OptiCloak: Blinding Vision-Based Autonomous Driving Systems Through Adversarial Optical Projection

Huixiang Wen, Shan Chang, *Member, IEEE*, Luo Zhou, Wei Liu, and Hongzi Zhu, *Senior Member, IEEE*

***Abstract*—Studies have proven that applying patch stickers generated through adversarial training to target objects can effectively deceive classifiers or target detectors. These "Print-and-paste" adversarial attacks however have three shortcomings. First, touching the target object physically is required, which may be infeasible in practice. Second, stickers might be taken as evidence to identify the attackers. Third, the attack effect decreases significantly in poor light, especially at long distances. To overcome above limitations, we introduce OptiCloak, a car vanishing attack, which fools the object detector (OD) of a vision-based autonomous driving systems with transient projection pattern. We establish three digital-to-physical mapping models to compensate the distortions caused by perspective deformation, double image, and partial light reflection in real world. Furthermore, to avoid adversarial functionality degeneration caused by the loss of patch details in long-range attacks, we utilize MeanShift Filtering to constrain the "*resolution*" of pixels in a patch during training. We propose a gradient-free patch updating (GPU) approach, which utilizes ZO-AdaMM to approximate gradients and model parameters through the confidence scores of OD, making OptiCloak can work well in both the white-box and black-box scenarios. We deploy OptiCloak in real-world driving scenarios, and the extensive experimental results demonstrate that the OptiCloak achieves similar attack success rates (ASRs) as printed patches in bright environments, while significantly improving the attack performance in gloomy environments. This effect is validated across all settings, including different angles, imaging devices, and film transparency rates. In black-box settings, the average ASR can reach 71%, with a maximum attack distance of approximately 10 m.**

***Index Terms*—Autonomous driving, black-box, light projection attack, object detection, physical-world attack.**

Manuscript received 16 February 2024; revised 16 April 2024; accepted 20 May 2024. Date of publication 24 May 2024; date of current version 23 August 2024. This work was supported in part by the Fundamental Research Funds for the Central Universities under Grant 2232023Y-01; in part by the Natural Science Foundation of Shanghai under Grant 22ZR1400200; and in part by the National Natural Science Foundation of China under Grant 62202001. *(Corresponding author: Shan Chang.)*

Huixiang Wen, Shan Chang, and Luo Zhou are with the School of Computer Science and Technology, Donghua University, Shanghai 201620, China (e-mail: changshan@dhu.edu.cn).

Wei Liu is with the School of Management Science and Engineering, Anhui University of Finance and Economics, Bengbu 233030, China (e-mail: liuwei628@aufe.edu.cn).

Hongzi Zhu is with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: hongzi@cs.sjtu.edu.cn).

Digital Object Identifier 10.1109/JIOT.2024.3405006

I. Introduction

VISION sensors capture complex surrounding information with low installation and maintenance cost, high frame rate and resolution (i.e., accurate shape, rich color, and texture information), and providing key perception capabilities for the automatic driving (AD) systems [1]. Supporters of the visual-based AD (VBAD) systems believe that the AD systems could become qualified drivers only through camera, deep neural networks (DNNs), and computer hardware, just like human beings through visual information and brain, represented by Tesla's FSD Beta, Baidu's Apollo lite, etc. The essential decision making component of VBAD is the DNN-based 2-D Object Detector (OD), which is responsible for recognizing and localizing multiple objects in an image. Unfortunately, OD is error-prone to adversarial attacks [2], [3], [4], as is known to all. There have been plenty of *digital-world* adversarial attacks proposed against vision-based OD, in which meticulously crafted digital images cause misclassification like traffic signs [5], [6] and missed detection, e.g., pedestrians [7] and vehicles [8], [9]. The main flaw of those attacks is the requirement of direct access to the digital images before being processed by the OD, which is however impractical. Such in-lab digital attacks usually demonstrate strong performance, but turn out ineffective when applied to the physical world. Comparatively, the *physical-world* adversarial attacks directly manipulate the objects in the physical environment, rather than the data inside the pipeline of the VBAD system. The difficulty and key point of those physical deployable adversarial attacks is to guarantee the robustness to the complex physical environmental conditions, i.e., viewing angles and distances, lighting conditions, camera exposure, etc.

"Print and paste" has become a common physical-world adversarial attack approach against detection of target objects, as its ease of deployment and low cost. Small perturbations are printed as patches or stickers, and then affixed onto the surface of a target object (e.g., a traffic sign) or placed in the attack scene (e.g., on the roadside) [10], [11], [12], [13]. After being captured by camera, i.e., mapped into digital world, the adversarial nature of those perturbations could be well preserved to fool OD, although undergone a bunch of distortions. However, due to adversarial patches need to be deployed in advance, such attacks have several limitations: 1) physically approaching the intended target object may be infeasible under many circumstances; 2) such attacks are

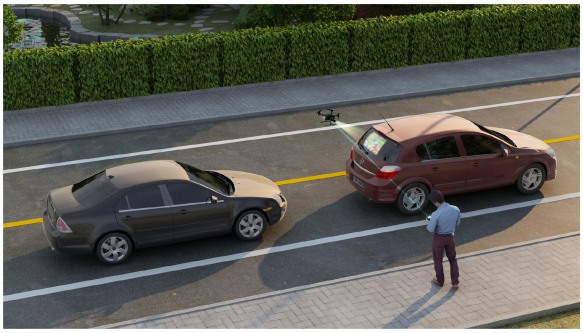

Fig. 1. Example of light projection-based attack.

hardly to be surreptitious, and the attacker might be exposed during the deployment of those permanent disturbances; 3) the deployment cannot be changed easily which makes the attack lack of flexibility. Furthermore, attack performances deteriorate significantly under poor lighting conditions.

Light projection-based attacks have been proven effective lately [14], [15], where transient light patterns projected on the target objects cause misidentifications. Comparing with the print-based attacks, the advantages of the projection-based attacks are remotely applicable and transitory without residual, thus more furtive. However, existing work targets static objects, e.g., traffic signs [5], [6], which limits the spatiotemporal context of the attacks. Furthermore, they assume that the projector is well placed to ensure high quality projections, which unfortunately can hardly be satisfied in dynamic scenarios, thus represent poor attack performance when applying to moving targets directly. Nassi et al. [16] used a projection-based *appearance attack*, named phantom attack, to prove that the Tesla's on-board OD may mistakenly identify light-projected virtual objects, e.g., people or traffic sign, as real ones.

On the contrary, in this work, we propose a *vanishing attack*, OptiCloak, where an (either white or black-box) attacker meticulously trains and projects an adversarial light pattern to the target object, i.e., moving car. Vanishing attack makes real objects disappearing in ODs, causing fatal crashes. Fig. 1 presents an example of launching a vanishing attack. An attacker manipulates an UAV equipped with a portable projector to project the adversarial light pattern onto the rear windshield of the target (i.e., front) vehicle, causing the OD of the victim (rear) vehicle fails to identify it. It is obviously more challenging comparing with phantom attack, which only faithfully project predetermined objects captured from real-world, especially under the black-box scenario. More importantly, we also emphasize that generating effective and robust projections is much more difficult than the printed ones. The following challenges hinder directly applying existing approaches to achieve satisfactory attack performance (see in Section VIII):

1) The projection distortions are caused by the two non-parallel planes, i.e., the projector lens and windshield, respectively, and thus cannot be resolved using traditional expectation over transformations (EOTs), e.g., translations or rotation.
2) On account of the double-layered design of the windshield, the projected light forms double images on the glass surface. The virtual image diminishes the adversarial functionality of the light patch.
3) According to safety regulations, such as GB7258-2017 [17], the light transmittance of windshield glass should be no less than 70%. It means only a small portion of light is reflected off the glass surface, captured by the camera, and fed into the victim's OD as input.

To solve above challenges, first, we utilize nonlinear perspective transformation rather than EOTs to model the perspective relation between the well-trained adversarial images cast from the projector, to the light patches projected on a new viewing plane, e.g., rear windshield. Second, to modeling the double image phenomenon, i.e., geometric overlap or semitransparent overlap of the same image, we generate the virtual image by multiplying the original patch with a transparency ratio. Then, the virtual image is attached to the original one with a small position offset, which is relevant to the angle of light incidence. Third, consider that the image captured by a camera is the combination of the light projection partially reflected off the windshield and the background of the projection area, i.e., objects right behind the windshield. OptiCloak models the light patch as a translucent film covering the target area instead of an opaque mask, and introduces opacity to simulate the reflectivity of the surface.

Furthermore, the main limitation of the existing attacks is short attack distance, typically within 5 m for print-based patches and light-based patches. Nevertheless, in driving scenarios, a rear vehicle should always keep a minimum distance of 10 m with its front vehicle in the same lane, even under very low speed. As the increase of attack distance, the details captured by the camera in the light patch will gradually be lost, which degrades the adversarial functionality of the patch. In other words, to guarantee the light patch can work well in long distance, it is necessary to introduce an extra "*resolution*" constraint on the during training procedure. Specifically, after each iteration of light patch training, we employ the MeanShift Filtering [18] to smooth colors within the shifting window, the size of which is determined according to the *resolution*. Finally but not the least, different from those approaches relying the transferability from a white-box model (whose architecture and hyperparameters are available to the attacker) to the target black model, we exploit Zo-AdaMM [19] to enable "real" black-box attacks. In detail, given a light patch, a same random perturbation is added to or subtracted from it, obtaining a pair of patches. Then, Zo-AdaMM approximates the stochastic gradients as well as model parameters by calculating the differential model confidences on the patch pairs, enabling gradient-free backpropagation.

We deploy OptiCloak in physical world and extensively evaluate its attack performance to state-of-the-art ODs, including one-stage (YOLO V2 [20], YOLO V3 [21], YOLO V4 [22], SSD [23]), and two-stage (Faster RCNN [24]). Experimental results demonstrate that our OptiCloak is not only robust to car appearance in terms of models and colors but also attack scenarios, e.g., backgrounds, angles, distances, illuminations, etc. OptiCloak exhibits satisfactory

attack performance, at a long attack distance, i.e., up to 10 m, even under the black-box attack scenario. Specifically, the maximum average *attack success rate (ASR)* is round 71% and 83%, for the black and white-box attackers, respectively. Furthermore, we use Grad-CAM to generate heatmaps that display the feature regions relied upon by the target model during predictions after it has been attacked.

II. Related Work

In this section, we initiate provide a broad introduction to the improvements and progress made in the field of digital-world adversarial attacks. Subsequently, we proceed to dissect the manifold manifestations of adversarial attacks within physical world.

A. Digital-World Attack

Inspired by Szegedy et al. [25], numerous approaches for adversarial attacks in digital-world are proposed by researchers. The attacker craftily implants carefully designed pixel-level perturbations into the input images of the victim model with the intent of guiding the image classifier to make incorrect decisions. Goodfellow et al. [26] proposed a gradient-based one-step method that performs only one iteration in the direction of the gradient to calculate a norm-bounded perturbation for adversarial attacks. However, they focus on the "efficiency" of perturbation computation rather than achieving high deception rates. The DeepFool attack [27] is currently recognized as an effective image-specific adversarial attack method, while ignoring the quantization aspect. Carlini and Wagner [28] proposed an attack method with transferability by calculating an additional perturbation with norm constraints, which significantly weakens the effectiveness of defensive distillation. In addition to deceiving image classifiers, Fischer et al. [29] introduced an adversarial example attack into semantic segmentation and OD. However, such pixel-level spoofing has little effect on high-resolution images. Brown et al. [30] proposed an adversarial patch attack, which breaks the limitations of pixel substitution. Expanding on this, more optimized adversarial patch attacks receive significant research attention, e.g., the method proposed by Ding et al. [7]. Nevertheless, extending these digital-world attacks into physical world may pose some challenges.

1) The attacker needs direct access to the digital image before it is processed by the OD, which is however impractical.
2) Environmental noise and natural variations have an effect on disrupt adversarial perturbations computed in digital space. For example, image blurring, noise, and JPEG compression can increase the disruption rate of the adversarial attacks.

B. Physical-World Attack

Compared to the digital-world attacks, physical attacks are more closely tied to the real-world scenarios, which may pose direct safety threats to the vehicles or individuals. Currently, the deployment methods for physical-world adversarial attacks primarily including two distinct paradigms: 1) printing and pasting adversarial patches and 2) adversarial light projection techniques.

Print and Paste Attack: Considering that the print and paste does not involve complex equipment and implementation techniques, it becomes an attractive choice for the attackers. Kurakin et al. [31] demonstrated that roughly printing adversarial examples from the digital world into the physical world can significantly reduce the toxicity of adversarial patterns. However, the factors that affect the physical environment are diverse (such as distance, angle, light intensity, etc). To address these challenges, Athalye et al. [2] developed EOTs to address a variety of physical environments, which involves constraining the expected distance between an adversarial example and the input image using a chosen distribution of transformation functions. Subsequently, Jan et al. [32] utilized conditional generative adversarial networks (cGANs) to learn the mapping process from the digital space to the physical space, thus improving the ability of adversarial example to handle noise interference. Bai et al. [12] exploited a multiscale generator and discriminator to create adversarial patches in a coarse-to-fine manner, while also optimizing their location within the physical backgrounds during the training process. Zhao et al. [11] put forward two methods, namely feature interference reinforcement (FIR) and enhanced reality-constrained generation (ERG), to enhance the robustness and adaptability of the adversarial examples in the physical world. While attackers can conveniently attach adversarial patches to the target object, these approaches may potentially reveal the identity of the attacker.

Light Projection-Based Attack: Recently, an innovative physical attack has emerged that exploits the transient nature of light to launch the real-time attacks. Hu et al. [33] projected color beams on the target objects to change the hues of captured images, leading to misclassification, e.g., recognizing "street sign" as "green light." Lovisotto et al. [34] fit a projection model to predict the projected output image, but it requires the projection-camera system to be completely stationary. Structured illumination is also used to alter the appearance of the target objects, which requires precise alignment between the projection and the target object [35]. Nguyen et al. [15] applied transformation-invariant adversarial light patterns to launch impersonation attacks. Additionally, Shen et al. [36] utilized visible and invisible light to perform stealthy attacks on the face recognition systems by projecting optimized light onto the target.

While these approaches demonstrate favorable attack efficacy, the field of light-projection attacks is still under explored in current research. Existing approaches mainly focus on the application of these attacks to objects with regular shapes, such as traffic signs or are limited to facial recognition systems with uncomplicated contextual context. Nonetheless, direct application of attack vectors in complex autonomous driving scenarios still presents significant challenges.

III. Attack Model

In this section, we provide a detailed description of the attacker's behaviors and capabilities, including attack goal, attack scenarios, and some foundational assumptions.

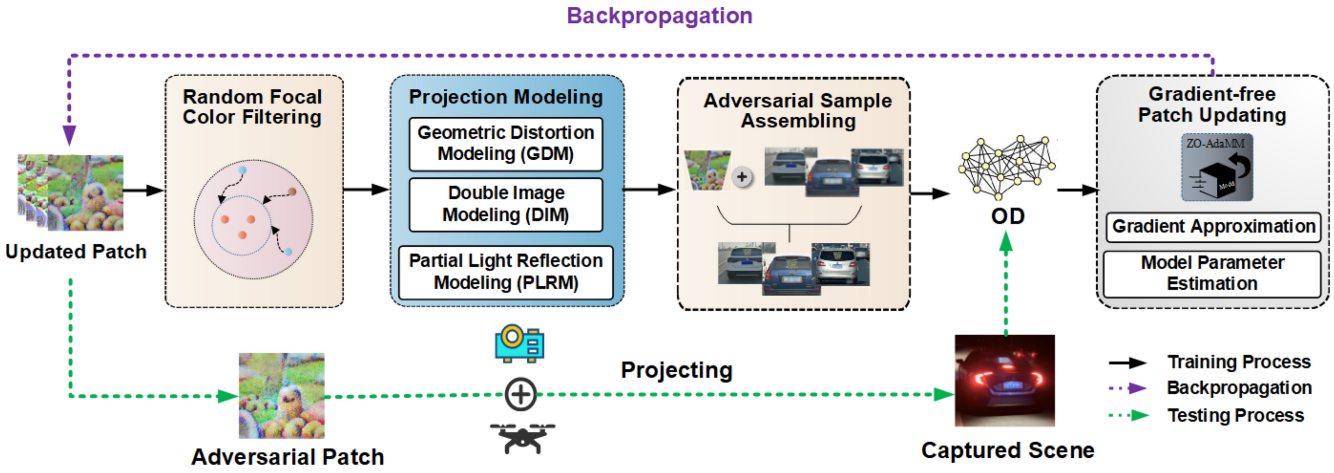

Fig. 2. Overview of OptiCloak. Initially, RFCF is applied to the updated patch after each iteration. Next, we perform digital-to-physical world PM for GDM, DIM, and PLRM. Finally, Zo-AdaMM is utilized for estimating model parameters, enabling a gradient-free patch update.

A. Attack Goal

The purpose of the attacker is to deceive the OD of the victim vehicle, making it unable to identify the vehicle being applied with the adversarial patch, i.e., target vehicle, satisfying the following characteristics.

1) No physical touch with the target vehicle.
2) No preconfiguration to the attack scene, i.e., supporting dynamic scene.
3) No evidence left over after attack.
4) Good performance under poor light condition.
5) Good performance in long distance.

B. Attack Scenarios

In our vanishing attack, an attacker can fly a drone equipped with a portable projector to launch remote attack several kilometers away. The attacker first identifies the victim vehicle through the real-time images captured by the camera on drone, and then manipulates the drone to approach the selected target vehicle and to project the light patch on it. Particularly, the target vehicle can be in front of the victim in the same lane or on the side of it in adjacent lanes. In both the cases, the light patch is projected onto the rear windshield of the target vehicle, causing the victim makes incorrect decisions, accelerating or changing lane, resulting in collision with the target vehicle.

C. Attacker Capabilities

We consider both the white-box and black-box attacks. In white-box attacks, the attacker knows the internal machine learning model of the OD, including architecture and parameters. It is a reasonable assumption as the attacker could potentially gain access to the source code of the OD system through the reverse engineering techniques. Conversely, in black-box attacks, the attacker can only access the output (i.e., confidence scores) of its victim's OD, however know nothing about it. In this case, the attacker can leverage model estimation techniques to train an adversarial patch by querying the OD output. It is also necessary to assume that the attacker can track its victim for a period of time so as to launch attack.

IV. OptiCloak Overview

OptiCloak is composed of three key modules.

1) *Random Focal Color Filtering (RFCF):* Guarantees the performance of the low-resolution light patch in long-distance attacks.
2) *Projection Modeling (PM):* Is responsible of compensating patch distortion when being applied to the physical world, as well as handling the problems of dual images and partial reflection induced from light projection.
3) *GPU:* Is applied to update the preprocessed patches by retrieving the target OD. A pipeline of OptiClock is shown in Fig. 2.

RFCF: Given a patch, either a random generated one at the beginning of the attack, or one learned from the final updating iteration) as the input. RFCF optimizes the patch toward "smoothing" the colors of pixels within a sliding window. The output of RFCF is actually a coarse grained version of the original patch, which balances its robustness to attack distance with adversarial functionality.

PM: It consists of three submodules: 1) geometric distortion modeling (GDM); 2) double image modeling (DIM); and 3) partial light reflection modeling (PLRM). GDM captures and rectifies the perspective distortion due to light projection. After GDM, the patch is fed into DIM. It models the phenomenon that the real image of a projection is superimposed with its virtual image (with a small position offset), the reason of which is the windshield is partially transparent and double layered. Then, PLRM is applied to the patch to take into account the reflectivity of glass surface by rendering the patch into semitransparent.

GPU: Those well prepared patches is attached to the designated regions (rear windshields of vehicles) in training samples which is then fed into the OD to obtain feedback confidence scores. In white-box attacks, GPU directly leverages the publicly available internal model parameters to accurately compute gradients and iteratively optimize patches

through backpropagation. In black-box attacks, GPU exploits a gradient-free backpropagation algorithm to update the adversarial patches "blindly."

V. Random Focal Color Filtering

Existing adversarial patch attacks usually suffer from poor performance at long distance. One of the key reasons is that the adversarial functionality of the patches is detail sensitive. However, in reality, the texture details of patches captured (by the in-car camera) will be lost as the attack distance (distance between the victim and the target vehicle) increases, leading to a sharp degradation of attack effectiveness. Therefore, it is desired that the adversarial functionality can be achieved as much as possible through the global structural features of patches, rather than the local texture details.

We apply MeanShift Filtering [18] to smooth out color details in patch, considering it has the following advantages: 1) it can effectively preserve edge information in patches while neutralizing the color details due to its nonlinearity and 2) it maintains global consistency of the patches, through iteratively regional color clustering.

In detail, given a patch p in digital domain, we denote the ith pixel in it as p_i. The coordinates of p_i on the 2-D spatial plane are $\mathbf{S}_{p_i} = (x_{p_i}, y_{p_i})$, and $\mathbf{C}_{p_i} = (R_{p_i}, G_{p_i}, B_{p_i})$ represents its RGB channels in color space. Then, p_i can be considered as a vector $(\mathbf{S}_{p_i}, \mathbf{C}_{p_i})$ in the feature space. For each pixel p_i in p, we conduct MeanShift Filtering, the process of which is as follows.

First, we consider p_i as the central point of an $n \times n$ sliding window, and calculate its probability density by performing kernel density estimation within the window as

$$f_h(p_i) = \frac{\mathcal{C}}{n^2 h} \sum_{j=1}^{n^2} \mathcal{K}\left(\left\| \frac{p_i - p_j}{h} \right\|^2 \right) \tag{1}$$

where $\mathcal{K}$ is the kernel function (Epanechnikov kernel), $\mathcal{C}$ is a constant. h is a vector of parameters, referred to as *resolution*, which is relevant to both the color range and spatial area (i.e., the size of the sliding window) of smoothing. A larger value of h implies a bigger spatial area of smoothing, as well as a wide range of colors mixed within the area. In other words, it focuses more on the global features, and achieves fusion of colors with relatively significant differences.

Second, we calculate the gradient of $f_h(p_i)$ by

$$\nabla f_h(p_i) = \frac{2\mathcal{C}}{n^2 h^2} \sum_{j=1}^{n^2} (p_i - p_j) \mathcal{K}'\left(\left\| \frac{p_i - p_j}{h} \right\|^2 \right) \tag{2}$$

where $\mathcal{K}'$ is the derivative of $\mathcal{K}$. Then, we move the central point in the direction of the gradient ascent for one step (e.g., a pixel), and update the current sliding window accordingly.

We repeat the above two steps using the new central point iteratively, until the convergence condition is met, i.e., reaching a central point with zero gradient. Finally we use the color of the final central pixel to replace the color of p_i, thereby completing the color shifting of p_i.

VI. Projection Modeling

Let a be a clean image containing a target object t without attack. $\mathcal{S}_a^{(t)}$ denotes the confidence score on t obtained from performing OD on a. To launching vanishing attack against t, p should first be passed the projector, reflected by the target t and then caught by the camera. We model the above procedure as $\mathcal{W}(\cdot)$, which outputs the patch $\tilde{p}$ attached on t in the adversarial image a_p under the attack. Thus, $\tilde{p}$ can be represented by

$$\tilde{p} = \mathcal{W}(p, a, z) \tag{3}$$

where z denotes the position at which p is projected. $\mathcal{W}(\cdot)$ is actually a series of transformations, including GDM, DIM, and PLRM, being executed in sequence.

A. Geometric Distortion Modeling

We designate the rear windshield of a car as the intended projection surface for the patch p_i and approximate this surface as a viewing plane. To obtain the new coordinates (x'_{p_i}, y'_{p_i}) after mapping p_i to the projection plane, we model geometric distortion using the perspective transformation, which can be described using a 3×3 homography matrix $\mathbb{H}_{3\times3}$ as

$$\begin{aligned} \phi \cdot \begin{bmatrix} x'_{p_i} \\ y'_{p_i} \\ 1 \end{bmatrix} &= \mathbb{H}_{3\times3} \cdot \begin{bmatrix} x_{p_i} \\ y_{p_i} \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & 1 \end{bmatrix} \cdot \begin{bmatrix} x_{p_i} \\ y_{p_i} \\ 1 \end{bmatrix}. \end{aligned} \tag{4}$$

Then, we reformulate the above transformation to obtain

$$\begin{bmatrix} x'_{p_i} \\ y'_{p_i} \end{bmatrix} = \frac{1}{\phi} \cdot \begin{bmatrix} m_{11}x_{p_i} + m_{12}y_{p_i} + m_{13} \\ m_{21}x_{p_i} + m_{22}y_{p_i} + m_{23} \end{bmatrix} \stackrel{\text{def}}{=} \mathcal{D}_{\mathbb{H}_{3\times3}}(p_i) \tag{5}$$

where ϕ is equal to $m_{31} \cdot x_{p^{(i)}} + m_{32} \cdot y_{p^{(i)}} + 1$.

The $\mathbb{H}_{3\times3}$ matrix reconstructs the shape of p through the coordinate system transformations, resulting in the transformed patch closely approximating the actual geometry. By providing the four vertices of the projected quadrilateral, $\mathcal{D}$ can be determined. It enables performing various geometric transformations on that quadrilateral (including translation, rotation, and affine transformations).

B. Double Image Modeling

When the projected light is incident on the windshield, it undergoes twice reflections on the front and rear surfaces of the double-layered glass. As the light reflected from the rear surface passes through the front surface again, a phase shift occurs, causing *double image*.

According to Fig. 3, a beam of light propagates from the projector to point $A(A')$, a portion of the light is reflected back and captured by the camera, following the propagation path projector $\rightarrow$ $A(A')$ $\rightarrow$ camera. The remainder of the light passes through the glass interlayer, reaches the rear surface at point $B(B')$, then undergoes secondary reflection and exits at point C on the front surface, and captured by the camera. Accordingly, the light propagation path is projector $\rightarrow$ $A(A') \rightarrow B(B') \rightarrow C(C')$ $\rightarrow$ camera. As a result, the projected patch captured by the camera forms two images on the front

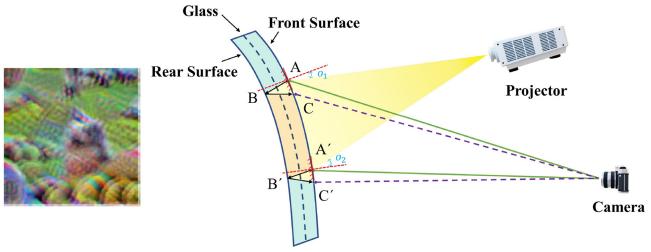

Fig. 3. Example of a double image patch and the light propagation path analysis of double imaging.

glass surface, specifically at points A and C. Furthermore, as the angle of incidence increases, the distance between the two points gradually enlarges, i.e., $\|AC\|_2 < \|A'C'\|_2$. Therefore, the phenomenon of double imaging becomes more obvious.

Conventional image deblurring techniques, such as those noise-based approaches that leverage the Gaussian or Poisson noise to accentuate the high-frequency characteristics of an image in order to enhance its visual clarity, cannot effectively deal with double image. Therefore, we introduce a function that transforms a digital patch p into its double image version, which is denoted as p^*. Specifically, we translate p along the horizontal and vertical axis of its coordinate system, each by the Δx and Δy pixels, respectively, and obtain a new image patch denoted as $p_{(\Delta x,\Delta y)}$. Then, we calculate p^* by performing the following operation:

$$p^* = \underbrace{\omega \cdot p_{(\Delta x,\Delta y)} + p}_{\mathcal{V}(p,\omega,\Delta x,\Delta y)} \tag{6}$$

where ω represents the transparency ratio, with a range of (0,1).

C. Partial Light Reflection Modeling

When an adversarial patch is projected on the windshield, the majority of the visible light passes through its surface with only a small fraction being reflected. Meanwhile, the light reflected by the object in the scene is not obstructed by the projection and is still reflected back. Consequently, the camera records reflected light from both the projected area and the background area simultaneously. In our attack, the adversarial patch acts as a semitransparent film that obscures certain parts of the image. Essentially, each pixel captured by the camera represents a mixture of the projected color and background color, with a specific mixing ratio being utilized.

It is assumed that $\beta \in (0, 1)$ represents the visible light reflectance of the rear window, and $d^{(i)}$ represents the pixel on the image that is being masked by $p^{(i)}$. Then, we can calculate the color of $p^{(i)}$ captured by the camera using the following equation:

$$\begin{bmatrix} R^*_{p_i} \\ G^*_{p_i} \\ B^*_{p_i} \end{bmatrix} = \beta \cdot \begin{bmatrix} R_{p_i} \\ G_{p_i} \\ B_{p_i} \end{bmatrix} + (1-\beta) \cdot \begin{bmatrix} R_{d_i} \\ G_{d_i} \\ B_{d_i} \end{bmatrix} \stackrel{\text{def}}{=} \mathcal{P}(p_i, d_i, \beta). \tag{7}$$

In this phase, we can rewrite $\mathcal{W}(\cdot)$ as follows:

$$\mathcal{W}(p, a, z) = \mathcal{P}\big(\mathcal{V}(\mathcal{D}_{\mathbb{H}_{3\times3}}(p), \omega, \Delta x, \Delta y), z, \beta\big). \tag{8}$$

VII. Gradient-Free Patch Updating

The fundamental requirement of launching vanishing attack successfully is to find a patch p in digital world so as to minimize the loss of OD on the target objects, denoted by $\mathcal{L}_{\text{obj}}$, which is equivalent to minimize the average confidence score of OD on a set of the targets $\mathcal{T}$. Particularly, each $t_k \in \mathcal{T}$ is projected by p. Then, minimizing $\mathcal{L}_{\text{obj}}$ can be written by

$$\underset{p}{\text{argmin}}\mathbb{E}\Big(\mathcal{S}_a^{(\mathcal{T})}(\mathcal{W}(p, a, z))\Big) = \underset{p}{\text{argmin}}\mathbb{E}\mathcal{S}_a^{(\mathcal{T})}(\tilde{p}) \tag{9}$$

where $\mathcal{S}_a(\tilde{p})$ refers to the confidence score when a is polluted by $\tilde{p}$.

Furthermore, we also introduce a smoothing loss $\mathcal{L}_{smh}$. It imposes constraint on color continuity of the patches [37], to avoid dramatic color change between the neighboring pixels since it is hardly to be sensed by the cameras precisely.

Specifically, suppose $p_{x,y}$ indicates the pixel located at (x, y), then minimizing $\mathcal{L}_{smh}$ is defined as

$$\underset{p}{\text{argmin}} \sum_{x,y} \sqrt{\left(\mathbf{C}_{p_{x,y}} - \mathbf{C}_{p_{x+1,y}}\right)^2 + \left(\mathbf{C}_{p_{x,y}} - \mathbf{C}_{p_{x,y+1}}\right)^2}. \tag{10}$$

Ultimately, p should be optimized toward minimizing the total loss $\mathcal{L}_{\text{tot}}$, which is the sum of $\mathcal{L}_{\text{obj}}$ and $\mathcal{L}_{smh}$, and we use λ to balance the two losses, i.e.,

$$\mathcal{L}_{\text{tot}} = \mathcal{L}_{\text{obj}} + \lambda \mathcal{L}_{smh}. \tag{11}$$

A. White-Box Patch Updating

In attack pipeline of OptiCloak, an attacker first generates a random digital patch p^0, and then applies $\mathcal{W}(\cdot)$ to p^0 to obtain $\tilde{p}^0$. A white-box attacker possesses complete knowledge about the target OD, thus is able to make fully use of it. The attacker employs a standard attack optimization algorithm, i.e., projected gradient descent (PGD) with l_∞ [38], to update $\tilde{p}^0$.

More specifically, suppose the current version of the patch is $\tilde{p}^n$, and correspondingly the target OD's total loss on $\mathcal{T}$ is $\mathcal{L}_{\text{tot}}^{(\mathcal{T})}$. The attacker first conducts PGD on the OD to obtain $\tilde{p}^{n+1}$. The procedure can be expressed as

$$\tilde{p}^{n+1} = \mathcal{PGD}^{(OD)}(\mathcal{L}_{\text{tot}},\ \tilde{p}^n). \tag{12}$$

Then, it computes the updated p^{n+1} by using the previously computed inverse of $\mathcal{W}$, denoted as $\mathcal{W}^{-1}$, i.e.,

$$p^{(n+1)} = \mathcal{W}^{-1}(\tilde{p}^{n+1}). \tag{13}$$

The above updating procedure is conducted iteratively, until convergence.

B. Black-Box Patch Updating

Under the black-box attacks, the target OD is inaccessible, making gradient-based optimization approaches unusable. We exploit ZO-AdaMM [19], a zeroth-order optimization algorithm, where the optimizer can directly update model parameters of Adam without relying on the knowledge on model structures or gradients of the target OD. ZO-AdaMM enables gradient-free approximation of $\tilde{p}^{n+1}$.

Specifically, the attacker first initializes Adam, and randomly selects a set of small perturbations, denoted as Υ. For each $\epsilon_i \in \Upsilon$, it estimates the gradient $g_{\epsilon_i}^{(\mathcal{T})}$ by calculating

$$g_{\epsilon_i}^{(\mathcal{T})} \approx \frac{\mathcal{S}_a^{(\mathcal{T})}(\mathcal{W}(p^n + \epsilon_i, a, z)) - \mathcal{S}_a^{(\mathcal{T})}(\mathcal{W}(p^n - \epsilon_i, a, z))}{2\epsilon_i}. \quad (14)$$

Then, adjust the learning rate γ_{ϵ_i} according to $g_{\epsilon_i}^{(\mathcal{T})}$ as well as predetermined hyperparameters, and update Adam using γ_{ϵ_i} and $g_{\epsilon_i}^{(\mathcal{T})}$ under a momentum-based update rule.

After being updated by using all the perturbations in Υ, the $\mathcal{L}_{\text{tot}}$ is back propagated to calculate $\tilde{p}^{n+1}$.

VIII. Evaluation

We first collect a data set of driving scenarios that satisfy the training requirements. Then, OptiCloak is performed on different ODs in relevant physical scenarios. In order to evaluate the robustness and transferability of OptiCloak, various physical environmental factors (e.g., lighting conditions, distances, angles, and appearances) are considered for testing. This section develops a more comprehensive understanding of the limitations and vulnerabilities of ODs under a variety of real-world scenarios.

A. Experimental Setup

We randomly select 2500 HD images from the collected videos of real driving scenes. These images are captured with a horizontal field of view (HFOV) of 45° and a vertical field of view (VFOV) of 37°, at locations, such as highways, campus loops, and parking lot entrances or exits. Since, we only focus on cars in the images, we filter the original data set and keep 1350 images with the vehicle height greater than 120 pixels as the training set. During our experimental, h is empirically set based on the experiential insights garnered through the preliminary testing and analysis. Specifically, the physical space radius and the color space radius are set to 40 and 50, respectively.

Then, we select different ODs as the target models, including FasterRCNN, SSD, YOLO V2, YOLO V3, and YOLO V4. In particular, YOLO V4 has introduced several novel techniques, such as SPP Block, CSP Bottleneck, and YOLO V4 Neck, which enable higher detection accuracy and faster detection speed. In addition, to compare the effectiveness of printed patches and projected patches in different testing environments, we employ ASRs as the metric, which is the ratio of video frames containing object scores below 0.5,[1] to the overall number of video frames.

For security reasons, we conduct the physical attack experiments in campus parking lots and campus roads. The experimental deployment of the physical attacks depicted in Fig. 4, the left image shows a printed patch attached to the rear windshield of the target car, while the right image depicts a drone (DJI Air 2) with a portable projector (Lenovo T6X) projecting the adversarial patch onto the target area.

[1]The detector typically generates many predictions, where any prediction exhibiting a probability lower than the established threshold of 0.5 undergoes a process of filtration to eliminate it from the final selection pool.

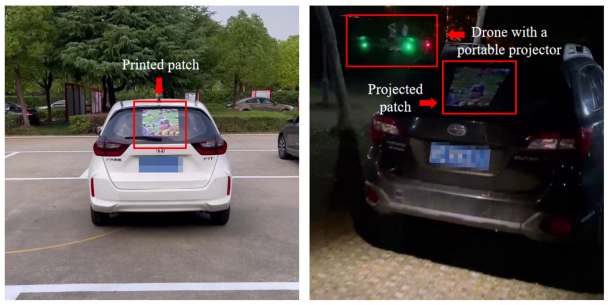

Fig. 4. Physical experiment deployment.

B. Determining Critical Parameter λ

The parameter λ of smoothing loss can significantly affect the quality and stability of the generated patches. If λ is set too small, it may result in jagged or other discontinuous patterns in the generated patches, which reduce their quality and reliability. If λ is set too large, it may cause the generated patches to be overly blurred, resulting in the loss of crucial features. Therefore, we make appropriate adjustments to λ with a step size of 0.5, i.e., $\lambda = (1.0, 1.5, \ldots, 3.0)$, and compare the object scores under different parameters during the training process, as shown in Fig. 5.

In white-box attack, when $\lambda = 2.5$ the model achieves an object score of 0.5 after 180 iterations due to the use of *RFCF*, which partially alleviates the issue of patch discontinuity. When $\lambda = 2.5$ in black-box, only 160 iterations is required to reduce the object score to 0.5. This is achieved by utilizing ZO-AdaMM to reduce noise estimation errors and combining it with the momentum mechanism of Adam, thus improving training efficiency and convergence speed. Therefore, in subsequent experiments, we select $\lambda = 2.5$ as the optimal choice.

C. Overall Performance

In order to evaluate the effectiveness of OptiCloak in physical scenarios, we conduct a series of experiments that cover dynamic driving scenes on designated roads at night under both white-box and black-box settings. Furthermore, we provide a visual representation to graphically illustrate the reasons behind the successful attack.

1) Attack on Consecutive Frames: Evaluation Procedure: specifically, we conduct three dynamic experiments to investigate the impact of attacks on YOLO V4 in a nighttime driving scenario. The experiments include a without-attack scenario, as well as the black-box and white-box attack scenarios. In without-attack experiment, we use a target car traveling at a constant speed of 10 km/h and record video clips, which are fed into OD to obtain the object score. In both the white-box and black-box attacks, the target car is configured identically to the scenario without-attack. Additionally, we employ an UAV to track the target car at a constant speed and maintain a consistent projected distance. Subsequently, we project the adversarial patches (trained using the white-box or black-box methods) onto the target area. The victim vehicle drives at any angle or speed closely behind the target car, and feeds back the captured video frames to OD. To facilitate comparative

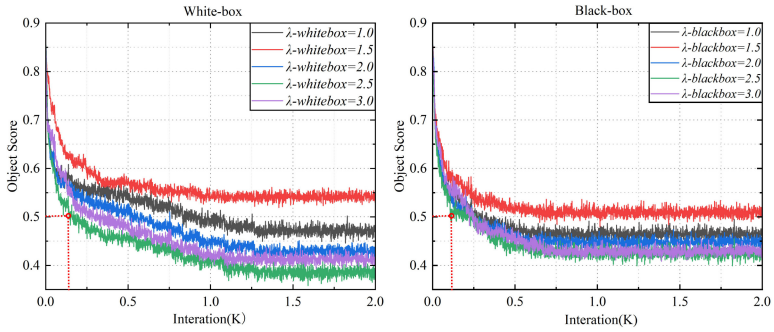

Fig. 5. Optimal parameter λ selection under white-box and black-box configurations.

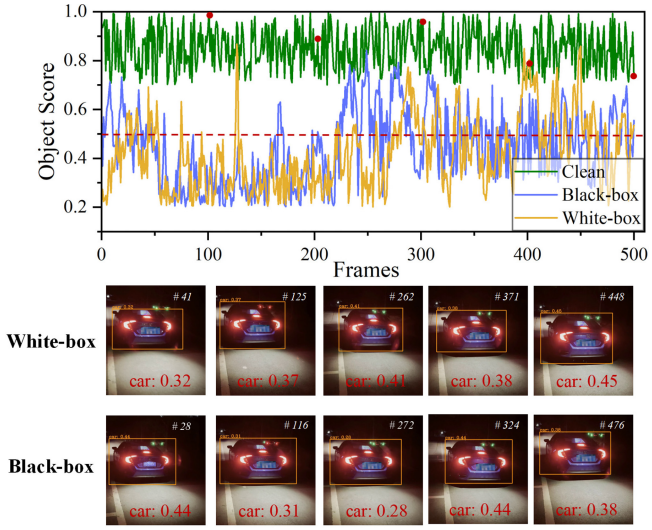

Fig. 6. Overall performance of OptiCloak in both black-box and white-box settings.

analysis, we select 500 consecutive frames of videos from each experiment as the test samples.

Results: in Fig. 6, we conduct a comparison and analysis of three experimental groups. Among them, "Clean" refers to the scenarios without any attacks, with an average object score of 0.87 and a maximum score of 1, indicating that YOLO V4 has sufficient ability to identify vehicles. The average object score is observed to be 0.38 in black-box attack. In a total number of 500 frames, a noteworthy observation is that 355 frames reflect object scores below the threshold of 0.5. Furthermore, it is worth noting that the maximum number of consecutive successful attack frames reach 126. A higher count of consecutive frames displaying a successful attack indicates that the attacker is able to exert a persistent influence on the outcome of the model's judgment. By comparison, in white-box attacks, the average object score is 0.32, and there are 83% of object scores below the threshold of 0.5. We randomly select one frame out of every 100 as a display in Fig. 6. In black-box attack scenario, the selected frames and their scores are "#28-0.44, #116-0.31, #272-0.28, #324-0.44, and #476-0.38." Similarly, in white-box scenario, the chosen frames and their scores are "#41-0.32, #125-0.37, #222-0.41, #371-0.38, and #448-0.45." These observations demonstrate that the OptiCloak poses a significant threat to the model's ability to accurately detect and identify objects in both the black-box and white-box scenarios.

2) Visualization Interpretation: To understand the decision-making process of the model, we employ Grad-CAM (gradient-weighted class activation mapping) [39] to provide visual insights into the neural network. This approach helps us identify which particular regions the model is prioritizing during the object detection.

The process begins by loading a pretrained YOLO V4 model and selecting the specific class of interest (i.e., cars). Next, we read in the images that we want to predict and visualize. For each target object, the predicted class and probability values are obtained, and the corresponding feature maps are extracted from the models. These feature maps, along with the predicted probability values, are then used as input to Grad-CAM to generate heatmaps for the associated vehicles. Finally, the heatmaps are overlaid on the original images to emphasize the regions and features that the model is focusing on. In other words, when an image is fed to a DNN for detection, Grad-CAM is able to determine which pixels are most important for the network to make a specific classification decision by computing gradients and weights and propagating them back to the input image. These salient pixels can be visualized as heatmaps, allowing us to intuitively understand the basis of the model's classification decisions.

In Fig. 7, it is not difficult to observe that images exposed to adversarial attacks may cause Grad-CAM to neglect some critical feature regions or focus its attention on irrelevant areas. This phenomenon arises due to the fact that adversarial patch attacks can disrupt the decision making capabilities of the model, leading to Grad-CAM's inability to reliably identify regions of interest.

D. Comprehensive Effectiveness

To check the effectiveness of OptiCloak (Generated by YOLO V4) in the physical world, we conduct a detailed analysis of the geometric factors (e.g., distance and angle) and firmware factors (e.g., glass film and camera equipment) under different lighting conditions.

1) Geometric Factors (Evaluation Procedure): We investigate the impact of varying distances and angles on the attacks. The iPhone 12 pro is utilized to capture videos of print-based attacks and OptiCloak in both the white-box and black-box settings. The attack distance, defined as the

(a) (b) (c)

Fig. 7. Visualization interpretation based on gradients, where (a), (b), and (c) represent without attack, white-box attack, and black-box attack, respectively.

distance between the victim and the target car, is divided into five regions ranging from 2 to 10 m, with 2 m intervals. Recordings are taken from four angles within each region: 0°, 30°, 45°, and 60°. In addition, we conduct experiments under varying lighting conditions, such as the entrance and exit of an underground parking lot with bright ambient and outdoor roads with gloomy ambient. Fig. 8 illustrates the ASRs of printed-based attacks and OptiCloak under bright and gloomy lighting conditions in both black-box and white-box settings, with corresponding regions marked accordingly. The depth of the background color is used to indicate varying degrees of ASRs, where a darker background indicates higher ASR and a lighter background indicates lower ASR.

Results: In Fig. 8(a)–(f) display the ASRs of printed-based attacks and OptiCloak in bright environment. In general, ASR decreases with increasing distance. For example, in subfigure (a), the average ASR at a distance of 8–10 m is 0.38, which is significantly lower than the 77% ASR at a distance of 4–6 m. When the victim is directly behind the target car (i.e., at 0° angle), ASR is slightly higher than the wide-angle attack. For instance, in subfigure (f), the full-range average ASR at 0° angle is 63%, while the average ASR in the other three angle ranges are 60%, 55%, and 56%, respectively. It is worth noting that the performance of OptiCloak is similar to the print-based attack in bright environments. We believe that better attack performance can be achieved with higher power projectors.

In Fig. 8(c)–(h) display the ASRs of printed-based attacks and OptiCloak in gloomy environment. It is clear that the OptiCloak is more effective than the printed-based attack. At distances between 6–10 m, printed-based attacks become almost ineffective, while OptiCloak can still achieve ASRs of 44% and 33% in both white-box and black-box settings at a distance of 10 m. This can be attributed to the fact that in gloomy environments, where the interference light is weaker, OptiCloak can generate high contrast and bright images. In contrast, printed-based attacks can be affected by factors, such as poor lighting and paper quality, resulting in distorted colors and details in the image. Overall, OptiCloak has higher environmental adaptability compared to the printed-based attack.

2) Firmware Factors (Evaluation Procedure): We further explore the impact of firmware on attacks, considering that hardware devices have different resolutions. Therefore, we use four different devices to capture videos, including the high-density pixel iPhone 12 pro and iPhone XR, as well as the low-end devices, such as Oneplus 6 and Xiaomi 8, to ensure inclusiveness in the tests. Furthermore, in order to evaluate the impact of glass film on attacks, we employ two colors (black and green) with three different transmittance rates, i.e., black (8%, 15%, and 30%), green (8%, 15%, and 30%). Finally, we choose five different types and colors of cars, including SUVs and sedans.

Results: We perform the above experiments in a black-box setting. As shown in the top of Fig. 9, the performance of attacks in bright environment gradually decreases with an increase in the light transmittance of the glass film. For example, when testing with an iPhone XR on a black glass film, the average ASRs at three different levels of light transmittance are 87%, 74%, and 38%, respectively. This can be attributed to the fact that as its light transmittance increases, more light can pass through the glass film and into the vehicle, resulting in a corresponding decrease in the amount of light being reflected back. Moreover, we observe that devices in bright environments have a negligible effect on the attack. Specifically, at a light transmittance level of 15% for the green glass film, the average ASRs using the four different devices are 82%, 84%, 84%, and 85%, respectively. This is mainly because the capture device can enhance the image brightness through its own exposure control. Another insight is that different car types and film colors have no effect on the attack.

As shown in the bottom of Fig. 9. The average ASR decreases significantly slower at different light transmittances of the glass films in the gloomy environment compared to the experiments performed in the bright environment. For example, ASRs of Oneplus 6 in the experiments with the black glass film are 80%, 77%, and 71%, respectively. In low-light conditions, the interference from ambient light is reduced and the difference in the absorption capacity between the glass films with different transmittance becomes less significant. In addition, the ASRs of Oneplus 6 may be slightly reduced compared to those of iPhone 12 pro in gloomy environment, as the device requires high ISO and long exposure times to capture images, and using a lower-end device may result in image noise or blur. Overall, the light transmittances of the glass films have some influence on the ASR in bright environment. However, such effect is significantly reduced in gloomy environment.

E. Performance of Attack Transferability

Given that the attackers typically seek to apply successful attack methods to different ODs that may operate within the varying hardware and software environments, evaluating the transferability of attacks can also enhance the generalization capability of adversarial patches.

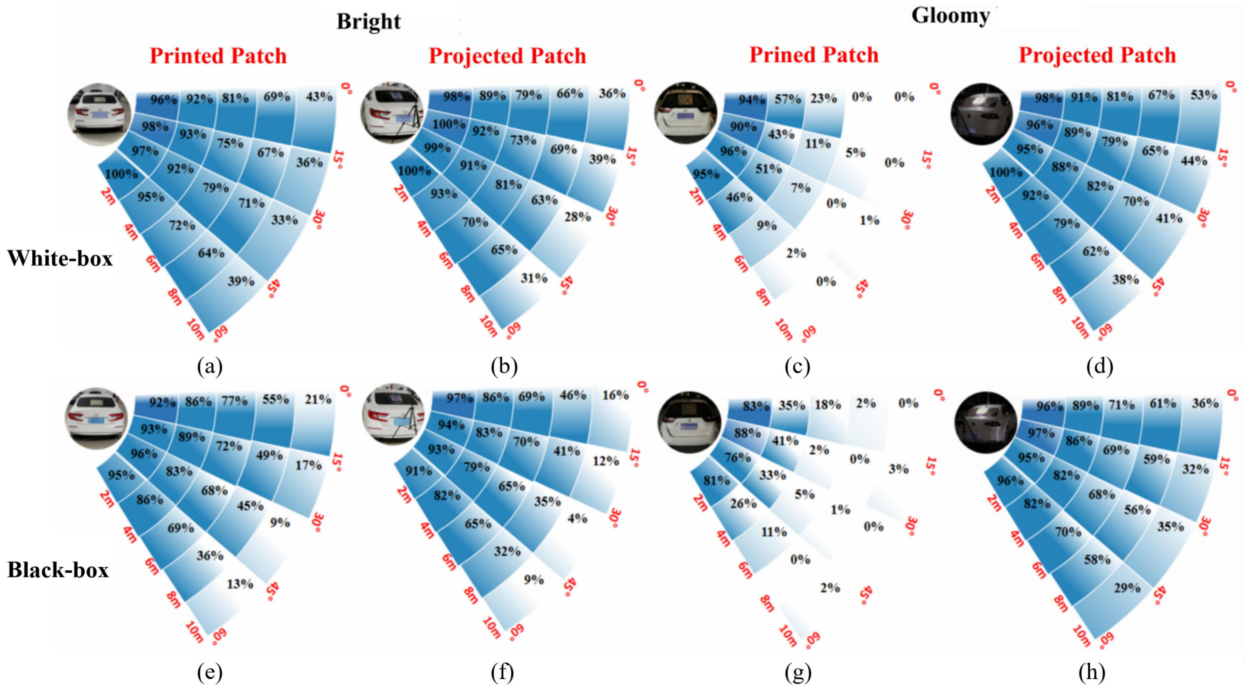

Fig. 8. ASRs of print-based and projection-based (OptiCloak) attacks are evaluated under different distances, angles, and illumination conditions in both white-box and black-box settings. Specifically, tests (a), (b), (e) and (f) are conducted in bright environments, while tests (c), (d), (g) and (h) are conducted in gloomy environments.

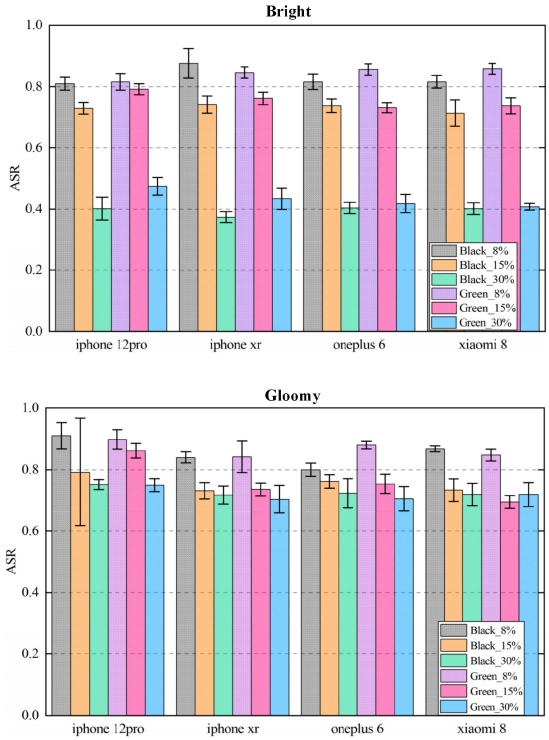

Fig. 9. ASRs of OptiCloak in bright and gloomy environments using different devices and glass films.

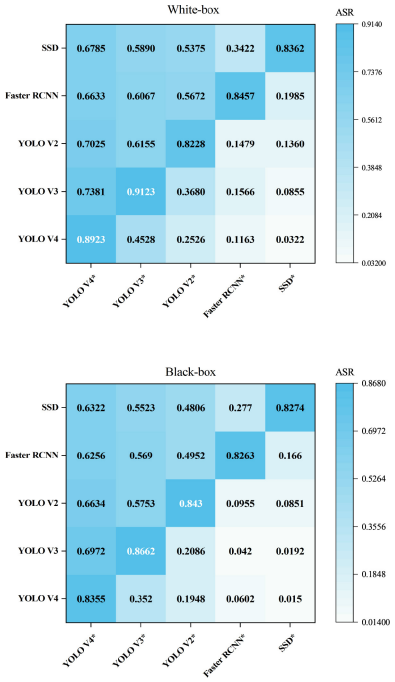

Fig. 10. Verification of attack transferability.

Evaluation Procedure: In white-box attacks, we test five different ODs. For each experiment, we select one of the models as the white-box model and employ it to train an adversarial patch. Subsequently, we launch a projection-based attack in a real-world scenario and capture a video clip. Such video is then fed back to the four other black-box models to evaluate the transferability of the adversarial patch attacks. For example, we train an adversarial patch on YOLO V4 and test the ASRs on Faster RCNN, SSD, YOLO V2, and YOLO V3, respectively. In black-box attacks, the experimental setup is similar to that of the white-box attacks, but the difference is that we cannot access the internal structure and parameters of the model during the patch training process.

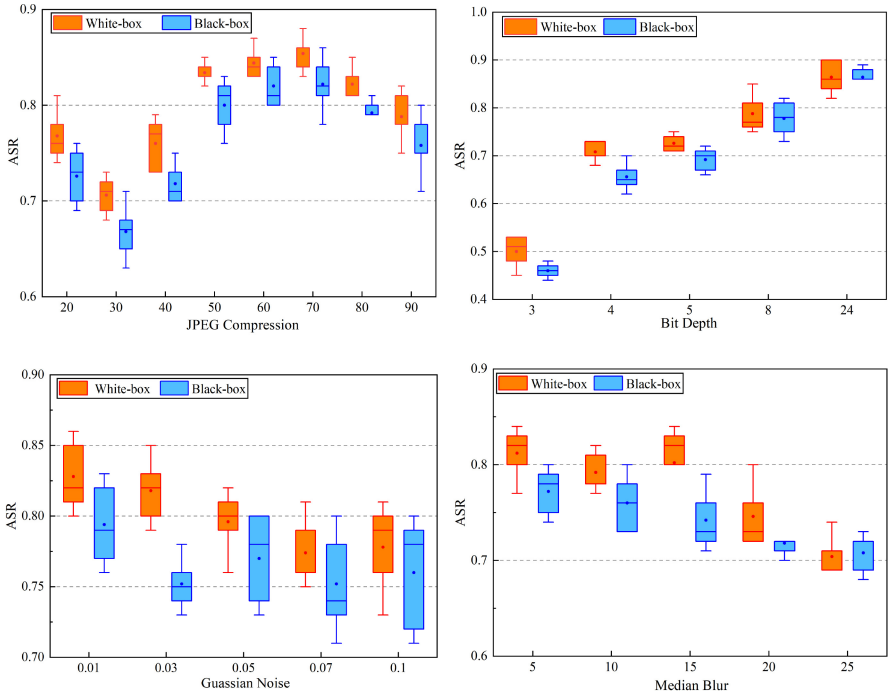

Fig. 11. Four directly applicable patch interference techniques.

Results: In Fig. 10, the horizontal axis shows different adversarial training models, while the vertical axis represents different validation models. The fill values in each cell represent the ASR obtained using the corresponding version of the patch and validation model. In white-box attacks, patches trained on the YOLO series can successfully attack Faster RCNN and SSD, while patches trained on Faster RCNN and SSD have poor transferability. For example, when employing the patch trained on YOLO V4 to attack Faster RCNN and SSD yields ASRs of 66% and 68%, respectively. Conversely, when using the patch trained on Faster RCNN or SSD to attack YOLO V4 results in ASRs of only 12% and 3%. This is because the YOLO series uses a global convolutional layer for feature extraction and outputs a dense coordinate grid, while Faster RCNN and SSD output sparse detection boxes, allowing adversarial patches trained on YOLO to successfully attack them. Moreover, we find that the patches trained on the strong models exhibit higher transferability than those trained on the weak models, as the strong models typically have higher representational capacity and contain more semantic and structural information. Similar conclusions are drawn in black-box attacks, although the overall ASRs are slightly lower than those in white-box attacks.

F. Performance of Attack Robustness

It is important to consider that certain complex physical environments can induce variability in the light projection pattern captured by the camera. If a camera is operated at high ISO or with long exposure times, its sensor may produce undesirable noise. To mitigate this noise, various noise reduction algorithms are used for camera systems. However, these algorithms can also compromise image details, leading to information loss and bit depth reduction. Moreover, continuous operation of the camera in a high-temperature environment can generate thermal noise, which may diminish the quality of the captured images.

Evaluation Procedure: We employ four widely used image processing techniques (i.e., JPEG compression, bit depth reduction, Gaussian noise, and median blur) to simulate the interference of complex environments on the patches. The quality factor of JPEG compression is used to control the degree of compression. We set the factor at different values ranging from 20 to 90, where lower values correspond to lower image quality. Meanwhile, the bit depth parameter is set to 3, 4, 5, 8, and 24 of images, with higher bit depths providing more gray-scale or color depth and leading to a more intricate and authentic representation of the image. Additionally, the Gaussian Noise is added to the patch images using various variance parameters (0.01, 0.03, 0.05, 0.07, and 0.1), where higher variance parameters lead to stronger noise intensity and produce more blurred and distorted images. Finally, median blur is employed with convolutional kernel sizes of 5, 10, 15, 20, and 25. Larger kernel sizes generate more pronounced blur effects during the image convolution. We project differently processed patches onto the five target vehicles, capture related videos, and analyse the ASRs of the patches with varying qualities.

Results: The JPEG compression experiment shown in Fig. 11, it is observed that there is no linear relationship between ASR and the quality factor. Specifically, the lowest ASR occurs at a quality factor of 30, while the highest is achieved at a quality factor of 70. For quality factors of 20 and 90, the ASR is similar. This phenomenon can be attributed to the fact that when the quality factor is low, the compression algorithm prioritizes compression ratio over image quality.

TABLE I
ABLATION STUDY OF PROJECTED PATCHES WITH DIFFERENT KINDS OF CONSIDERING

Transformation	Car1	Car2	Car3	Car4	Car5	Average
Original	24.31%	19.00%	25.86%	19.72%	23.45%	22.47%
$\mathcal{D}(\cdot)$	26.21%	25.34%	24.31%	27.59%	29.03%	26.50%
$\mathcal{V}(\cdot)$	43.72%	46.21%	45.52%	41.76%	42.24%	43.89%
$\mathcal{P}(\cdot)$	24.14%	28.83%	25.72%	24.83%	26.90%	26.08%
$\mathcal{V}(\mathcal{D}(\cdot))$	58.45%	58.79%	62.41%	61.03%	59.48%	60.03%
$\mathcal{P}(\mathcal{V}(\cdot))$	30.86%	32.07%	30.00%	32.41%	35.17%	32.10%
$\mathcal{P}(\mathcal{D}(\cdot))$	32.59%	33.72%	33.45%	32.24%	31.90%	32.78%
$\mathcal{P}(\mathcal{V}(\mathcal{D}(\cdot)))$	**84.31%**	**84.79%**	**86.72%**	**82.07%**	**83.41%**	**84.26%**

Moreover, setting the quality factor of 20 or 90 takes them both far from the optimal working range of the JPEG compression algorithms, yielding similar results. However, even when the quality factor is set to 30, the ASR of the black-box attack can still achieve 60%.

Additionally, in the bit depth experiment, ASRs gradually decreases as the bit depth reduces. When the bit depth is reduced to 3, too much image information is lost, causing the average ASR to decrease to 45%. The Gaussian noise and median blur experiments are illustrated in Fig. 11, the inclusion of noise does affect the overall ASR to some extent, but the fluctuations are not substantial. Even under the worst-case scenario, where the variance parameter of the Gaussian noise is set to 0.1 and the convolution kernel size is set to 25, our attack still manages to achieve a success rate exceeding 70%.

G. Ablation Study

To determine the impact of each module on the attack performance in OptiCloak, we present the ablation studies. Specifically, we gradually remove one or more modules, retrain the model, and compare it with the original model.

Evaluation Procedure: Our main emphasis lies in examining the impact of the three proposed modeling methods on the overall effectiveness of the attack. We denote the GDM as $\mathcal{D}(\cdot)$, DIM as $\mathcal{V}(\cdot)$, and PLRM as $\mathcal{P}(\cdot)$. The original patch without any transformation is used as the baseline method, which is then projected onto the target vehicle in real-world scenario. YOLO V4 is employed to record the ASRs of five different vehicles.

Next, we test the contribution of each individual module to the attack, namely $\mathcal{D}(\cdot)$, $\mathcal{V}(\cdot)$, and $\mathcal{P}(\cdot)$. Then, we gradually add them to the adversarial patch training task in various combinations, including $\mathcal{V}(\mathcal{D}(\cdot))$, $\mathcal{P}(\mathcal{V}(\cdot))$, $\mathcal{P}(\mathcal{D}(\cdot))$, and $\mathcal{P}(\mathcal{V}(\mathcal{D}(\cdot)))$. Similarly, we record the ASRs for each combination on five different vehicles.

Results: Table I presents the results of using the different combination modules in projection-based attacks. It is clear that using only GDM or PLRM yields similar attack performance to the original patch attack. However, incorporating DIM can enhance the attack performance to some extent, such as achieving 21.42% improvement over the original attack's 22.47%. Moreover, among the pairwise combinations, the combination of DIM and GDM can effectively learn the noise and transformation patterns in the physical environment, providing certain adversarial capabilities to further improve ASR. Notably, when all the three modules are fully combined with training, the attack performance improves significantly, with an average ASR of 84.26% for the tested vehicles. We attribute this result to the fact that a single technical module requires the collaboration of the other technical modules to maximize the performance of the model, making it more robust and flexible.

IX. CONCLUSION

In this work, we propose a light-projection vanishing attack named OptiCloak, which is specifically designed for the vehicle detectors. The attacker can remotely control a drone to project an adversarial patch onto the rear windshield of the target vehicle. Specifically, we incorporate GDM, DIM, and PLRM into the training process of the PM. Additionally, we utilize *RFCF* to effectively increase the attack distance. Finally, we present a black-box attack framework capable of achieving gradient-free patch updates. A comprehensive set of experiments demonstrate that OptiCloak exhibits stronger robustness and effectiveness compared to the printed patch attacks.

In the future, we intend to advance projection-based attack into the domain of 3-D object detection. This involves leveraging light projection techniques to actively disrupt the real-time depth estimation algorithms. Furthermore, our research goals encompass the development of defense mechanisms tailored to mitigate this form of attack, with a particular emphasis on exploiting the spatial pyramid structure.

REFERENCES

[1] H. Zhu, M. Li, Y. Zhu, and L. M. Ni, "HERO: Online real-time vehicle tracking," *IEEE Trans. Parallel Distrib. Syst.*, vol. 20, no. 5, pp. 740–752, May 2008.

[2] A. Athalye, L. Engstrom, A. Ilyas, and K. Kwok, "Synthesizing robust adversarial examples," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2018, pp. 284–293.

[3] K. Liang and B. Xiao, "StyLess: Boosting the transferability of adversarial examples," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2023, pp. 8163–8172.

[4] X. Sun, G. Cheng, L. Pei, H. Li, and J. Han, "Threatening patch attacks on object detection in optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, pp. 1–10, May 2023.

[5] C. Sitawarin, A. N. Bhagoji, A. Mosenia, M. Chiang, and P. Mittal, "DARTS: Deceiving autonomous cars with toxic signs," 2018, *arXiv:1802.06430*.

[6] K. Eykholt et al., "Robust physical-world attacks on deep learning visual classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2018, pp. 1625–1634.
[7] L. Ding et al., "Towards universal physical attacks on single object tracking," in *Proc. Conf. Artif. Intell. (AAAI)*, 2021, pp. 1236–1245.
[8] N. Morgulis, A. Kreines, S. Mendelowitz, and Y. Weisglass, "Fooling a real car with adversarial traffic signs," 2019, *arXiv:1907.00374*.
[9] G. Tang, T. Jiang, W. Zhou, C. Li, W. Yao, and Y. Zhao, "Adversarial patch attacks against aerial imagery object detectors," *Neurocomputing*, vol. 537, pp. 128–140, Jun. 2023.
[10] N. Hingun, C. Sitawarin, J. Li, and D. Wagner, "REAP: A large-scale realistic adversarial patch benchmark," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2023, pp. 4640–4651.
[11] 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 *Proc. ACM Conf. Comput. Commun. Security. (CCS)*, 2019, pp. 1989–2004.
[12] T. Bai, J. Luo, and J. Zhao, "Inconspicuous adversarial patches for fooling image-recognition systems on mobile devices," *IEEE Internet Things J.*, vol. 9, no. 12, pp. 9515–9524, Jun. 2022.
[13] R. Lapid, E. Mizrahi, and M. Sipper, "Patch of invisibility: Naturalistic black-box adversarial attacks on object detectors," 2023, *arXiv:2303.04238*.
[14] A. Gnanasambandam, A. M. Sherman, and S. H. Chan, "Optical adversarial attack," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2021, pp. 92–101.
[15] D.-L. Nguyen, S. S. Arora, Y. Wu, and H. Yang, "Adversarial light projection attacks on face recognition systems: A feasibility study," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. Workshops. (CVPRW)*, 2020, pp. 814–815.
[16] B. Nassi, Y. Mirsky, D. Nassi, R. Ben-Netanel, O. Drokin, and Y. Elovici, "Phantom of the ADAS: Securing advanced driver-assistance systems from split-second phantom attacks," in *Proc. ACM Conf. Comput. Commun. Security. (CCS)*, 2020, pp. 293–308.
[17] "Technical specifications for safety of power-driven vehicles operating on roads." 2017. [Online]. Available: https://openstd.samr.gov.cn/
[18] T. Li, T. Grenier, and H. Benoit-Cattin, "Color space influence on mean shift filtering," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, 2011, pp. 1469–1472.
[19] X. Chen et al., "ZO-AdaMM: Zeroth-order adaptive momentum method for black-box optimization," in *Proc. 33rd Adv. Neural Inf. Process. Syst. (NIPS)*, 2019, pp. 1–12.
[20] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2016, pp. 779–788.
[21] A. Farhadi and J. Redmon, "YOLOv3: An incremental improvement," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2018, pp. 1–6.
[22] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.
[23] W. Liu et al., "SSD: Single shot MultiBox detector," in *Proc. 14th Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 21–37.
[24] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2015, pp. 1–9.
[25] C. Szegedy et al., "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2015, pp. 1–9.
[26] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," 2015, *arXiv:1412.6572*.
[27] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "DeepFool: A simple and accurate method to fool deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2016, pp. 2574–2582.
[28] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *Proc. IEEE Symp. Security Privacy (SP)*, 2017, pp. 39–57.
[29] V. Fischer, M. C. Kumar, J. H. Metzen, and T. Brox, "Adversarial examples for semantic image segmentation," 2017, *arXiv:1703.01101*.
[30] T. B. Brown, D. Mané, A. Roy, M. Abadi, and J. Gilmer, "Adversarial patch," 2018, *arXiv:1712.09665*.
[31] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2017, pp. 39–57.
[32] S. T. Jan, J. Messou, Y.-C. Lin, J.-B. Huang, and G. Wang, "Connecting the digital and physical world: Improving the robustness of adversarial attacks," in *Proc. Conf. Artif. Intell. (AAAI)*, 2019, pp. 962–969.
[33] C. Hu, W. Shi, and L. Tian, "Adversarial color projection: A projector-based physical attack to DNNs," 2023, *arXiv:2209.09652*.
[34] G. Lovisotto, H. Turner, I. Sluganovic, M. Strohmeier, and I. Martinovic, "SLAP: Improving physical adversarial examples with {Short-Lived} adversarial perturbations," in *Proc. 30th USENIX Security. Symp. (USENIX Security.)*, 2021, pp. 1865–1882.
[35] Y. Zhong, X. Liu, D. Zhai, J. Jiang, and X. Ji, "Shadows can be dangerous: Stealthy and effective physical-world adversarial attack by natural phenomenon," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2022, pp. 15345–15354.
[36] M. Shen, Z. Liao, L. Zhu, K. Xu, and X. Du, "VLA: A practical visible light-based attack on face recognition systems in physical world," *ACM Interact. Mobile Wearable Ubiquitous Technol. (IMWUT)*, vol. 3, no. 3, pp. 1–19, 2019.
[37] 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 *Proc. ACM Conf. Comput. Commun. Security. (CCS)*, 2016, pp. 1528–1540.
[38] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," 2019, *arXiv:1706.06083*.
[39] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2017, pp. 618–626.

Huixiang Wen received the M.S. degree from the Department of Electronics and Communication Engineering, Jiangsu University of Science and Technology, Zhenjiang, China, in 2020. He is currently pursuing the Ph.D. degree with the School of Computer Science and Technology, Donghua University, Shanghai, China.

His research interests include Internet of things, machine learning security, sensor security, and system security.

Shan Chang (Member, IEEE) received the Ph.D. degree in computer software and theory from Xi'an Jiaotong University, Xi'an, China, in 2012.

From 2009 to 2010, she was a Visiting Scholar with the Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Kong. She was also a Visiting Scholar with BBCR Research Lab, University of Waterloo, Waterloo, ON, Canada, from 2010 to 2011. She is currently a Professor with the Department of Computer Science and Technology, Donghua University, Shanghai, China. Her research interests include security and privacy in mobile networks and sensor networks.

Dr. Chang is a member of IEEE Computer Society, Communication Society, and Vehicular Technology Society.

Luo Zhou received the M.S. degree from the Department of Electronics and Communication Engineering, Jiangsu University of Science and Technology, Zhenjiang, China, in 2020. He is currently pursuing the Ph.D. degree with the School of Computer Science and Technology, Donghua University, Shanghai, China.

His research interests include ubiquitous and pervasive computing, mobile computing/sensing, and edge computing.

Wei Liu received the Ph.D. degree in computer science from Donghua University, Shanghai, China, in 2021.

He is now an Associate Professor with the Department of Computer Science and Technology, Anhui University of Finance and Economics, Bengbu, China. His research interests include ubiquitous and pervasive computing, mobile computing, and wireless sensing.

Hongzi Zhu (Senior Member, IEEE) received the Ph.D. degree in computer science from Shanghai Jiao Tong University, Shanghai, China, in 2009.

He was a Postdoctoral Fellow with the Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Kong, and the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, USA, in 2009 and 2010, respectively. He is currently a Professor with the Department of Computer Science and Engineering, Shanghai Jiao Tong University. His research interests include mobile sensing and computing and Internet of Things.

Prof. Zhu received the Best Paper Award from IEEE Globecom 2016. He was a leading Guest Editor of *IEEE Network Magazine*. He is an Associate Editor of the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY. He is a member of IEEE Computer Society, IEEE Communication Society, and IEEE Vehicular Technology Society. For more information, please visit http://lion.sjtu.edu.cn.