Received: 13 July 2022 | Revised: 18 July 2022 | Accepted: 24 August 2022 | Published online: 28 September 2022

RESEARCH ARTICLE

Journal of Computational and Cognitive Engineering
2022, Vol. 1(4) 152–158
DOI: 10.47852/bonviewJCCE2202322

BON VIEW PUBLISHING

Multiview Robust Adversarial Stickers for Arbitrary Objects in the Physical World

Scott Oslund[1], Clayton Washington[2], Andrew So[3], Tingting Chen[3,*] and Hao Ji[3]

[1]*University of California, USA*

[2]*Ohio State University, USA*

[3]*California State Polytechnic University, USA*

Abstract: Among different adversarial attacks on deep learning models for image classification, physical attacks have been considered easier to implement without assuming access to victims' devices. In this paper, we propose a practical new pipeline for launching multiview robust physical-world attacks, by creating printable adversarial stickers for arbitrary objects. In particular, a 3D model is used to estimate the camera pose in the photo. Then, by perturbing a part of the 3D model's texture, rendering it, and overlaying the perturbation onto the physical images, realistic training images can be obtained for training a robust adversarial sticker. Experiments with our pipeline show that highly effective adversarial stickers can be generated for many different objects of different sizes and shapes while also achieving a higher attack success rate than attacks that do not utilize camera pose estimation and 3D models. In addition, by using different backgrounds in training and adding randomness to training images, the created stickers continue to function in varied environments. Attacks also remain robust in black-box tests.

Keywords: adversarial attacks, image classification, physical world

1. Introduction

In recent years, deep learning neural networks have been extensively studied, especially with applications in many critical systems such as autonomous driving, security screening, and medical imaging (Akhtar & Mian, 2018; Minagi & Takemoto, 2022). At the same time, deep neural networks (DNNs) are facing the threat of adversarial examples which add perturbations to the original input in order to fool the DNN models and cause system malfunctions (Szegedy et al., 2014). In the Computer Vision domain, adversarial examples have been well explored in the 2D realm by changing the pixel values in images (Akhtar et al., 2021; Goodman et al., 2020). However, these types of adversarial attacks rely on the assumption that the attacker can access the images in the short time frame between the image being generated and sent to the prediction model. With physically secured cameras, it is unrealistic for attackers to manipulate the 2D images. Physical attacks are believed to be more practical (Brown et al., 2017; Ren et al., 2021).

To launch physical adversarial attacks, two approaches have been studied, i.e., (1) 3D-printing the physical adversarial object (Athalye et al., 2018; Tsai et al., 2020), and (2) creating a 2D adversarial sticker and attaching it to the target object (Duan et al., 2019; Eykholt et al., 2018; Komkov & Petiushko, 2021; Thys et al., 2019). For the 3D-printing approach, the main limitation is that in realistic scenarios, 3D-printed objects will lose the function of the original ones and thus make the attack less meaningful. For example, in order to hide a real gun from an automatic security scan system by fooling the vision classifier, the attacker will not choose to 3D-print a perturbed object that looks like a gun but does not function. On the other hand, adversarial stickers are easy to directly apply on the real target objects. Although stickers are usually more perceptible, in realistic computer vision systems such as in fully autonomous vehicles, decisions are made in real time without much if any human intervention. Existing works on adversarial stickers have shown the effectiveness of this approach in achieving high attack success rates. However, due to the 2D nature of the stickers, in previous research, most experiments have been performed on objects with flat surfaces (e.g., stop sign) or close-to-flat surfaces (e.g., microwave oven).

In this paper, we explore the boundary of realistic physical attacks with adversarial stickers, that can be attached to objects with irregular shapes and arbitrary surfaces, and remain effective from various viewpoints in different environmental conditions. We identify the technical challenges as follows: (1) with only changing the color (texture) of the stickers, it is not known what transformations are needed to keep the 2D perturbation effective

*Corresponding author: Tingting Chen, California State Polytechnic University, USA. Email: tingtingchen@cpp.edu.

on arbitrary 3D surfaces; (2) when choosing the data used for training the stickers, we need to make sure sufficient images are included to achieve the robustness of the attack, and at the same time consider the realistic situation when public datasets with physical photos are not available; (3) in order to determine the locations of texture perturbations (i.e., where to put the stickers on the object), more work is needed to find the most susceptible areas to adversarial perturbations; (4) on top of the three aforementioned challenges, achieving attack robustness from various viewing angles and distances in different environmental conditions is difficult.

To tackle the challenges of non-flat object surfaces, our approach leverages a photo-realistic 3D model of the target physical object in training. With the 3D model, our pipeline estimates the camera pose of the physical photos of the target, to facilitate the transformation. By perturbing a part of the 3D model's texture, rendering it, and overlaying the perturbation onto the physical images, realistic training images can be obtained for training a robust adversarial sticker. Since smaller and therefore less conspicuous perturbations require a relatively large amount of training data over multiple iterations of training, using our algorithm an attacker may save considerable effort by taking the few photos required to produce a 3D reconstruction of a real object and training on a virtual dataset using a differentiable renderer rather than taking hundreds of photos and applying a physical perturbation to the object after each iteration of training.

The contributions of our work in this paper include the following.

- We propose a realistic physical attack method with adversarial stickers that can be attached to arbitrary target objects with irregular shapes, to fool deep learning image classifiers.
- Our adversarial sticker generation pipeline takes a few physical photos of the target as input and leverages a 3D model to train the texture of the sticker. 3D model-aided camera pose estimation and randomness are introduced to achieve attack robustness across multiple viewpoints and physical environment conditions.
- We have performed extensive experiments with target objects of different shapes and in a variety of viewpoints and backgrounds, which show the high attack success rates of our stickers.

2. Related Work

To date, studies regarding adversarial machine learning have focused on applying minimally conspicuous alterations to inputs for classifier neural networks to bias the output of the classifiers away from the correct output class. Just as parameters in DNNs are changed using gradients to minimize loss during training, components of inputs into a classifier are altered using gradients in an adversarial attack to achieve a specific objective.

2.1. Digital adversarial attacks

Adversarial attacks were first studied in the context of still 2D images. In the first instances of these attacks, all pixel colors were subject to alteration in the adversarial training task, and the resulting perturbations were barely perceptible to the human eye despite fooling image classifiers reliably (Goodfellow et al., 2015). This spurred the development of algorithms like Fast Gradient-Sign Method (FGSM) (Goodfellow et al., 2015) to facilitate faster adversarial training and Basic Iterative Method (BIM) (Kurakin et al., 2016) to improve FGSM by making many small iterative gradient-based optimizations to the input to achieve the adversarial objective. Following these early attacks, studies like Moosavi-Dezfooli et al. (2016) and Papernot et al. (2016) have focused on minimizing the size of effective perturbations through more targeted selection of input features to perturb, with (Vargas & Sakurai, 2019) achieving attack robustness while only modifying one pixel in the victim image.

Beyond altering 2D images to produce adversarial examples, researchers have also executed adversarial attacks by passing unedited renders of adversarially perturbed 3D digital objects and rendering environments into classifier neural networks. In Yao et al. (2020), the textures of various 3D digital objects were adversarially altered using BIM so that renders of the perturbed objects taken from many angles would fool the Inception v3 CNN architecture, hence achieving multiview robustness. Zeng et al. (2019) takes a different approach, manipulating the position of the camera in the rendering environment relative to the 3D digital objects to decrease a classifier's ability to classify the contents of rendered images and impede visual question answering without altering the 3D objects themselves. Work has also been done to adversarially perturb the geometries of 3D objects composed of polyhedral meshes (Yang et al., 2018) and point clouds (Cao et al., 2019; Liu et al., 2019; Xiang et al., 2019). These 3D digital adversarial attacks demonstrate the breadth of attributes of data that can be adversarially modified, going beyond the original attacks on 2D static images.

2.2. Physical adversarial attacks

Since beginning in virtual environments, adversarial attacks have been applied to real physical objects. A physical attack consists of applying some physical attachment to a physical object or near one in a photograph, photographing the object with the attachment applied or in the background, and passing the photograph through a classifier without applying any of the techniques discussed in the first subsection. The first physical adversarial examples were achieved by printing adversarially perturbed 2D static images and passing photos of the prints taken by a smartphone into a classifier (Kurakin et al., 2016). Another pioneer in physical-world adversarial attacks is Adversarial Patch (Brown et al., 2017) in which researchers placed perturbed stickers near victim objects to fool classifiers. Other significant early work defined important properties of physical adversarial perturbations. Sharif et al. (2016) introduces the Non-Printability Score loss function for physical adversarial attacks to decrease difference between colors in digital perturbations and physical realizations of the perturbations. Sharif et al. (2016) also uses a Total Variation loss function to generate localized perturbations with minimal visible color difference within small local areas of a perturbation, encouraging visible smoothness of the perturbations.

Following foundational work in physical adversarial attacks, researchers have further developed techniques for generating adversarial patches that are multiview-robust, camouflaged, and can be applied to objects with non-flat geometries, while others have deviated from the adversarial patch framework entirely. Eykholt et al. (2018) demonstrates that single-view robust adversarial stickers can be trained from a single 2D physical photo and that multiview-robust flat stickers applied to 3D objects can be effectively adversarially trained using multiple 2D physical photos. Athalye et al. (2018) demonstrates another multiview robust physical adversarial attack in which the authors created adversarially colored 3D-printed objects, utilizing Expectation over Transformation to achieve some multiview robustness from

few images. Researchers have also generated adversarial 3D-printed objects by sampling points from adversarially generated digital point clouds (Tsai et al., 2020). Other attacks like Komkov & Petiushko (2021) and Thys et al. (2019) have fooled detection and classification of DNNs by applying adversarial stickers to human faces to the ends of dodging and impersonation. Duan et al. (2019) shows that stickers in physical attacks can be applied in the background of photographs containing the victim object and can be constrained to appear semantically relevant to the victim object, such as an adversarially perturbed brand-name banana sticker in an image of a banana. Researchers have also extended the current patch approach to adversarial computer vision tasks to attack object detector DNNs (Chen et al., 2019; Du et al., 2021). Some work such as Xu et al. (2020) has been done to extend adversarial attacks to objects with dynamic non-flat surface geometry, but such techniques do not currently generalize to many object types. Physical adversarial attacks incorporate many techniques from the 2D static image and digital rendering environment domains while also adapting to unique challenges posed by the variability of real physical environments, necessitating the investigation of robust techniques that use little physical data to achieve multiview robustness over a range of real-world conditions.

3. Proposed Methodology

3.1. Selecting perturbation location

To locate the regions of the victim object that are most vulnerable to adversarial attack, we apply Gradient Class-Activation Mapping (Grad-CAM) (Selvaraju et al., 2019) to a set of images of the victim object that are correctly classified by a DNN classifier. Grad-CAM then produces a heat map as shown in Figure 1 that visualizes the importance of regions of the image to the classifier's class choice. We then produce a texture mask M_T to be multiplied point-wise with the 3D digital object's texture image T during training so that only a small bounded portion of T is subjected to adversarial perturbation.

3.2. Camera pose estimation with physical photos

We next seek to overlay an appropriately transformed image of the sticker perturbation onto each real image using camera pose estimation within the PyTorch3D (Ravi et al., 2020) differentiable rendering environment.

Prior to training, we obtain a set of real images $\{R\}$ of the victim object that include the attack region, a 3D digital reconstruction $X(T)$ of the victim object with a texture image T, and a set of edited images $\{R_W\}$ containing each image in $\{R\}$. At each round of camera pose optimization, a silhouette render Y_i of $X(T)$ is produced with the camera at position $(\rho, \theta, \phi)_{i-1}$ and compared with the target image $t_W \in \{R_W\}$ and optimized using Adam to produce a camera position $(\rho, \theta, \phi)_i$, such that after all k iterations of optimization the camera pose in t is learned as $(\rho, \theta, \phi)_k$ and can be used to produce a mask matrix M_P to cover all background area of the render other than the region to be perturbed and another mask matrix M_B to cover the region to be perturbed and expose the background area.

3.3. Sticker training loop

3.3.1. Introducing randomness to training

In order to make the perturbations more robust to physical conditions, we apply multiple types of random modification to each training image $tr \in \{Tr\}$. Assume the training image generated pertains to some physical image $t \in \{R\}$.

Our images are matrices of RGB color values in the space $[0,1]^3$. First, we generate a random noise matrix N_R with the same dimensions as the training image and random values ranging from -0.025 to 0.025. We also seek to achieve robustness to slight variations in the position of the applied sticker. To add randomness to the sticker location, we slightly perturb the camera position to achieve some (ρ, θ, ϕ)'. From this camera position, we obtain a mask to expose only the region to be perturbed in the physical image from $\{R\}$ and another mask M_B to expose only the areas of the physical image that are not subjected to perturbation.

Composing these approaches to introducing randomness, we can generate our training image out of the digital rendered image so that

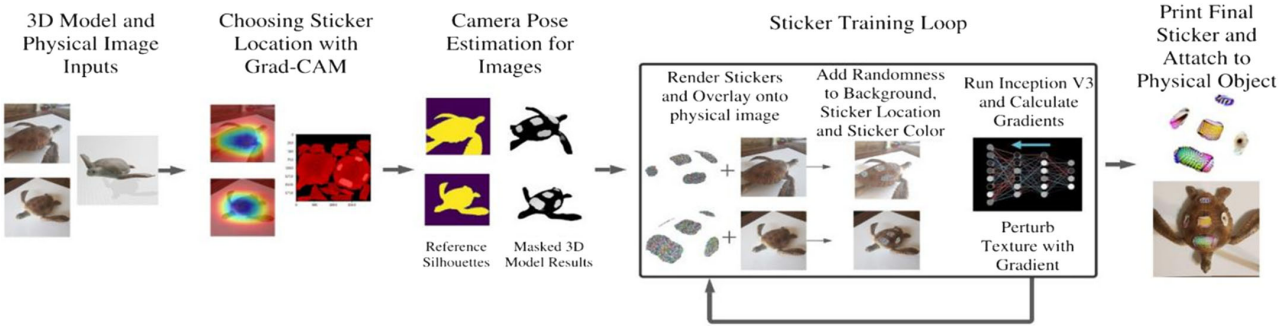
$$tr = Y * M_P + (t + N_R) * M_B$$

where $*$ denotes point-wise matrix multiplication, M_P and M_B are produced by adding some small random perturbation to the camera pose, and tr is clamped to the $[0,1]^3$ space.

3.3.2. Loss functions

Let $C_{correct}$ be the correct class label of an image Y and $C_{predict}$ be the predicted class label of a DNN classifier $f(\bullet)$. In this paper, we rely on true positive images for which the class label

Figure 1
Adversarial sticker generation pipeline making use of physical photos, 3D models, and randomness in training



$C_{predict} = C_{correct}$, to learn adversarial perturbation on the chosen local regions. Let S_X be the set of pixels in the digital texture image T that comprise the perturbation. Our loss function consists of three losses:

- L_{CE} (S_X,Y) is the cross-entropy loss given by:

$$L_{CE}(S_X, Y) = \log p_{Y,c}$$

where $p_{Y,c}$ is the classifier-predicted confidence that the image Y is of the class $C_{correct}$.

- L_{NPS} (S_X) is the Non-Printability Score loss function described in Sharif et al. (2016) given by

$$L_{NPS}(S_X) = \sum_{S \in S_X} \prod_{p \in P} |s - p|$$

where $P \subset [0,1]^3$ is a set of printable RGB triplets. This encourages the perturbations to take color values that can be produced by an ordinary color printer enumerated in P.

- L_{TV} (S_X) is the Total Variation loss function also used in Sharif et al. (2016) given by:

$$L_{TV}(S_X) = \sum_{(i,j) \in S_X} \left((q_{i,j} - q_{i+1,j})^2 + (q_{i,j} - q_{i,j+1})^2 \right)^{\frac{1}{2}}$$

where $q_{i,j}$ is a pixel in the set S_X located at position (i,j) in the texture image T. This function smooths the image by encouraging low color difference between adjacent pixels.

Thus, for each image, the overall loss is evaluated by:

$$L(S_X, Y) = \epsilon * L_{CE}(S_X, Y) + \alpha * L_{NPS}(S_X) + \alpha * L_{TV}(S_X)$$

where ϵ is the noise magnitude and α is the smoothing and color correction magnitude and $sgn(\epsilon) = -sgn(\alpha)$.

3.3.3. Complete training procedure

Algorithm 1 demonstrates the complete procedure for training the perturbations. Figure 1 visually shows the same procedure.

Input $\{R\}$, $\{R_W\}$, $X(T)$, M_T, {Cam_poses}, num_iterations
Output perturbed texture T
ϵ = 0.01;
for *i in num_iterations* **do**
  gradient = 0;
  for $R \in \{R\}$, $R_W \in \{R_W\}$, $cam\_pose \in \{Cam\_poses\}$ **do**
    (ρ, θ, ϕ) = Add_noise_cam(cam_pose);
    M_P, M_B = Mask($X(T)$, M_T, ρ, θ, ϕ);
    $Y = r(X(T), \rho, \theta, \phi)$;
    tr = Add_noise_im(Y, M_P, R, M_B);
    gradient = gradient + $sign(\nabla_{M_T} L(M_T, tr))$;
  end
  $T = T + \epsilon *$ gradient;
end

Algorithm 1: Adversarial stickers training

3. Experiments

To examine our pipeline, we choose five objects, varying in size and shape, to create stickers for, i.e., iPhone 5, Combination lock, Coffee Mug, Sea Turtle, and Computer Monitor. Between 7 and 16 physical training images were used for each object. In our experiments, Inception v3 is the victim classifier unless otherwise stated. Attack success rate refers to the percentage of misclassified test images, or in the case of targeted attacks, the percentage of test images classified as the target class. Figure 2 shows sample testing images taken from experiments.

3.1. Effectiveness of attacks

We first test our full pipeline in a variety of environments, while examining the impact of backgrounds and lighting in training images.

We test stickers trained with images using a white background and stickers trained with varied backgrounds. We then test the results in two different physical situations, on a dimly lit black kitchen table and on an artificially lit patterned bed with a map in the background. Examples of each environment can be found in Figure 2. The resulting attack success rates, both of the original stickers trained with a white background and of the new stickers trained on varied backgrounds, can be found in Table 2.

3.2. Comparisons to modified pipeline

Next, to investigate the contribution of each component of our adversarial sticker training pipeline in achieving a high attack success rate, we perform comparison experiments to modified versions of our full sticker training pipeline as mentioned in Table 1. In each variation, we modify only one component of our sticker generation pipeline at a time (i.e., without using 3D models in training, using 3D renderings instead of physical photos, and including no randomness). With other training parameters and conditions kept the same, we print out the stickers generated with each modified training pipeline. Using a consistent testing process, we find the attack success rates of each variation in Table 3.

3.3. Transferablility of attacks

To study the transferability of our attack method, we run black-box experiments using various other image classifiers. From the results in Table 4, we observe a significant increase in the attack success rate compared to the misclassification rate when no attack is launched (with the minimum increase being from 0% to 10.1% and the maximum increase being 0.3% to 100.0%).

4. Conclusion

In this paper, we study realistic physical attacks on deep learning image classifiers. We expand upon past research in the domain of adversarial stickers by using the baseline methodology and extending it further. In our attack methods, we incorporate ideas from past research such as using Grad-CAM to determine optimal sticker placement and making the loss function a combination of several functions to reduce variation in the sticker and increase printability. Our method also uses randomness in the training process, by adding perturbations to each image during training, forcing stickers to be robust to imperfections in the real world.

Furthermore, unlike past research, our method uses a 3D model in conjunction with physical photos and camera pose estimation to render adversarial stickers onto a physical photo from many different viewpoints. In this way, stickers can be trained from

Figure 2
Sample images and classifications of each object from each experiment. Each column corresponds to a pipeline variation listed in Table 1. Green boundaries indicate correct classifications and red boundaries indicate misclassifications

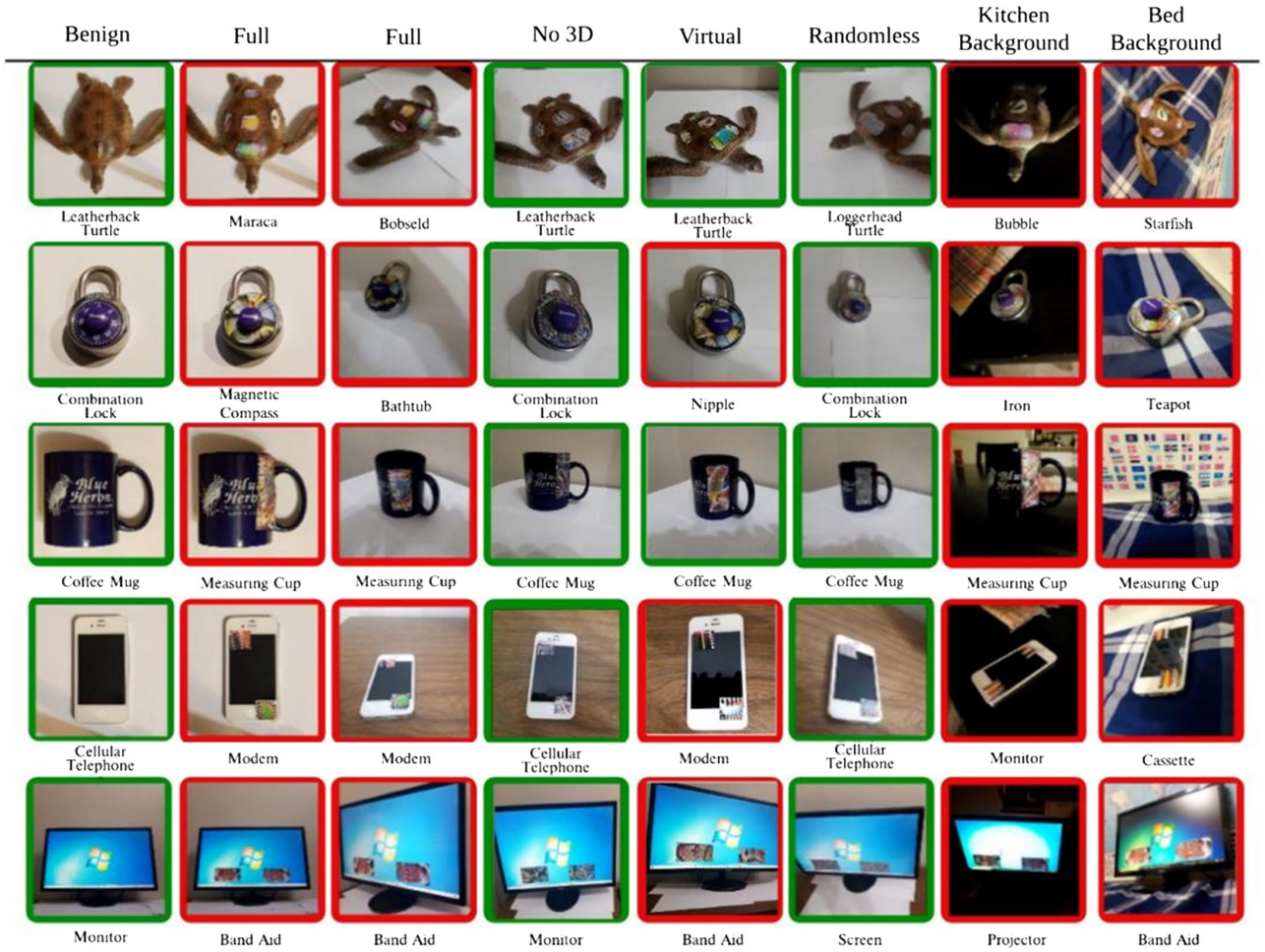


Table 1
A summary of the variations of the full pipeline used for testing our pipeline and the corresponding labels

Label of modification	Modifications made compared to the full pipeline
Full	Full adversarial sticker training pipeline
No 3D	No 3D models used in the training of the object
Virtual	Trained using 3D renderings instead of physical photos
Randomless	Trained with no randomness integrated into the pipeline

Table 2
Attack success rate (%) in different environments using stickers trained with a white background and stickers trained with various different backgrounds

	Well lit patterned bed with map in testing			Dim lit kitchen in testing		
Object	Benign	White backgrd.	Various backgrd.	Benign	White backgrd.	Various backgrd.
iPhone	28.0	82.8	96.8	47.9	83.8	95.6
Combination lock	50.6	92.2	93.5	40.8	79.6	71.3
Coffee mug	13.4	77.9	98.2	50.1	80.4	78.3
Sea turtle	34.2	99.3	99.8	26.2	87.0	94.6
Monitor	5.0	93.2	96.5	33.4	57.3	70.9

Table 3
Attack success rate (%) comparison of different variations of the full pipeline

Object	Benign	Full	No 3D	Virtual	Randomless
iPhone	8.5	98.2	15.9	86.1	22.4
Combination lock	0.3	99.7	35.4	72.7	75.4
Coffee mug	3.2	88.4	8.6	56.1	20.4
Sea turtle	2.7	100.0	15.6	99.4	18.3
Monitor	0.0	88.4	3.9	82.7	2.5

Table 4
Black-box attack success rates for image classifiers (%) (Benign/Adversarial)

Object	ResNet	MNASNet	DenseNet	ResNeXt
iPhone	13.7/84.0	46.2/100.0	0.30/100.0	18.9/98.5
Combination lock	31.9/99.8	29.6/97.1	3.0/55.6	12.3/94.5
Coffee mug	11.4/73.3	22.2/32.7	2.3/55.6	2.7/48.2
Sea turtle	61.9/94.0	27.3/96.1	0.0/58.7	7.7/78.3
Monitor	0.9/65.1	7.6/98.0	0.0/10.1	3.2/99.7

multiple viewpoints at once to create a sticker that functions from many angles, while retaining the realism of physical photos. This offers an improvement from past research which would either use rendered photos losing realism or use physical photos but only from limited viewpoints due to the need for a human to map the sticker renderings onto the photo.

Our contributions in this paper offer many benefits. By using our method of rendering stickers onto physical photos, stickers are no longer limited to being flat. Instead, they can be fit onto a 3D model in any position and rendered onto a physical photo. Previously, stickers were usually limited to being on flat surfaces, since their position would be manually decided. Furthermore, we find in our experiments that attacks using both multiple viewpoints in training and physical photographs reach higher attack success rates than methods without one of these attributes. Previous research was limited to using few viewpoints and physical photos or many viewpoints and rendered photos. Our algorithm offers a way to obtain many physical photos from many viewpoints, increasing attack success rates. Our attacks continue to function well in a variety of conditions. Extensive experiments on a variety of objects, viewpoints, environments, and across different classifiers in black-box tests show that our attack continues to be robust.

Beyond our paper, there is a need to explore the domain of adversarial attacks utilizing physical photos and 3D models further. In particular, research into improved camera pose estimation using rudimentary phone-generated 3D models could greatly increase the practicality and ease in carry out this attack. Additionally, using frames from a video in conjunction with these methods to achieve a large number of physical training images could make adversarial attacks extremely simple to pull off.

Acknowledgment

This work was supported in part by NSF grant CNS-2050826.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

References

Akhtar, N., & Mian, A. (2018). Threat of adversarial attacks on deep learning in *computer vision*: A survey, arXiv:1801.00553 [cs.CV].

Akhtar, N., Mian, A., Kardan, N., & Shah, M. (2021). Advances in adversarial attacks and defenses in computer vision: A survey. *IEEE Access 9*, 155161–155196. https://doi.org/10.48550/arXiv.2108.00401

Athalye, A., Engstrom, L., Ilyas, A., & Kwok, K. (2018). Synthesizing robust adversarial, examples. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*. https://arxiv.org/abs/1707.07397

Brown, T. B., Mané, D., Roy, A., Abadi, M., & Gilmer, J. (2017). Adversarial patch. *CoRR* abs/1712.09665, arXiv:1712.09665. http://arxiv.org/abs/1712.09665

Cao, Y., Xiao, C., Cyr, B., Zhou, Y., Park, W., Rampazzi, S., … Mao, Z. M. (2019). Adversarial sensor attack on LiDAR-based perception in autonomous driving. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security* (November 2019). https://doi.org/10.1145/3319535.3339815

Chen, S.-T., Cornelius, C., Martin, J., & Chau, D. H. (2019). ShapeShifter: Robust Physical Adversarial Attack on Faster R-CNN Object Detector. In *ECML PKDD 2018: Machine Learning and Knowledge Discovery in Databases*, Lecture Notes in Computer Science (pp. 52–68). https://doi.org/10.1007/978-3-030-10925-7_4

Du, A., Chen, B., Chin, T.-J., Law, Y. W., Sasdelli, M., Rajasegaran, R., & Campbell, D. (2021). Physical adversarial attacks on an aerial imagery object detector, arXiv:2108.11765 [cs.CV].

Duan, R., Ma, X., Wang, Y., Bailey, J., Qin, A. K., & Yang, Y. (2019). Adversarial camouflage: Hiding physical-world attacks with natural styles, arXiv:2003.08757 [cs.CV].

Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., … Song, D. (2018). Robust PhysicalWorld Attacks on Deep Learning Visual Classification. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 1625–1634). https://doi.org/10.1109/CVPR.2018.00175

Goodfellow, J., Shlens, J., & Szegedy, C. (2015). Explaining and harnessing adversarial examples, arXiv:1412.6572 [stat.ML].

Goodman, D., Xin, H., Yang, W., Yuesheng, W., Junfeng, X., & Huan, Z. (2020). Advbox: a toolbox to generate adversarial examples that fool neural networks, arXiv preprint arXiv:2001.05574.

Komkov, S., & Petiushko, A. (2021). AdvHat: Real-world adversarial attack on ArcFace face ID system. In *2020 25th International Conference on Pattern Recognition (ICPR)*. https://doi.org/10.1109/icpr48806.2021.9412236

Kurakin, I., Goodfellow, J., & Bengio, S. (2016). Adversarial examples in the physical world. CoRR abs/1607.02533, arXiv:1607.02533. http://arxiv.org/abs/1607.02533

Liu, D., Yu, R., & Su, H. (2019). Extending adversarial attacks and defenses to deep 3D point cloud classifiers, arXiv:1901.03006 [cs.CV].

Minagi, H. H., & Takemoto, K. (2022). Natural images allow universal adversarial attacks on medical image classification using deep neural networks with transfer learning. *MDPI Journal of Imaging*, *8*, 38. https://doi.org/10.3390/jimaging8020038

Moosavi-Dezfooli, S.-M., Fawzi, A., & Frossard, P. (2016). DeepFool: A simple and accurate method to fool deep neural networks, arXiv:1511.04599 [cs.LG].

Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Celik, Z. B., & Swami, A. (2016). The limitations of deep learning in adversarial settings. In *2016 IEEE European Symposium on Security and Privacy (EuroSP)* (pp. 372–387). https://doi.org/10.1109/EuroSP.2016.36

Ravi, N., Reizenstein, J., Novotny, D., Gordon, T., Lo, W.-Y., Johnson, J., & Gkioxari, G.. 2020. Accelerating 3D deep learning with PyTorch3D, arXiv:2007.08501 [cs.CV].

Ren, H., Huang, T., & Yan, H. (2021). Adversarial examples: Attacks and defenses in the physical world. *International Journal of Machine Learning and Cybernetics*, *12*, 3325–3336. https://doi.org/10.1007/s13042-020-01242-z

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2019). Grad-CAM: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision 128*, 336–359. https://doi.org/10.1007/s11263-01901228-7

Sharif, M., Bhagavatula, S., Bauer, L., & Reiter, M. K. (2016). Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In *Proceedings of the 23rd ACM SIGSAC Conference on Computer and Communications Security*.

Vargas, J., Su, D. V., & Sakurai, K. (2019). One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation 23*, 828–841. https://doi.org/10.1109/tevc.2019.2890858

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus, R. (2014). Intriguing properties of neural networks, arXiv:1312.6199 [cs.CV].

Thys, S., Van Ranst, W., & Goedemé, T. (2019). Fooling automated surveillance cameras: adversarial patches to attack person detection, arXiv:1904.08653 [cs.CV].

Tsai, T., Yang, K., Ho, T.-Y., & Jin, Y. (2020). Robust adversarial objects against deep learning models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 01, April 2020 (pp. 954–962). https://doi.org/10.1609/aaai.v34i01.5443

Xiang, C., Qi, C. R., & Li, B. (2019). Generating 3D adversarial point clouds, arXiv:1809.07016 [cs.CR].

Xu, K., Zhang, G., Liu, S., Fan, Q., Sun, M., Chen, H., . . . Lin, X. (2020). Adversarial T-shirt! Evading person detectors in a physical world, arXiv:1910.11099 [cs.CV].

Yang, D., Xiao, C., Li, B., Deng, J., & Liu, M. (2018). Realistic adversarial examples in 3D meshes. CoRR abs/1810.05206, arXiv:1810.05206. http://arxiv.org/abs/1810.05206

Yao, P., So, A., Chen, T., & Ji, H. (2020). On multiview robustness of 3D adversarial attacks. In Practice and Experience in Advanced Research Computing (pp. 372–378).

Zeng, X., Liu, C., Wang, Y.-S., Qiu, W., Xie, L., Tai, Y.-W., . . . Yuille, A. L. (2019). Adversarial attacks beyond the image space, arXiv:1711.07183 [cs.CV].

How to Cite: Oslund, S., Washington, C., So, A., Chen, T., & Ji, H. (2022). Multiview Robust Adversarial Stickers for Arbitrary Objects in the Physical World. *Journal of Computational and Cognitive Engineering* 1(4), 152–158, https://doi.org/10.47852/bonviewJCCE2202322