\mathcal{P}^*$ is the proposals which can arybe detected as true label y, and y' is the target label for attacking. $\Delta\vec{d}$ denotes the distortion offset. We select ℓ_2 norm, *i.e.*, $l = 2$ in Eq. 5. Eq. 4 and Eq. 5 are designed for fooling the classifier and the regressor, respectively, and are referred to as C&R attack (marked as green in Fig. 2(a)). For **untargeted attack**, we set $y = y'$ for maximizing (instead of minimizing) Eq. 4.

Algorithm 1 Algorithm of UPC

Input: Training images $\mathcal{X}$; Target label y'; Balance parameters λ_1, λ_2; Iteration parameters $iter_s$ and $iter_{max}$; Fooling rate threshold r_s;
Output: Universal adversarial pattern δ; Fooling rate r;

$\delta^0 \leftarrow random$, $\Delta\delta \leftarrow 0$, $r \leftarrow 0$, $t \leftarrow 0$
while $t < iter_{max}$ and $r < r_s$ **do**
  $t \leftarrow t + 1$, $\delta^t \leftarrow Proj_\infty(\delta^{t-1} + \Delta\delta, I, \epsilon)$
  for all $x_i \sim \mathcal{X}$ **do**
    Choose the transformation of T_r and T_c randomly
    $\hat{x_i} = clip\ (T_r(x_i + T_c(\delta^t)), 0, 1)$
  end for
  Caculate the fooling rate r of perturbed images $\hat{\mathcal{X}}$
  if $t < iter_s$ and $r < r_s$ **then**
    $\underset{\Delta\delta}{argmin} \underset{\hat{x_i} \sim \hat{\mathcal{X}}}{\mathbb{E}} L_{rpn} + L_{tv}$
  else
    $\underset{\Delta\delta}{argmin} \underset{\hat{x_i} \sim \hat{\mathcal{X}}}{\mathbb{E}} (L_{rpn} + \lambda_1 L_{cls} + \lambda_2 L_{reg}) + L_{tv}$
  end if
end while

3.5. Two-Stage Attacking Procedure

In summary, UPC generates the physical universal adversarial perturbations by considering all the factors above:

$$\underset{\Delta\delta}{argmin} \mathop{\mathbb{E}}_{\hat{x} \sim \hat{\mathcal{X}}} (L_{rpn} + \lambda_1 L_{cls} + \lambda_2 L_{reg}) + L_{tv}(\delta^t), \tag{6}$$

where δ and $\hat{\mathcal{X}}$ denote the universal pattern and the set of perturbed images, respectively. L_{tv} stands for the **total variation loss** [24] with ℓ_2 norm constraint applied. We note that L_{tv} is important for reducing noise and producing more natural patterns.

The overall procedure of UPC is illustrated in Algorithm 1, where we alternately update the universal perturbation pattern δ and the perturbed images $\hat{x} \sim \hat{\mathcal{X}}$ until the fooling rate becomes larger than a certain threshold or the attack iteration reaches the maximum. δ is updated using a two-stage strategy. During the first stage, we exclusively attack the RPN to reduce the number of valid proposals, *i.e.*, set $\lambda_1 = 0$ and $\lambda_2 = 0$ in Eq. 6. After significantly reducing the number of high-quality proposals, our attack then additionally fools the classification and bounding box regression head in the second stage. By minimizing Eq. 6, the generated perturbation δ substantially lowers the quality of proposals and thereby achieves a high fooling rate.

Figure 3. **Examples of virtual scene experiments.** Virtual scenes (*i.e.*, ***AttackScenes***) are shown in the first row, including indoors and outdoors environments. The second rows shows results captured under various physical conditions with different pattern schemes.

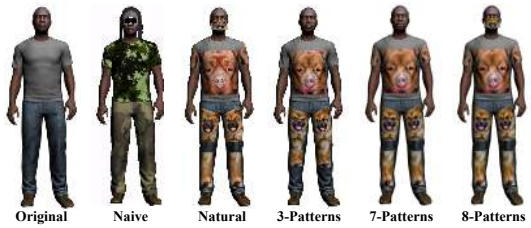


Figure 4. **Examples of pattern schemes in the virtual scenes experiment. Original:** humans without camouflage patterns; **Naive:** humans with simple camouflages (*i.e.*, army camouflage cloths, pilot cap and snow goggles); **Natural:** humans with natural images as camouflage patterns. **3/7/8-Patterns:** according to the heatmaps of detection models, we pre-define 3/7/8 regions on human accessories to paint on the generated camouflage patterns.

4. AttackScenes Dataset

Due to the lack of a standardized benchmark dataset, earlier works measure the performance under irreproducible physical environments, which makes it difficult to make fair comparisons between different attacks. To this end, we build the first standardized dataset, named ***AttackScenes***, for fair and reproducible evaluation.

Environments. ***AttackScenes*** includes 20 virtual scenes under various physical conditions (Fig. 3). Specifically, there are 10 indoors scenes (*e.g.*, bathroom, living room) and 10 outdoors scenes(*e.g.*, bridge, market) in total.

Camera Setting. For each virtual scene, 18 cameras are placed for capturing images from different viewpoints. To ensure the diversity of images, these cameras are located at different angles, heights and distances (Fig. 2(b)).

Illumination Control. To the best of our knowledge, earlier studies usually conduct tests in bright environments. However, this simulated condition is quite limited since there exist many dark scenes in the real world. Accordingly, we extend the testing environment to better simulate different daily times like evening and dawn. Area lights and directional light sources are used to simulate indoors and outdoors illuminations, respectively. The illumination varies from dark to bright at 3 levels by controlling the strength of light sources (*i.e.*, L1∼L3).

5. Experiments

In this section, we empirically show the effectiveness of the proposed UPC by providing thorough evaluations in both virtual and physical environments.

5.1. Implementation Details

We mainly evaluate the effectiveness of our method on "person" category due to its importance in video surveillance and person tracking [16]. We collect 200 human images with various attributes (*e.g.*, hair color, body size) as our training set to generate universal adversarial patterns. Following [43], we evaluate the performance of faster r-cnn using 2 network architectures (*i.e.*, VGG-16 [34] and ResNet-101[8]) which are either trained on the PascalVOC-2007 `trainval`, or on the combined set of PascalVOC-2007 `trainval` and PascalVOC-2012 `trainval`. We denote these models as FR-VGG16-07, FR-RES101-07, FR-VGG16-0712 and FR-RES101-0712.

Parameters setting. We set fooling rate threshold $r_s = 0.95$, $iter_s = 100$ and the maximum iteration $iter_{max} = 2000$ in Algorithm 1. More parameters and transformation details are recorded in sec. 1 of supplementary material.

Evaluation Metric. For faster r-cnn, we set the threshold of NMS as 0.3 and the confidence threshold as 0.5 (instead of the default value 0.8). Even though IoU is used for standard evaluation of object detection, we do not use this metric here since our focus is whether the detector hits or misses the true label of the attacked instance. To this end, we extend the metrics in [4, 6] to be applicable in our experiments, precision $p_{0.5}$, to measure the probability of whether the detector can hit the true category:

$$p_{0.5} = \frac{1}{|\mathcal{X}|} \sum_{v\sim\mathbb{V},b\sim\mathbb{B},s\sim\mathbb{S}} \left\{ \underset{x\in\mathcal{X}}{C(x) = y}, \underset{\hat{x}\in\hat{\mathcal{X}}}{C(\hat{x}) = y} \right\}, \quad (7)$$

where x is the original instance and $\hat{x}$ denotes the instance with camouflage patterns. $\mathbb{V}, \mathbb{L}, \mathbb{S}$ denote the sets of camera viewpoints, brightness and scenes, respectively; C is the prediction of detector and y is the groundtruth label (*i.e.*, person, car).

Table 2. Average precision $p_{0.5}$ in virtual scene experiments after attacking faster r-cnn. Note that $p_{0.5}$ is averaged over all viewpoints of each pattern scheme under 3 brightness conditions.

Network	FR-VGG16-0712				FR-RES101-0712			
Schemes	Standing				Standing			
	L1	L2	L3	Avg (Drop)	L1	L2	L3	Avg (Drop)
Original	0.97	0.97	1.0	0.98 (-)	0.99	0.99	1.0	0.99 (-)
Naive	0.97	0.97	0.99	0.97 (0.01)	0.99	0.99	0.99	0.99 (0.0)
Natural	0.95	0.96	0.98	0.96 (0.02)	0.97	0.97	0.98	0.97 (0.02)
3-Patterns	0.64	0.36	0.18	0.39 (**0.59**)	0.73	0.69	0.70	0.69 (**0.30**)
7-Patterns	0.55	0.33	0.22	0.37 (**0.61**)	0.51	0.48	0.64	0.54 (**0.45**)
8-Patterns	0.15	0.03	0.02	0.07 (**0.91**)	0.10	0.09	0.13	0.11 (**0.88**)
Schemes	Walking				Walking			
	L1	L2	L3	Avg (Drop)	L1	L2	L3	Avg (Drop)
Original	0.93	0.94	0.99	0.95 (-)	0.98	0.99	1.0	0.99 (-)
Naive	0.92	0.94	0.96	0.94 (0.01)	0.98	0.97	0.98	0.98 (0.01)
Natural	0.91	0.93	0.95	0.93 (0.02)	0.98	0.99	0.98	0.98 (0.01)
3-Patterns	0.37	0.26	0.16	0.26 (**0.69**)	0.44	0.50	0.50	0.48 (**0.51**)
7-Patterns	0.28	0.25	0.16	0.23 (**0.72**)	0.31	0.33	0.34	0.33 (**0.66**)
8-Patterns	0.06	0.05	0.01	0.04 (**0.91**)	0.05	0.06	0.06	0.06 (**0.93**)
Schemes	Sitting				Sitting			
	L1	L2	L3	Avg (Drop)	L1	L2	L3	Avg (Drop)
Original	0.97	0.99	0.99	0.98 (-)	1.0	0.99	0.99	0.99 (-)
Naive	0.93	0.94	0.95	0.94 (0.04)	0.93	0.92	0.93	0.93 (0.06)
Natural	0.94	0.94	0.98	0.95 (0.03)	0.97	0.98	0.98	0.98 (0.01)
3-Patterns	0.83	0.64	0.63	0.70 (**0.28**)	0.75	0.77	0.79	0.77 (**0.22**)
7-Patterns	0.83	0.77	0.63	0.74 (**0.24**)	0.77	0.78	0.78	0.78 (**0.21**)
8-Patterns	0.60	0.47	0.32	0.46 (**0.52**)	0.49	0.57	0.62	0.56 (**0.43**)

5.2. Virtual Scene Experiment

Human Model and Pattern Schemes. We select human models in ***AttackScenes*** with different poses (*i.e.*, standing, walking and sitting) as the attacking target. 6 different schemes (Fig. 4) are used under the material constraint (Sec. 3.2) for experimental comparison.

Comparison Between Pattern Schemes. In the virtual scene experiment, $1080(20\times3\times18)$ images are rendered for each pattern scheme. Without loss of generality, we choose "dog" and "bird" as target labels to fool detectors in our experiment. We use 6 different pattern schemes illustrated in Fig. 4 for validating the efficacy of the proposed UPC.

As shown in Table 2, we find that the attack strength is generally weaker in darker environments. This can be attributed to the fact that the adversarial patterns are badly captured when the level of brightness is low, which induces low-quality attacks. Additionally, we observe that for different human poses the average precision almost stays at the same level via attacking Naive/Natural pattern scheme which indicates that simply using naive camouflage or natural images as adversarial patterns is invalid for physical attacks. By contrast, our method yields a distinct drop rate of $p_{0.5}$ for all 3 pattern schemes (*i.e.*, 3/7/8-Pattern schemes), among which 8-Pattern scheme observes the highest performance drop (*i.e.*, Standing: $p_{0.5}$ drops from 0.98 to 0.07 using FR-VGG16). It is no surprise to observe such a phenomenon since using more generated patterns for physical attack results leads to a higher fooling rate. The detection result further shows our attack is invariant to different viewing conditions (*e.g.*, viewpoints, brightness). Additionally, we also find that among these 3 poses "Sitting" is the most difficult to attack since some patterns (*e.g.*, pants or cloth patterns) are partially occluded (see sampled images from Fig. 1 and Fig. 3).

Table 3. Performance comparison with prior arts of physical attacks under different settings. We record $p_{0.5}$ and drop rate averaged over all viewpoints of 8-pattern scheme.

Network	FR-VGG16-0712		
Setup	Standing	Walking	Sitting
UPC_{rc} **(ours)**	0.07 (**0.91**)	0.04 (**0.91**)	0.46 (**0.52**)
UPC_r **(ours)**	0.66 (0.32)	0.33 (0.62)	0.76 (0.22)
CLS_{rc} **(ours)**	0.18 (0.80)	0.06 (0.89)	0.54 (0.44)
$Shape$ [4]	0.70 (0.28)	0.39 (0.56)	0.78 (0.20)
ERP^2 [6]	0.85 (0.13)	0.48 (0.47)	0.87 (0.11)
$AdvPat$ [39]	0.77 (0.21)	0.31 (0.64)	0.78 (0.20)
Network	FR-RES101-0712		
Setup	Standing	Walking	Sitting
UPC_{rc} **(ours)**	0.11 (**0.88**)	0.06 (**0.93**)	0.56 (**0.43**)
UPC_r **(ours)**	0.73 (0.26)	0.42 (0.57)	0.86 (0.13)
CLS_{rc} **(ours)**	0.30 (0.69)	0.16 (0.83)	0.65 (0.34)
$Shape$ [4]	0.83 (0.16)	0.47 (0.52)	0.88 (0.11)
ERP^2 [6]	0.79 (0.20)	0.44 (0.55)	0.91 (0.08)
$AdvPat$ [39]	0.91 (0.08)	0.71 (0.28)	0.93 (0.06)

Compare with Existing Attacks. We compare UPC with existing physical attacks under the following settings (Table 3): (1) both internal deformations T_c and external physical environments T_r are simulated in Eq. 2, denoted as UPC_{rc}; (2) only external physical environments are modeled, *i.e.*, T_r is used in Eq. 2, denoted as UPC_r. (3) only attack the classification head, *i.e.*, L_{cls} is used to generate patterns, denoted as CLS_{rc}; (4) ShapeShifter [4], *i.e.*, only use T_r in Eq. 2 and attack against the classifier, denoted as $Shape$. (5) we follow [36] by extending RP^2 [6] for attacking faster r-cnn, denoted as ERP^2, and (6) Adversarial Patches [39], which utilize various transformations to fool all proposals across images, denote as $AdvPat$. These six scenarios were tested under same training setup (detailed in sec.1 of supplementary material).

The performance of 8-patterns scheme is recorded in Table 3, and the implications are two-fold. First, we can see the drop rates of $p_{0.5}$ in UPC_{rc} and CLS_{rc} are significantly higher than those of UPC_r, SS and ERP^2. These quantitative results indicate that the proposed transformation function T_c can effectively mimic the deformations (*e.g.*, stretching) of complex objects. Second, UPC_{rc} and UPC_r outperform CLS_{rc} and $Shape$, which suggest that the joint attack paradigm (*i.e.*, RPN and C&R attack) generally shows stronger attacking strength than only attacking the classification head [4]. In conclusion, all these experimental results demonstrate the efficacy of the proposed transformation term T_c as well as the joint attack paradigm for fooling object detectors in the wild. Moreover, our proposed UPC outperforms existing methods [4, 6, 39], and thereby establish state-of-the-art for physical adversarial attack on proposal-based object detectors.

The visualization of discriminative regions are showed in supplementary material [32]. We can observe that the UPC has superior attacking capability while other methods can not depress the activated features of un-occluded parts effectively, which may lead higher detection accuracy.

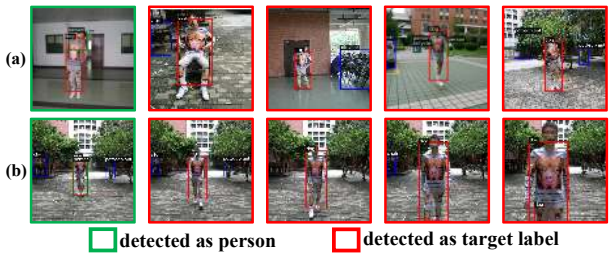


Figure 5. **Experimental results in (a) stationary testing and (b) motion testing.** The camouflage is generated by FR-VGG16.

5.3. Physical Environment Experiment

Following the setup of virtual scene experiments, we stick the same camouflage pattern on different volunteers with diverse body sizes and garment styles. During the physical experiment, we use Sony$\alpha 7r$ camera to take photos and record videos. Our physical experiments include two parts: **stationary testing** and **motion testing**.

Stationary Testing. In the physical world, we choose 5 scenes including indoors and outdoors scenes under different lighting conditions. Similar to virtual scene experiments, we take 18 photos of the attacked person for each pattern scheme. To evaluate the robustness of our method under different deformations, the person is required to switch from 6 different poses (*i.e.*, standing, sitting, leg lifting, waving hands, fork waist, shaking head) during photographing (Fig. 5(a)). We record the average precision $p_{0.5}$ and drop rates of FR-VGG16-0712 and FR-RES101-0712 under three brightness conditions in Table 4 (detailed in sec.2 of supplementary material). Similar to our findings in Sec. 5.2, UPC expresses its superior attacking capability in the real physical world compared to natural image patterns which results in nearly zero drop rate in every posture.

As can be seen from Table 2 and Table 4, the behaviors of detectors exhibit similar trends under different physical conditions such as lighting conditions in both virtual scenes and physical environments. Another noteworthy comment is that the generated patterns from virtual scene experiments demonstrate high transferability to the real physical world (Table 4). These facts indicate that our ***AttackScenes*** is a suitable dataset to study physical attacks.

Motion Testing. To further demonstrate the efficacy of UPC, we also test our algorithm on human motions. The video clips were obtained under different physical conditions (*e.g.*, different lighting conditions, scenes) while the volunteers are walking towards the camera. Meanwhile, they are randomly changing postures from the 6 classes as mentioned above. A total of 3693 frames where 583, 377, 219, 713, 804 and 997 frames are collected under 5 different physical scenes so as to make this dataset diverse and representative. And the detection precisions are 26% (150/583), 21% (80/377), 17% (37/219), 34% (240/713), 15% (118/804) and 24% (240/997), respectively. Experiments in all physical scenes have observed low detection rates, which further confirms the effectiveness of UPC. The detection results of some sampled frames are shown in Fig. 5(b), where people are detected as "dog". We find this attack is much more effective under brighter conditions. This phenomenon coincides with previous observations in virtual scene studies (Sec. 5.2), and also further justify the potential value of ***AttackScenes***. Moreover, we find that blurred camouflage patterns during motion make UPC less effective, which lead to higher detection accuracy.

We also plot the relationship between the detection precision *vs.* angle/distance under 8-Pattern schemes as in Fig. 6. It can be concluded that when the absolute value of the angle/distance between the person and the camera becomes larger, camouflage patterns are captured with lower quality and thus hampering the attacks.

Table 4. Average precision $p_{0.5}$ and drop rate under 3 brightness conditions in stationary testing.

Network	FR-VGG16-0712		FR-RES101-0712	
Schemes	Standing	Shaking Head	Standing	Shaking Head
Original	1.0 (-)	1.0 (-)	1.0	1.0 (-)
Natural	0.98 (0.02)	0.98 (0.02)	0.98 (0.02)	1.0 (0.0)
3-Patterns	0.67 (**0.33**)	0.74 (**0.26**)	0.72 (**0.28**)	0.76 (**0.24**)
7-Patterns	0.59 (**0.41**)	0.59 (**0.41**)	0.59 (**0.41**)	0.57 (**0.43**)
8-Patterns	0.17 (**0.83**)	0.20 (**0.80**)	0.19 (**0.81**)	0.20 (**0.80**)
Schemes	Fork Waist	Leg Lifting	Fork Waist	Leg Lifting
Original	1.0 (-)	1.0 (-)	1.0 (-)	1.0 (-)
Natural	1.0 (0.0)	1.0 (0.0)	1.0 (0.0)	1.0 (0.0)
3-Patterns	0.72 (**0.28**)	0.74 (**0.26**)	0.76 (**0.24**)	0.71 (**0.29**)
7-Patterns	0.56 (**0.44**)	0.54 (**0.46**)	0.57 (**0.43**)	0.57 (**0.43**)
8-Patterns	0.20 (**0.80**)	0.26 (**0.74**)	0.24 (**0.76**)	0.30 (**0.70**)
Schemes	Rasing Hands	Sitting	Rasing Hands	Sitting
Original	1.0 (-)	1.0 (-)	1.0 (-)	1.0 (-)
Natural	0.98 (0.02)	1.0 (0.0)	1.0 (0.0)	1.0 (0.0)
3-Patterns	0.83 (**0.17**)	0.76 (**0.24**)	0.85 (**0.15**)	0.74 (**0.26**)
7-Patterns	0.65 (**0.35**)	0.54 (**0.46**)	0.69 (**0.31**)	0.59 (**0.41**)
8-Patterns	0.35 (**0.65**)	0.22 (**0.78**)	0.35 (**0.65**)	0.26 (**0.74**)

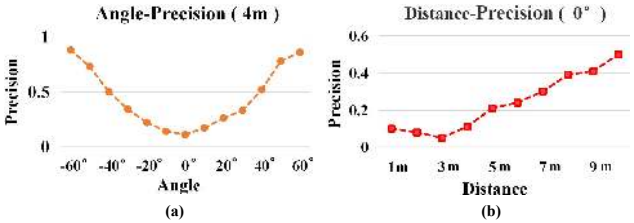


Figure 6. **The precision $p_{0.5}$ of detectors under different angle/distance conditions**. We note that high viewing angle or far distance can make attacks less effective.

5.4. Transferability Experiment

We generate camouflage patterns from one architecture to attack other models. In our experiment, FR-VGG16-0712 and FR-RES101-0712 are used to compute camouflage patterns. We introduce ResNet-50, ResNet-152 and MobileNet [9] based faster r-cnn which are trained on MS-COCO2014 [20] dataset as transfer-testing models. Other architecture models including R-FCN (ResNet-101) [5], SSD (VGG-16) [21], Yolov2 [28], Yolov3 [29] and RetinaNet [19] are considered in our transferability experiments. Eight models are publicly available, and we denote them as FR-RES50-14, FR-RES152-14, FR-

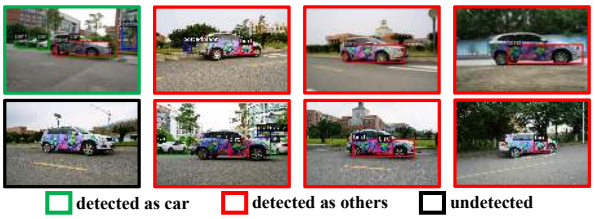


Figure 7. **The results of attacking Volvo XC60 (top row) and Volkswagen Tiguan (bottom row)**. The generated camouflage patterns fool detectors to misrecognize the car as bird.

Table 5. Average precision $p_{0.5}$ in transferability testing. First seven rows show the results of cross-training transfer testing, and rest five rows display the cross-network transfer's results (**bold** in "Network" column).

Network	Original	FR-VGG16-0712	FR-RES101-0712
		Average (Drop)	Average (Drop)
FR-VGG16-0712	0.95	**0.04 (0.91)**	0.10 (0.85)
FR-RES101-0712	0.99	0.78 (0.21)	**0.06 (0.93)**
FR-VGG16-07	0.95	0.08 (0.87)	0.11 (0.84)
FR-RES101-07	0.99	0.51 (0.48)	0.10 (0.89)
FR-RES50-14	1.0	0.85 (0.15)	0.78 (0.22)
FR-RES152-14	1.0	0.62 (0.38)	0.43 (0.57)
FR-MN-14	0.99	0.51 (0.48)	0.25 (0.74)
RFCN-RES101-07 [5]	0.98	0.64 (0.34)	0.41 (0.57)
SSD-VGG16-0712 [21]	0.75	0.13 (0.62)	0.16 (0.59)
Yolov2-14 [28]	1.0	0.59 (0.41)	0.38 (0.62)
Yolov3-14 [29]	1.0	0.69 (0.31)	0.71 (0.29)
Retina-14 [19]	1.0	0.72 (0.31)	0.49 (0.51)

MN-14, RFCN-RES101-07, SSD-VGG16-0712, Yolov2-14, Yolov3-14 and Retina-14. The confidence threshold of all models is set as 0.5 for evaluation.

The following experiments are conducted: (1) **Cross-Training Transfer.** The transferability between source and attacked models have the same architecture but are trained on different datasets (*e.g.*, using the pattern generated from FR-VGG16-0712 to attack FR-VGG16-07); (2) **Cross-Network Transfer**. The transferability through different network structures (*e.g.*, using the pattern computed from FR-VGG16-0712 to attack Yolov3-14).

For transfer experiments, virtual walking humans with 8-Patterns scheme (Fig. 4) are used to evaluate the transferability under transfer attacks. The transfer performances are illustrated in Table 5. The original pattern scheme is used to calculate the baseline precision of each model (denoted as "Original" in Table 5). We observe the precisions of all detectors have dropped, which means the generated patterns exhibits well transferability and generality across different models and datasets. It is noteworthy to mention our proposed UPC also successfully breaks 4 state-of-the-art defenses [18, 42, 7, 27] (see Supplementary).

5.5. Generalization to Other Categories

To demonstrate the generalization of UPC, we construct camouflage patterns by untargeted attacks to fool the "car" category (*i.e.*, rigid but non-planar object). We use Volvo XC60 (champagne) and Volkswagen Tiguan (white) as the attacking target in the real world. The pattern will be regarded as car paintings by human observers. In order to not affect driving, we restrict the camouflage coverage regions to exclude windows, lightings, and tires. We collect 120 photos which includes different distances ($8 \sim 12m$) and angles ($-45° \sim 45°$) in 5 different environments (Fig. 7). The video is recorded simultaneously at same angles. The performance of pure non-camouflage car is $p_{0.5} = 1$, while after attacking only 24% (29/120) images and 26% (120/453) frames are detected as "car" correctly, which verifies the efficacy of UPC.

Figure 8. **Generated camouflage patterns are semantically meaningful**. Even for unconstrained patterns, human observer can relate the generated camouflage patterns to the targeted label.

6. Discussion

Abstract Semantic Patterns. A side finding is that the generated patterns without semantic constraint (Eq. 1) can be less semantic meaningful but exhibits abstract meanings (Fig. 8). This observation suggest that human and machine classification of adversarial images are robustly related as suggested in [47].

Defense Method Evaluation. With the development of defense methods in digital domain [22, 44], we hope the collected dataset, ***AttackScenes***, can benefit future research of defense methods against physical attacks.

7. Conclusion

In this paper, we study the problem of physical attacks on object detectors. Specifically, we propose UPC to generate universal camouflage patterns which hide a category of objects from being detected or to misdetect objects as the target label by state-of-the-art object detectors. In addition, we present the first standardized benchmark dataset, ***AttackScenes***, to simulate the real 3D world in controllable and reproducible environments. This dataset can be used for accessing the performance of physical-world attacks at a fair standard. Our study shows that the learned universal camouflage patterns not only mislead object detectors in the virtual environment, *i.e.*, ***AttackScenes***, but also attack detectors successfully in the real world.

Acknowledgements

This work was supported by National Key Research and Development Plan in China (2018YFC0830500), National Natural Science Foundation of China under Grant (61972433), Fundamental Research Funds for the Central Universities (19lgjc11, 19lgyjs54).

References

[1] Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. *arXiv preprint arXiv:1707.07397*, 2017.

[2] Arjun Nitin Bhagoji, Warren He, Bo Li, and Dawn Song. Practical black-box attacks on deep neural networks using efficient query mechanisms. In *European Conference on Computer Vision*, pages 158–174. Springer, 2018.

[3] Tom B Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. Adversarial patch. *arXiv preprint arXiv:1712.09665*, 2017.

[4] Shang-Tse Chen, Cory Cornelius, Jason Martin, and Duen Horng Polo Chau. Shapeshifter: Robust physical adversarial attack on faster r-cnn object detector. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 52–68. Springer, 2018.

[5] Jifeng Dai, Li Yi, Kaiming He, and Sun Jian. R-fcn: Object detection via region-based fully convolutional networks. 2016.

[6] Ivan Evtimov, Kevin Eykholt, Earlence Fernandes, Tadayoshi Kohno, Bo Li, Atul Prakash, Amir Rahmati, and Dawn Song. Robust physical-world attacks on deep learning models. *arXiv preprint arXiv:1707.08945*, 1:1, 2017.

[7] Chuan Guo, Mayank Rana, Moustapha Cisse, and Laurens Van Der Maaten. Countering adversarial images using input transformations. *arXiv preprint arXiv:1711.00117*, 2017.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[9] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[10] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[11] Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. *arXiv preprint arXiv:1804.08598*, 2018.

[12] Steve TK Jan, Joseph Messou, Yen-Chen Lin, Jia-Bin Huang, and Gang Wang. Connecting the digital and physical world: Improving the robustness of adversarial attacks. In *The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI'19)*, 2019.

[13] Valentin Khrulkov and Ivan Oseledets. Art of singular vectors and universal adversarial perturbations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8562–8570, 2018.

[14] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2016.

[15] Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. Textbugger: Generating adversarial text against real-world applications. *arXiv preprint arXiv:1812.05271*, 2018.

[16] Jianan Li, Xiaodan Liang, ShengMei Shen, Tingfa Xu, Jiashi Feng, and Shuicheng Yan. Scale-aware fast r-cnn for pedestrian detection. *IEEE transactions on Multimedia*, 20(4):985–996, 2018.

[17] Yuezun Li, Daniel Tian, Xiao Bian, Siwei Lyu, et al. Robust adversarial perturbation on deep proposal-based models. *arXiv preprint arXiv:1809.05962*, 2018.

[18] Fangzhou Liao, Ming Liang, Yinpeng Dong, Tianyu Pang, Xiaolin Hu, and Jun Zhu. Defense against adversarial attacks using high-level representation guided denoiser. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1778–1787, 2018.

[19] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal loss for dense object detection. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[20] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[21] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng Yang Fu, and Alexander C. Berg. Ssd: Single shot multibox detector. In *European Conference on Computer Vision*, 2016.

[22] Xuanqing Liu, Yao Li, Chongruo Wu, and Cho-Jui Hsieh. Adv-bnn: Improved adversarial defense through robust bayesian neural network. *arXiv preprint arXiv:1810.01279*, 2018.

[23] Xin Liu, Huanrui Yang, Ziwei Liu, Linghao Song, Hai Li, and Yiran Chen. Dpatch: An adversarial patch attack on object detectors. *arXiv preprint arXiv:1806.02299*, 2018.

[24] Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5188–5196, 2015.

[25] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1765–1773, 2017.

[26] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2574–2582, 2016.

[27] Aaditya Prakash, Nick Moran, Solomon Garber, Antonella DiLillo, and James Storer. Deflecting adversarial attacks with pixel deflection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8571–8580, 2018.

[28] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017.

[29] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[30] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region

proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[31] Suranjana Samanta and Sameep Mehta. Towards crafting text adversarial samples. *arXiv preprint arXiv:1707.02812*, 2017.

[32] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 618–626, 2017.

[33] Mahmood Sharif, Sruti Bhagavatula, Lujo Bauer, and Michael 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*, pages 1528–1540. ACM, 2016.

[34] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[35] Chawin Sitawarin, Arjun Nitin Bhagoji, Arsalan Mosenia, Mung Chiang, and Prateek Mittal. Darts: Deceiving autonomous cars with toxic signs. *arXiv preprint arXiv:1802.06430*, 2018.

[36] Dawn Song, Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Florian Tramer, Atul Prakash, and Tadayoshi Kohno. Physical adversarial examples for object detectors. In *12th {USENIX} Workshop on Offensive Technologies ({WOOT} 18)*, 2018.

[37] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[38] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

[39] Simen Thys, Wiebe Van Ranst, and Toon Goedemé. Fooling automated surveillance cameras: adversarial patches to attack person detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[40] Zhibo Wang, Siyan Zheng, Mengkai Song, Qian Wang, Alireza Rahimpour, and Hairong Qi. advpattern: Physical-world attacks on deep person re-identification via adversarially transformable patterns. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[41] Rey Reza Wiyatno and Anqi Xu. Physical adversarial textures that fool visual object tracking. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[42] Cihang Xie, Jianyu Wang, Zhishuai Zhang, Zhou Ren, and Alan Yuille. Mitigating adversarial effects through randomization. *arXiv preprint arXiv:1711.01991*, 2017.

[43] Cihang Xie, Jianyu Wang, Zhishuai Zhang, Yuyin Zhou, Lingxi Xie, and Alan Yuille. Adversarial examples for semantic segmentation and object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1369–1378, 2017.

[44] Cihang Xie, Yuxin Wu, Laurens van der Maaten, Alan L Yuille, and Kaiming He. Feature denoising for improving adversarial robustness. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 501–509, 2019.

[45] Xiaohui Zeng, Chenxi Liu, Yu-Siang Wang, Weichao Qiu, Lingxi Xie, Yu-Wing Tai, Chi Keung Tang, and Alan L Yuille. Adversarial attacks beyond the image space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[46] Yang Zhang, Hassan Foroosh, Philip David, and Boqing Gong. Camou: Learning physical vehicle camouflages to adversarially attack detectors in the wild. 2018.

[47] Zhenglong Zhou and Chaz Firestone. Humans can decipher adversarial images. *Nature communications*, 10(1):1334, 2019.