Universal Physical Camouflage Attacks on Object Detectors

Lifeng Huang1,2 Chengying Gao1 Yuyin Zhou3 Cihang Xie3 Alan Yuille3 Changqing Zou4,5 Ning Liu1,2 *
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.1.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://mesunhf.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.
vehicle body, board [39]), which can be less effective for complex objects (articulated non-rigid or non-planar objects, e.g., human). (3) constructing meaninglessness 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 implemented in 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>
<thead>
<tr>
<th>Methods</th>
<th>Rigid</th>
<th>Non-Rigid</th>
<th>Planar</th>
<th>Non-Planar</th>
<th>Universal</th>
<th>Semantic</th>
</tr>
</thead>
<tbody>
<tr>
<td>[4]</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[36]</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>[46]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Ours</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

### 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 scenario. 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.
We propose to craft a universal pattern for faster-rcnn by simulating internal attributes of objects, deformations, 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).

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 $X$.

In summary, the perturbed images are generated by:

$$\delta^t = \text{Proj}_{I, \epsilon}(\delta^{t-1} + \Delta \delta, I, \epsilon),$$

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

Eq. 1 is the semantic constraint, where $\delta^t$ and $\Delta \delta$ denote the adversarial pattern and its updated vector at iteration $t$, respectively. $\text{Proj}_{I, \epsilon}$ projects generated pattern onto the surface of $L_\infty$ norm-balls with radius $\epsilon$ 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). \( T_r \) 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 \( \mathcal{P} = \{ p_i | p_i = (s_i, \hat{d}_i); i = 1, 2, 3...M \} \), where \( s_i \) is the confidence 
score of \( i \)-th bounding box and \( \hat{d}_i \) represents the 
coordinates of \( i \)-th bounding box. We define the objective function for attacking the RPN as following:

\[
L_{rpn} = \mathbb{E}_{p_i \sim \mathcal{P}} \left( \mathcal{L}(s_i, y^i) + s_i \| \hat{d}_i - \Delta \hat{d}_i \|_p \right),
\]

where \( y^i \) is the target score, and we set \( y^i \) for background 
and \( y^0 \) for foreground; \( \mathcal{L} \) is the Euclidean distance loss; \( \Delta \hat{d}_i \) is a pre-defined 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 \( \mathcal{P}^* \). These top-scored 
proposals \( \mathcal{P}^* \) 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:

\[
\begin{align*}
L_{cls} &= \mathbb{E}_{p_i \sim \mathcal{P}^*} C(p) y^i + \mathbb{E}_{p_i \sim \mathcal{P}^*} \mathcal{L}(C(p), y^i), \\
L_{reg} &= \sum_{p_i \sim \mathcal{P}^*} ||R(p) y^i - \Delta \hat{d}_i||_t,
\end{align*}
\]

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

---

**Algorithm 1** Algorithm of UPC

**Input:** Training images \( \mathcal{X} \); Target label \( y^i \); Balance parameters \( \lambda_1, \lambda_2 \); Iteration parameters \( iter_s \) and \( iter_{max} \); Fooling rate threshold \( r_s \).

**Output:** Universal adversarial pattern \( \delta \); Fooling rate \( r \);

```plaintext
1: \( \delta^0 \leftarrow \text{random}, \Delta \delta \leftarrow 0, r \leftarrow 0, t \leftarrow 0 \\
2: \text{while } t < iter_{max} \text{ and } r < r_s \text{ do} \\
3: \quad t \leftarrow t + 1, \delta^t \leftarrow \text{Proj}_{\lambda_s}(\delta^{t-1} + \Delta \delta, I, \epsilon) \\
4: \quad \text{for all } x_i \sim \mathcal{X} \text{ do} \\
5: \quad \quad \text{Choose the transformation of } T_r \text{ and } T_c \text{ randomly} \\
6: \quad \quad \hat{x}_i = \text{clip} (T_r(x_i + T_c(\delta^t)), 0, 1) \\
7: \quad \text{end for} \\
8: \quad \text{Calculate the fooling rate } r \text{ of perturbed images } \hat{X} \\
9: \quad \text{if } t < iter_s \text{ and } r < r_s \text{ then} \\
10: \quad \quad \argmin_{\delta^t \sim \mathcal{X}} \mathbb{E} \ L_{rpn} + L_{tv} \\
11: \quad \quad \text{else} \\
12: \quad \quad \argmin_{\delta^t \sim \mathcal{X}} \mathbb{E} \ (L_{rpn} + \lambda_1 L_{cls} + \lambda_2 L_{reg}) + L_{tv} \\
13: \quad \text{end if} \\
14: \text{end while}
```

proposals which can arye detected as true label \( y \) and \( y^i \) 
is the target label for attacking. \( \Delta \hat{d} \) 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 regres-
sor, respectively, and are referred to as C&R attack (marked 
as green in Fig. 2(a)). For untargeted attack, we set \( y = y^i \) 
for maximizing (instead of minimizing) Eq. 4.

3.5. Two-Stage Attacking Procedure

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

\[
\argmin_{\delta} \mathbb{E}_{\hat{x} \sim \hat{X}} (L_{rpn} + \lambda_1 L_{cls} + \lambda_2 L_{reg}) + L_{tv}(\delta^t),
\]

where \( \delta \) and \( \hat{X} \) 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 \( \delta \) and the perturbed images \( \hat{x} \) until the fooling rate becomes larger than a certain threshold or the attack iteration reaches the maximum. \( \delta \) 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 \( \delta \) substantially lowers the quality of proposals and thereby achieves a high fooling rate.
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:

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

where $x$ is the original instance and $\hat{x}$ denotes the instance with camouflage patterns. $\mathcal{V}, \mathcal{L}, \mathcal{S}$ 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.

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Setup</th>
<th>Standing</th>
<th>Walking</th>
<th>Sitting</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FR-VGG16</td>
<td>FR-RES101</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Original</td>
<td>0.97 0.97 0.99 0.99 (0.01)</td>
<td>0.99 0.99 0.99 0.99 (0.00)</td>
<td>0.99 0.99 1.00 1.00 (0.00)</td>
<td></td>
</tr>
<tr>
<td>Naive</td>
<td>0.97 0.97 0.99 0.97 (0.01)</td>
<td>0.99 0.99 0.99 0.99 (0.00)</td>
<td>0.99 0.99 1.00 1.00 (0.00)</td>
<td></td>
</tr>
<tr>
<td>Natural</td>
<td>0.95 0.96 0.98 0.96 (0.02)</td>
<td>0.97 0.97 0.98 0.97 (0.02)</td>
<td>0.99 0.99 1.00 1.00 (0.00)</td>
<td></td>
</tr>
<tr>
<td>3-Pattern</td>
<td>0.64 0.68 0.70 0.69 (0.09)</td>
<td>0.73 0.69 0.70 0.69 (0.09)</td>
<td>0.73 0.69 0.70 0.69 (0.09)</td>
<td></td>
</tr>
<tr>
<td>7-Pattern</td>
<td>0.55 0.31 0.22 0.37 (0.61)</td>
<td>0.51 0.48 0.64 0.54 (0.45)</td>
<td>0.51 0.48 0.64 0.54 (0.45)</td>
<td></td>
</tr>
<tr>
<td>11-Pattern</td>
<td>0.15 0.03 0.02 0.07 (0.91)</td>
<td>0.10 0.09 0.11 0.11 (0.88)</td>
<td>0.10 0.09 0.11 0.11 (0.88)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>L1 L2 L3 Avg (Drop)</td>
<td></td>
<td>L1 L2 L3 Avg (Drop)</td>
</tr>
</tbody>
</table>

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.

<table>
<thead>
<tr>
<th>Network</th>
<th>FR-VGG16-0712</th>
<th>FR-RES101-0712</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setup</td>
<td>Standing</td>
<td>Walking</td>
</tr>
<tr>
<td>$UPC_{rc}$ (ours)</td>
<td>0.07 (0.91)</td>
<td>0.04 (0.91)</td>
</tr>
<tr>
<td>$UPC_{rc}$ (ours)</td>
<td>0.66 (0.32)</td>
<td>0.33 (0.62)</td>
</tr>
<tr>
<td>$CLS_{rc}$ (ours)</td>
<td>0.18 (0.80)</td>
<td>0.06 (0.89)</td>
</tr>
<tr>
<td>$Shape$ [4]</td>
<td>0.70 (0.28)</td>
<td>0.39 (0.56)</td>
</tr>
<tr>
<td>$ERP^2$ [6]</td>
<td>0.85 (0.13)</td>
<td>0.48 (0.47)</td>
</tr>
<tr>
<td>$AdvPat$ [39]</td>
<td>0.77 (0.21)</td>
<td>0.31 (0.64)</td>
</tr>
</tbody>
</table>

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(20x3x18) 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).

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 $T_e$ are simulated in Eq. 2, denoted as $UPC_{rc}$; (2) only external physical environments are modeled, i.e., $T_e$ is used in Eq. 2, denoted as $UPC_c$. (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 $T_e$ 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 $UPC_c$, $SS$ 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 $UPC_c$ 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 cannot depress the activated features of un-occluded parts effectively, which may lead higher detection accuracy.
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 α7R 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.

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-
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).

<table>
<thead>
<tr>
<th>Network</th>
<th>Original</th>
<th>FR-VGG16-0712</th>
<th>FR-RES101-0712</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network</td>
<td>Average (Drop)</td>
<td>Average (Drop)</td>
<td></td>
</tr>
<tr>
<td>FR-VGG16-0712</td>
<td>0.95</td>
<td>0.04 (0.91)</td>
<td>0.10 (0.85)</td>
</tr>
<tr>
<td>FR-RES101-0712</td>
<td>0.99</td>
<td>0.78 (0.21)</td>
<td>0.06 (0.93)</td>
</tr>
<tr>
<td>FR-VGG16-07</td>
<td>0.95</td>
<td>0.08 (0.87)</td>
<td>0.13 (0.83)</td>
</tr>
<tr>
<td>FR-RES101-07</td>
<td>0.99</td>
<td>0.51 (0.48)</td>
<td>0.10 (0.89)</td>
</tr>
<tr>
<td>FR-RS50-14</td>
<td>1.0</td>
<td>0.85 (0.15)</td>
<td>0.78 (0.22)</td>
</tr>
<tr>
<td>FR-RES152-14</td>
<td>1.0</td>
<td>0.62 (0.38)</td>
<td>0.43 (0.57)</td>
</tr>
<tr>
<td>FR-MN-14</td>
<td>0.99</td>
<td>0.51 (0.48)</td>
<td>0.25 (0.74)</td>
</tr>
<tr>
<td>RFCN-RES101-07 [5]</td>
<td>0.98</td>
<td>0.64 (0.34)</td>
<td>0.41 (0.57)</td>
</tr>
<tr>
<td>SSD-VGG16-0712 [21]</td>
<td>0.75</td>
<td>0.13 (0.62)</td>
<td>0.16 (0.59)</td>
</tr>
<tr>
<td>Yolov2-14 [28]</td>
<td>1.0</td>
<td>0.59 (0.41)</td>
<td>0.38 (0.62)</td>
</tr>
<tr>
<td>Yolov3-14 [29]</td>
<td>1.0</td>
<td>0.69 (0.31)</td>
<td>0.71 (0.29)</td>
</tr>
<tr>
<td>Retina-14 [19]</td>
<td>1.0</td>
<td>0.72 (0.31)</td>
<td>0.49 (0.51)</td>
</tr>
</tbody>
</table>

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 performance is 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 ($-45^\circ \sim 45^\circ$) 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.

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, 19gyjs54).
References


