

Breaking Rigidity in Adversarial Patch Attacks

Vishesh Kumar¹, Guha Balakrishnan², and Akshay Agarwal¹

¹ Trustworthy BiometraVision Lab, Indian Institute of Science Education and Research Bhopal, India

² Rice University

{vishesh22, akagarwal}@iiserb.ac.in

{guha}@rice.edu

<https://github.com/visheshrajput2408/SmuDPatch.git>

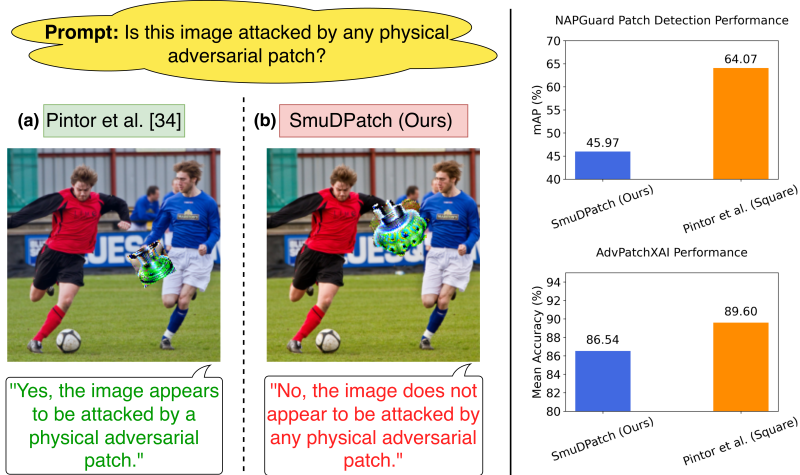


Fig. 1: Comparison between existing fixed (rectangular) shape patches and our proposed deformable patches, and how defense algorithms handle them. The first part (a) illustrates how existing VLM models accurately describe the adversarial nature of images containing fixed-shape patches, and (b) highlights their vulnerability when the proposed deformable and smooth patches are applied to the images. The prompt we use to query the VLM appears at the top of the first part. The second part demonstrates how deformable patches compromise the performance of recent SOTA adversarial patch detectors, including (c) NAPGuard [55] and (d) AdvPatchXAI [24], during detection.

Abstract. Adversarial patch attacks that aim to fool deep neural network models in the real world prove effective in ‘every’ possible network type, including vision-language models (VLMs). Surprisingly, most of the existing adversarial patch generation algorithms develop patches of fixed shape (presumably rectangular or circular in a few instances). Due to such biases toward the shape of an adversarial patch, existing adversarial patch detectors, including VLMs, which yield high accuracy, provide a false sense of security. Our extensive experiments equivalently showcase the bias towards the fixed shape of adversarial patches and their high detection, which might leave a hole in adversarial patch defense against

adaptive patches. In response, we have first proposed a novel deformable adversarial patch generation algorithm with multifold aims: (i) to highlight the vulnerabilities of state-of-the-art (SOTA) image classification algorithms and (ii) to help provide a strong foundation for developing a universal adversarial patch detector. Introducing smooth and deformable patches that significantly increase practical threats in real-world scenarios. Using the proposed algorithm, we developed large-scale adversarial patch detection datasets comprising **140,000** adversarial patch images for rigorous robustness evaluation and generalizable adversarial detection. The proposed deformable patches effectively fool several SOTA image classification models. While the defense is not the primary focus, our preliminary study highlights that detecting the proposed deformable patches is hard. Still, they can enhance security against various patch shapes, including fixed and deformable patches.

Keywords: adversarial patch attack · robustness · adversarial patch defense · VLMs · DNNs

1 Introduction

Since Deep Neural Networks (DNNs) and Vision-Language Models (VLMs) are utilized in a wide range of contexts, including security-related and mission-critical applications [3, 4, 9, 26, 38, 63], *researchers consider understanding their proper security of utmost significance nowadays*. Nonetheless, DNNs and VLMs are vulnerable to adversarial perturbation of input data [34, 43, 54, 61], posing a significant security risk to DNN-based classifiers and the current trend of VLMs for image classification tasks in the real world. To mitigate potential risks and further investigate the robustness of DNNs and VLMs, *researchers highly benefit at the current research stage by identifying and revealing as many of their “blind spots” as possible and ensure that researchers develop universal security*.

On that line, numerous techniques develop adversarial patch examples [2, 5, 8, 36] to identify the vulnerability of DNNs. Interestingly, most of the existing patch-based attacks employ fixed shapes like circles or squares, limiting the concept of universality in defense. To the best of our knowledge, only a few deformable shape-based adversarial patch methods have recently been proposed, such as [7]. However, these approach generate adversarial patches locally for each image, optimizing shape and texture in a per-image manner rather than learning a single universal patch across a dataset. Moreover, their deformation strategy relies on a central point with radial rays, which often produces sharp and irregular boundaries, resulting in unnatural shapes that are difficult to deploy in real-world scenarios. In contrast, our method focuses on generating a smooth deformable shape universal adversarial patch for a chosen target class, sequentially optimizing geometry and texture to improve generalization, and practical applicability. The patch is called universal because the same optimized patch is applied to many different images. Not just from the visual perspective, Fig. 1 highlights the necessity of having smooth deformable patches over fixed-shape patches for trustworthy security against adversarial patch attacks. Fig. 1(a)

shows that LLaVA-NeXT [27] successfully detects fixed square patches, explicitly stating that the image appears adversarial. In contrast, Fig. 1(b) demonstrates that the same model fails to recognize our smooth deformable patches. One might argue that VLM is not trained for patch detection, but it is true for fixed shapes as well; however, VLM is still effective in its detection, but fails against the proposed patches. Moreover, it is worth noting that we have not utilized any VLM knowledge in patch generation. Fig. 1(c) shows that a recent adversarial patch detector, NAPGuard [55], achieves 64.07% mAP on square patches but drops to 45.97% mAP when evaluated on proposed deformable patches, highlighting poor generalization. Finally Fig. 1(d) represents the mean detection accuracy of AdvPatchXAI [24] in unseen patch setting on square and our proposed deformable patches, reinforcing the vulnerability of existing defense methods.

In response to advancing the research in defending adversarial patches, we have proposed an universal adversarial patch generation algorithm overcoming the following primary challenges: **(1)** existing deformable patches are often irregular, insufficiently flexible, or optimized separately for each image, which limits their generalization in universal attack settings [8, 15]; **(2)** Adversarial patches should have a high attack performance. They must be transferable across different DNNs and VLMs because attackers typically do not have detailed knowledge of the target model in the real world. To address these issues, we design a novel shape search method using a centripetal Catmull-Rom spline to achieve compact and smooth shapes discussed in section 3. The proposed deformable patches have extensively demonstrated their attack success against several state-of-the-art image classification models. The vulnerabilities of image classification models highlight the need for varied, deformable patches to reveal true network sensitivity. Our proposed patches also serve as stronger training data for patch detectors, emphasizing that robust security depends on both model design and data quality. Although defense development is not our primary focus, it remains an important direction for future work. In summary, the contributions of this research paper are:

- We propose a novel universal optimization framework to generate smooth, deformable adversarial patches with high attack success and low robust accuracy.
- We introduce two adversarial patch datasets (20 patches, 10 classes, 2 shape types) as benchmarks for evaluating vulnerabilities of SOTA vision models.
- Extensive experiments demonstrate strong transferability and superiority over existing adversarial patch methods.
- A preliminary study shows that training with our deformable patches improves robustness against diverse patch attacks, and we further assess the capability of VLMs in detecting adversarial patches [37, 61].

2 Related work

This section provides a comprehensive overview of the recent and state-of-the-art adversarial patch generation algorithms, followed by a discussion on defense algorithms tackling adversarial patches.

2.1 Adversarial Patch Attacks

Sharif et al. [42] introduced the first physical attack on deep neural networks by developing adversarial eyeglass frames to evade face recognition systems. Brown et al. [5] later proposed a universal patch attack, optimizing patches on an ensemble of models to achieve targeted misclassification. Recently, Pintor et al. [36] proposed ImageNet patches of a fixed shape of 50×50 to evaluate the robustness of CNNs. They optimized the texture of patches using the ensemble of pure convolutional models, leaving the space for robustness evaluation of current SOTA transformer models. A few studies present various deformable shape patch attacks targeting object detection models, while a few present deformable patch attacks for classification models. Xu et al. [56] designed adversarial T-shirts embedding deformable patches to fool human detectors. Wei et al. [50] introduced hotcold blocks using warming and cooling pastes to target infrared detectors, while another study by Wei et al. [52] focused specifically on infrared pedestrian detection. Zhu et al. [66] proposed infrared adversarial patterns generated using 3D mesh shadows and aluminum foil. Chen et al. [7] introduced DAPatch, designed for classification tasks and jointly optimizing shape and texture. However, their resulting shapes tend to be highly irregular and difficult to fabricate, potentially restricting their practical effectiveness in real-world applications. While patch attacks exist to deceive the DNNs, specifically detector models, minimal studies exist on the vulnerability of VLMs against adversarial attacks. Co-Attack by Zhang et al. [60] is one of the initial methods to execute white-box, untargeted attacks on various Vision-Language Models (VLMs). Building on this foundation, subsequent research has introduced several techniques aimed at enhancing the transferability of adversarial attacks in black-box, untargeted scenarios [32, 57, 59, 65]. The cross-prompt attack by Luo et al. [33] presents an innovative approach using language model prompts to improve adversarial transferability. It is to be noted here that the majority of the existing adversarial patch generation algorithms provide fixed-shape (positively square or rectangular) patches [18, 23, 28, 46, 49], and in a white-box setting specific for VLM, making the development of defense algorithms biased towards that shape. Further, no existing study showcases the transferability of adversarial attacks into VLMs generated by DNNs. In response, for the first time, we have generated novel smooth deformable ImageNet patches that can fool pure convolutional models, transformer models, and VLM models, as shown in Fig. 1.

2.2 Adversarial Patch Defense

Due to the stealthy nature of adversarial attacks, numerous detection methods have been proposed [1], yet they remain limited, particularly against imperceptible perturbations. Adversarial training is a strong defense, and [62] extended it to include patch-based attacks; however, it is computationally expensive and suffers from poor generalization. Several benchmark and detection frameworks have been introduced [35, 41, 44, 55], but most defenses focus on fixed-shape patches and fail to generalize across varying patch types [20]. As shown in Fig. 1, existing methods, including VLM-based defenses, struggle to detect deformable adversarial patches.

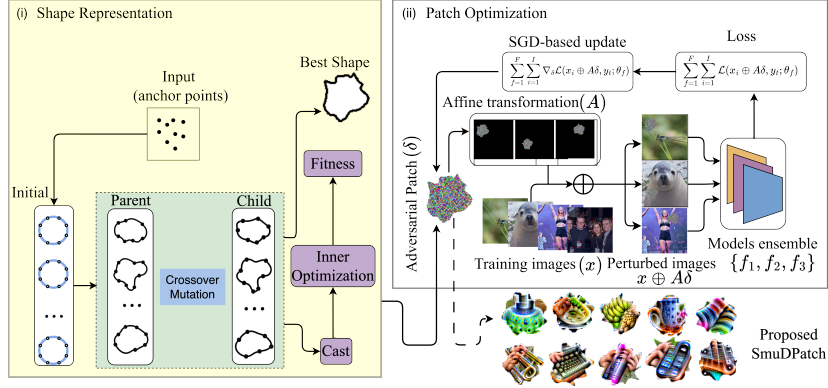


Fig. 2: Overview of the proposed SmuDPatch generation framework. (i) Shape representation: the initial input anchor points form a series of circles. It generates new candidate shapes by applying mutation and crossover to the parent population, producing a child population with diverse, smooth, deformable shapes. The method performs inner optimization to obtain the best deformable shape. (ii) Patch optimization: the best shape adversarial patch applies to the training image through affine transformations and updates the patch parameters using SGD to minimize the loss of an ensemble of models, resulting in robust and transferable adversarial patches, SmuDPatch.

3 Proposed SmuDPatch Deformable Adversarial Patch Generator

This section discusses the proposed method for creating diverse deformable shape adversarial patches to benchmark various DNN image classifiers. The goal of this method is to generate a universal adversarial patch for a chosen target class. The patch is called universal because we apply the same optimized patch to many different images. We begin by defining adversarial patch attacks on image classification, and we perform the optimization in two stages. Unlike fixed shape patches, this method first optimize a smooth deformable shape patch boundary using an evolutionary search over a low-dimensional radial parameterization. Second, once the evolutionary search selects the best shape, gradient descent optimizes the patch texture. Thus, we optimize shape and texture sequentially, not jointly. The findings of joint optimization of shape and texture are provided in the appendix/supplementary.

3.1 Problem Definition

Given a clean image x with corresponding ground truth or label as y for an image classification task, the adversarial patch attack aims to make the image classifier f misclassify an adversarial image x^{adv} .

$$x^{adv} = (1 - \mu) \odot x + \mu \odot \delta, \quad (1)$$

where μ is the mask that is the tensor with the same size as the input image x , whose components are ones where the patch should be applied and zeros elsewhere [21]. δ is the adversarial patch that we compute. $\odot$ represents the element-wise Hadamard product. By introducing an adversarial patch, we aim to manipulate the model’s output. The patch forces a misclassification for untargeted attacks, $\hat{y} \neq y$. The patch directs the model towards a target class y_t for targeted attacks, ensuring $\hat{y} = y_t$. This study focuses on targeted attacks, considering a patch effective if it successfully guides the model’s prediction to an intended class.

3.2 Shape Representation

We first define a shape representation model to yield a universal adversarial patch. Here are the details of initializing smooth deformable shapes for adversarial patches.

1. **Smooth and Deformable Patch (SmuDPatch):** The closest work to ours is by [51]. However, their Differential Evolution-based methods focus solely on shape and typically optimizes the shape on a single visible and infrared image pair. It result in a locally optimal patch shape that fails to generalize across different objects or scenarios in the physical world [31]. Our method represents the patch shape by a smooth closed contour constructed from a set of radial variables. Unlike [7], who use a central point with corresponding rays to form a polygon, we employ multiple anchor points distributed along the patch contour. Moreover, instead of directly optimizing a dense binary mask, we define the contour by N anchor points placed at uniformly spaced angles around the patch center. In our implementation, $N = 25$. Let the patch center be $c = (c_a, c_b)$ and let

$$\mathbf{r} = [r_0, r_1, \dots, r_{N-1}] \quad (2)$$

denote the vector of radial distances. For the i -th anchor point, the angular position is fixed as

$$\theta_i = \frac{2\pi i}{N}, \quad i = 0, 1, \dots, N-1, \quad (3)$$

and the corresponding anchor point is

$$P_i = \begin{bmatrix} c_a + r_i \cos \theta_i \\ c_b + r_i \sin \theta_i \end{bmatrix}. \quad (4)$$

Thus, the radial values $\mathbf{r}$ entirely control the degrees of freedom, while the angular positions remain fixed. This produces a single closed star-convex-like contour around the patch center. To obtain a smooth boundary, we interpolate the anchor points using a centripetal Catmull–Rom spline. This avoids the jagged boundaries that would arise from directly connecting the

anchor points with straight line segments. We denote the resulting smooth contour by μ_{con} . We formulate the spline process as follows:

To generate a curve segment C_i between P_i and P_{i+1} , we use four anchor points: P_{i-1}, P_i, P_{i+1} , and P_{i+2} with a centripetal Catmull-Rom spline function $CCRS(\cdot)$ [48]. This spline function provides several advantages: it prevents loops or self-intersections within a curve segment, avoids cusps, and follows anchor points closely, which helps the curve to be smoother and natural. The formulation for C_i is as follows: $C_i = CCRS(P_{i-1}, P_i, P_{i+1}, P_{i+2})$. When we combine n curve segments $C_0, C_1, \dots, C_{N-1}$, the patch contour μ_{con} can be expressed as:

$$\mu_{con} = \cup_{i=0}^{N-1} C_i,$$

with μ_{con} closed, we define μ as follows:

$$\mu(\alpha) = \begin{cases} 1, & \alpha \text{ inside } \mu_{con} \\ 0, & \alpha \text{ outside } \mu_{con} \end{cases} \quad (5)$$

We can represent the patch's shape by following Eq. (5). The material in the appendix provides detailed formulas for this process.

2. **Area-matched baseline:** We control the patch size so that shape geometry, not area, determines the attack performance. We normalize every candidate shape to a fixed target area. Let $A(\mu_{con})$ denote the polygonal area enclosed by the spline contour. We set the target area to

$$A_{\text{target}} = 2500 \text{ pixels}, \quad (6)$$

which corresponds to $\approx 5\%$ of a 224×224 image. We compute the scaling factor

$$s = \sqrt{\frac{A_{\text{target}}}{A(\mu_{con})}}, \quad (7)$$

and radially scale all contour points around the center so that the shape matches the target area. This normalization allows a fair comparison between different shapes under the same perturbation budget. We also generate patches covering 8% and 10% of the image area (see appendix).

3. **Evolutionary Shape Optimization:** We optimize the patch shape by an evolutionary algorithm operating on the radial vector $\mathbf{r}$. This is necessary because the shape-generation pipeline includes non-differentiable operations such as spline sampling, polygon rasterization, integer rounding, and binary thresholding. Consequently, we do not use standard backpropagation to optimize the shape.

Each individual in the population represents a candidate radial vector:

$$\mathbf{r}^{(k)} = [r_0^{(k)}, r_1^{(k)}, \dots, r_{N-1}^{(k)}].$$

We generate the initial population by perturbing a roughly circular shape:

$$r_i^{(k)} \sim \max(5, r_0 + \epsilon_i), \quad \epsilon_i \sim \mathcal{U}(-10, 10), \quad (8)$$

with $r_0 = 40$ in our implementation.

For each candidate shape, we generate a binary mask $\mu(\mathbf{r}^{(k)})$, and we initialize a temporary random patch texture within that mask. We then optimize this temporary texture for a small number of gradient steps, while keeping the shape fixed. We use the final targeted loss after this short inner optimization to define the fitness of the candidate shape. Hence, the quality of a shape is measured indirectly by how well it supports a quickly optimized adversarial texture.

Let $L_{\text{shape}}(\mathbf{r})$ denote the final inner-loop loss associated with shape $\mathbf{r}$. We define the fitness as

$$F(\mathbf{r}) = \frac{1}{L_{\text{shape}}(\mathbf{r}) + \varepsilon}, \quad (9)$$

where ε is a small positive constant for numerical stability.

At each generation, we rank the population by fitness, retain the top half as parents, and generate the remaining individuals using one-point crossover and Gaussian mutation. For two parent vectors $\mathbf{r}^{(c)}$ and $\mathbf{r}^{(d)}$, crossover produces a child by concatenating a prefix from one parent and a suffix from the other. Mutation then perturbs each radius as

$$r'_i = \max(5, r_i + \eta_i), \quad \eta_i \sim \mathcal{N}(0, \sigma^2), \quad (10)$$

where σ denotes the mutation strength. After a fixed number of generations, we select the highest-fitness individual as the final shape. We set the population size equal to 30 for each generation and use 20 total generations. We have added more details in the appendix material

3.3 Optimization of Adversarial Patches

Once we find the best shape, we fix the corresponding binary mask. As adversarial patches are designed for real-world use, they must remain effective under unavoidable affine transformations, such as rotation, translation, and scaling. Therefore, we include these transformations in the optimization process to ensure effectiveness in physical-world settings. Consequently, we express the loss function to be minimized as:

$$\min_{\delta} \mathbb{E}_{A \sim \mathcal{T}} \left[\sum_{f=1}^F \sum_{i=1}^I \mathcal{L}(x_i \oplus A\delta, y_t; \theta_f) \right] \quad (11)$$

Where δ is the adversarial patch that we compute, x_i is one of image (I) samples of the training data, θ_f is the parameter of the targeted model f , A is an affine transformation randomly sampled from a set of affine transformation $\mathcal{T}$, $\mathcal{L}$ is the cross-entropy loss that quantifies the classification error between the target label and the predicted one and $\oplus$ is a function that applies the patch on the input image. Unlike the approach by [36], we use varying shapes, and by introducing random affine transformations at each epoch, we diverge from [5]. We optimize the objective function in Eq. 11 using gradient descent, iterating through all

models in the ensemble, accumulating gradients, and updating the patch at the end of each epoch. Fig. 2 summarizes our proposed framework to generate smooth and deformable adversarial patches.

4 Proposed SmuDPatch Dataset

This section discusses characteristics of our two large-scale datasets, the ImageNet-Patch and COCO-Patch adversarial datasets. We use them to evaluate the robustness of image classification models and for out-of-distribution adversarial patch detection using defense models. As presented here, the introduction of large-scale, deformable adversarial patch datasets marks a crucial advancement for rigorous benchmarking, facilitating future research into robust universal defense systems. The appendix gives the detailed characteristics of the developed datasets.

4.1 ImageNet SmuDPatch Dataset

We use the validation set of the ImageNet [11] database, containing 50000 images, and select a test subset of 5000 images, aligning with the RobustBench [10] dataset to facilitate direct comparisons of model robustness evaluation against adversarial patch attacks. We create the dataset for optimizing adversarial patches from the remaining validation images, excluding those used for testing. We randomly sample 20 images from various classes, excluding the target class of each attack. For example, if the target class is “cup,” we select one image from each of the 20 other classes among the remaining 999 classes in ImageNet.

We optimize the adversarial patches using an ensemble of three state-of-the-art ImageNet-trained deep neural networks: ViT [12], ResNet50 [16], and MobileNetV2 [40]. Various shape patches are created with a learning rate of 1, based on 20 training samples selected as described earlier. Patches are optimized for 2000 training epochs with cross-entropy loss. Affine transformations during patch optimization has been applied, constraining rotations to $\pm \frac{\pi}{8}$ and translations to ± 68 pixels from the image center to avoid boundary proximity. Ultimately, we obtain 20 patches for two different shapes and target 10 ImageNet classes. In brief, we generate two patch shapes (deformable, circle) for each class (“soap dispenser (patch-0),” “cornet (patch-1),” “plate (patch-2),” “banana (patch-3),” “cup (patch-4),” “typewriter keyboard (patch-5),” “electric guitar (patch-6),” “hair spray (patch-7),” “sock (patch-8),” “cellular phone (patch-9)”). These patches are then applied to each of the 5000 images, resulting in a dataset of 100000 ($5000 \times 10 \times 2$) perturbed images.

4.2 Out-of-Distribution SmuDPatch Dataset

Interestingly, most of the work that generates adversarial patches focuses on only one dataset, leaving no space for the development of an out-of-distribution adversarial patch detection. Therefore, the proposed study aims to address this

gap by introducing another large-scale novel adversarial patch dataset that can be used specifically for testing the adversarial patch detectors. By following the procedure outlined by Ojaswee et al. [35], we are introducing an adversarial dataset focusing on physical adversarial patches using the COCO [29] dataset (Common Objects in Context). We attack images with 20 proposed physical adversarial patches. *We select 2000 clean images from the COCO validation set and apply 20 different adversarial patches to each, generating 40,000 adversarial patch images in total.* Another set of 2000 clean images from the COCO validation set serves as a clean subset. We divide clean and patched images into training and testing sets in a 3:2 ratio, respectively. For example, we divide 2000 clean images of COCO and 2000 images with a single patch into 800 test images and 1200 train images. The primary aim of this dataset is to evaluate the generalizability of the defense algorithm. We implement the proposed datasets and image classification models in a PyTorch environment with an A6000 GPU with 48 GB of memory.

5 Experimental Results and Analysis

To justify the stealthiness of the proposed smooth and deformable adversarial patches, we evaluated them against several state-of-the-art image classification models. In this subsection, we first present the evaluation metrics and target models that we use to assess the effectiveness of the proposed smooth and deformable adversarial patches. Later, extensive evaluation of state-of-the-art image classification models demonstrates their deceptive nature.

5.1 Evaluation Metrics and Target Models

Following the literature [36], we evaluate the effectiveness of the proposed patches using the ImageNet patch datasets using three metrics: (i) clean accuracy: performance of a model on clean (unattacked) images, (ii) robust accuracy: accuracy of a model in the presence of an adversarial patch, and (iii) attack success rate (ASR): success rate of the attack in fooling a model.

To rigorously evaluate the superiority of our proposed adversarial method, we conducted comprehensive experiments involving 15 state-of-the-art deep neural networks (DNNs) trained on the ImageNet dataset. We organize these models into five groups for our analysis: *ensemble (ENS)*, *standard*, *adv-robust (ADV)*, *augmentation (Augm.)*, and *more-data*. The ensemble group refers to the models trained in combination, and we evaluated the robustness of each model individually. The ensemble models includes ResNet50 [16], MobileNetV2 [40], and ViT [12]. The standard group comprises five models trained individually: ResNet18 [16], AlexNet [22], GoogLeNet [45], SqueezeNet [19], and ConvNeXt [30]. The adv-robust group consists of three adversarially robust models available on RobustBench³: models by Salman et al. [13, 39], and Wong et al. [53]. We also considered four models from the ImageNet testbed repository proposed by

³ <https://github.com/RobustBench/robustbench>

Table 1: The clean accuracy (Clean Acc. (%)), robust accuracy (Robust (%)), and attack success rate (ASR (%)) of proposed shapes for different groups of models. The proposed deformable (SmuDPatch) patches drastically reduce the performance of models belonging to different training groups.

Metric	ENS			Standard					Adv			Augm.		More-Data	
	MobileNetV2	ResNet50	ViT	ResNet18	AlexNet	GoogleNet	SqueezeNet	ConvNext-L	[13]	[39]	[53]	[17]	[14]	[58]-(a)	[58]-(b)
Clean Acc. (%)	71.9	76.7	80.9	69.7	56.2	69.8	58.0	83.6	62.5	64.1	53.5	76.8	68.5	81.1	82.1
DAPatch Robust ↓ (%)	60.5	66.9	79.0	59.6	45.4	60.1	48.3	82.1	56.4	56.4	47.0	73.6	64.2	75.6	77.3
DAPatch ASR ↑ (%)	1.4	3.5	0.8	0.6	0.2	0.8	0.6	1.0	0.1	0.1	0.1	0.3	0.4	1.8	1.7
Circle Robust ↓ (%)	46.3	49.7	74.2	53.7	38.9	55.3	43.2	81.0	51.9	50.9	41.1	64.1	56.6	67.4	69.3
Circle ASR ↑ (%)	17.9	27.9	7.4	2.3	0.4	4.0	2.0	2.3	0.2	0.2	0.2	5.8	1.4	10.5	8.7
SmuDPatch Robust ↓ (%)	31.6	28.5	65.5	45.5	31.6	46.0	34.4	76.0	49.2	47.7	38.4	52.2	50.6	57.6	56.8
SmuDPatch ASR ↑ (%)	44.2	62.1	20.4	12.0	1.3	18.4	8.3	10.3	0.6	0.3	0.2	24.2	7.1	23.5	26.9

Taori et al. [47] to analyze the effect of non-adversarial augmentation techniques and training on large augmentation. Specifically, we selected two models trained for robustness to common image perturbations and corruptions: models by Hendrycks et al. [17] and Engstrom et al. [14]. We group these models under the augmentation strategy. Additionally, we selected two models proposed by Yalniz et al. [58], trained on datasets with substantially more training data than the standard ImageNet training set. We group these models under more data. We want to highlight that the ensemble models are part of the **seen (or known) models** that we used to optimize our adversarial patches. The remaining **unseen or unknown models** showcase the transferability of the proposed deformable adversarial patches.

5.2 Attack Effectiveness of the Proposed Patches

Table 1 shows the success of each of the proposed adversarial patch attacks in models, whether it is adversarially robust or trained with heavy augmentation. The literature observed that the ensemble models are one of the most vigorous defense models to be fooled by adversarial perturbations [64]. However, the results show that the proposed attack drastically reduces their robustness accuracy and yields a high attack success rate for each model used in the ensemble. The patches’ success is not only in pure convolutional models but also in transformer models. For example, a ViT model yields a clean accuracy of 80.9%, which drops to 65.5% under the effect of deformable patches. As mentioned earlier, the ensemble models are more resilient than the standard models, which can also be seen from the fooling rate of the standard models. The results show that the standard model suffers a significant drop in recognition accuracy across the patch shapes. For example, the recognition performance of GoogLeNet drops from 69.8% to 46.0% when we use deformable patches to attack it. Other than ensemble models, adversarially robust models that claim their robustness against adversarial attacks are equally vulnerable to the proposed adversarial patches. For example, the adversarially robust model by Wong et al. [53] suffers a drop of 15.1% and 12.4% when we perform the attacks with deformable and circle patches, respectively. The drops show high transferability of patches.

Augmentation models and more data models that utilize heavy data labels (such as *billions* of images) are also not robust against the proposed deformable

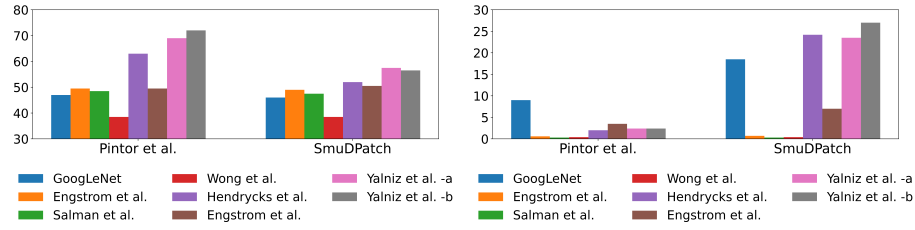


Fig. 3: Comparison of the proposed SmuDPatch with Pintor et al. [36] in terms of robust accuracy (left) and ASR (right). The results shows the proposed patches outperform the recent SOTA universal patches of [36] both in terms of robust accuracy (lower the better) and ASR (higher the better).

adversarial patches. While these models yield significantly better capacity (clean accuracy) than standard models, their vulnerability against adversarial patches raises concerns about spending energy sources in training such models. For example, the model by Hendrycks et al. [17] achieves a robust accuracy of 52.2%, drastically lower than its accuracy on clean images (76.8%). This indicates that while data augmentation techniques enhance robustness against minute adversarial attacks or natural noises, they are less effective against the proposed deformable and circular adversarial patches. The results show a similar vulnerability for the models grouped under more data categories. In brief, our extensive experimentation shows that each category of existing SOTA image classification models, which broadly cover almost every form of network configuration, is not genuinely secure even when we do not use them in the attack generation. Further, they are more vulnerable to the proposed deformable patches than traditional fixed-shape patches. We assert that understanding their underexplored sensitivity can help boost their robustness against adversarial patches.

5.3 Physical-World Attack Success

Inspired by implementing adversarial patches in the physical world [36], we printed and applied our patches to real-world objects. While Pintor et al. [36] used the three objects, we have extended our evaluation by selecting four everyday items: an orange, a knife, a computer mouse, and a remote, and captured images of them with our ten SmuDPatch variants applied under three different roto-translations, resulting in a dataset of 120 images. We then evaluated the robustness of standard models, ResNet50, MobileNetV2, and ViT, by reporting robust accuracy and attack success rate (ASR). Our findings show a notable drop in robustness under physical attacks: ResNet50’s accuracy dropped from 66.6% to 2.5%, MobileNetV2 from 53.33% to 4.17%, and ViT from 74.17% to 31.67%. The higher ASRs of ResNet50, MobileNetV2, and ViT further demonstrate the effectiveness and transferability of SmuDPatch in the real world.

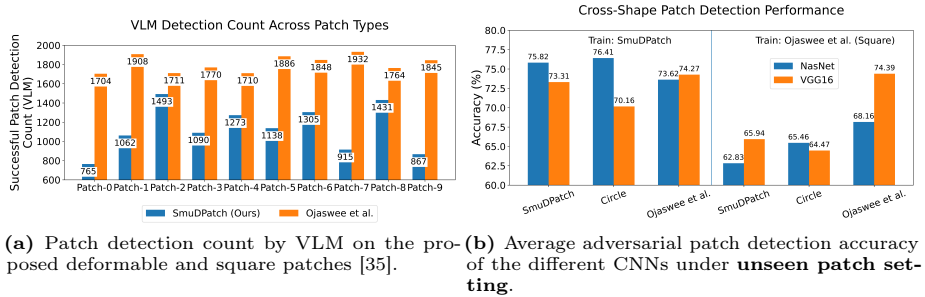


Fig. 4: Adversarial patch detection using VLM (left) and DNNs (right).

5.4 Comparison with Existing Deformable and Fixed Shape Patches

We have compared the effectiveness of the proposed patches in two terms: (i) attack success rate compared to fixed and deformable shape patches, and (ii) complexity in their detection. As shown in Fig. 3, the proposed adversarial patches significantly outperform the recently proposed fixed-shaped patches [36] in terms of robust accuracy and attack success rate. Our proposed patches yield superior performance to the state-of-the-art (SOTA) deformable patch approach, namely DAPatch [7], which utilizes a gradient-based deformable shape optimization. Our method consistently outperformed DAPatch as shown in Table 1, yielding lower robust accuracy (28.5% vs. 66.9%) and higher attack success rates (62.1% vs. 3.5%) on ResNet50. DAPatch fails drastically in the universal attack setting because its method optimizes patches locally for each specific image, preventing the learned shape and texture from generalizing effectively across unseen images.

5.5 Connection to Adversarial Defense

While the defense is not the core philosophy of this paper, to connect the proposed attack with existing adversarial attack detection, we have demonstrated the preliminary efficacy of defense approaches. We have assessed a detection model, NAPGuard [55], and observed a mean Average Precision (mAP) of 45.97% when the SmuDPatch comes for evaluation, which is 14.1% lower than the square patches [35, 36] and 29.9% lower than the circle patch (for patch-wise results see appendix). To further demonstrate the resilience of our patches, we evaluated their detection against the most recent AdvPatchXAI [24] defense and observed lower detectability than the fixed-shape patches. For example, detecting our deformable patch-8 is 14% lower than fixed-shape patch-8 [36].

We also evaluate the ability of the VLM LLaVaNext [27] to detect physical adversarial patches by prompting it with a generic query, “Is this image attacked by any physical adversarial patch?” without providing any patch-specific details. We measure detection accuracy by counting responses that begin with “Yes,” using 2,000 randomly selected images per patch shape and class. The results shown in Fig. 4a indicate that LLaVaNext detects significantly fewer of our proposed

deformable patches (765 detections) compared to prior patches such as patch-0 of Pintor et al. [36] (1,704 detections), indicating that our deformable patches are harder to identify.

Apart from the LLaVaNext, we additionally evaluated a SOTA segmentation model, SAM-3 [6], on 8,000 patched images per shape family (800 images per patch). We use the same text prompt, “adversarial patch”, for both square and SmuDPatch images. As reported in Table 2 SAM-3 localizes square patches much more reliably than deformable patches: mIoU drops from 39.54% on square patches to 9.41% on SmuDPatch, and successful detections reduce from 3212/8000 to 778/8000. This confirms that the difficulty is not limited to VLM-based recognition; even promptable segmentation models struggle to localize our smooth deformable patches.

Table 2: SAM 3 segmentation using the prompt “adversarial patch”.

Shape	mIoU	mDice	mBF1	IoU>0.5	Dets.
Square	39.54	39.84	40.14	41.6	3212/8000
SmuDPatch	9.41	9.56	9.68	10.5	778/8000

Further, motivated by the success of NASNet and VGG16 in the literature [25, 35], we have trained these networks, along with a pre-trained vision language model (VLM), to comprehensively benchmark the out-of-distribution (OOD) patch detection using our patches applied on the COCO dataset. Fig. 4b presents the adversarial attack detection accuracy under the ‘**unseen patch detection or out-of-distribution (OOD)**’ setting, demonstrating the efficacy and limitation of existing detectors on deformable patches. (see the appendix for more details).

5.6 Extension Beyond Classification

Classification does not fully capture the threat of physical patches. Following the same object-detection protocol of Kumar and Agarwal [25], we conducted additional experiments on SOTA detectors and report COCO-style mAP. SmuD-Patch consistently degrades detection performance across Faster R-CNN and recent YOLO variants: the average mAP drops from 41.86 on clean images to 36.63 under SmuDPatch, with the largest drop on YOLO-v11 (45.66→35.09). These results show that the proposed deformable patches transfer beyond image classification and also affect object detection even when they are not trained for this task.

5.7 Naturalness-Enhanced Differentiable Variant of SmuDPatch

While the original SmuDPatch is designed to demonstrate the effectiveness of smooth deformable shape optimization under an evolutionary search framework,

Table 3: Object detection under SmuDPatch. We report mAP on clean and attacked images.

Metric	Faster R-CNN	YOLO-v8	YOLO-v9	YOLO-v10	YOLO-v11	Average
Clean mAP	36.88	36.69	51.87	38.22	45.66	41.86
SmuDPatch mAP	32.50	33.25	47.47	34.86	35.09	36.63

we further explore a differentiable extension aimed at improving patch naturalness in the image. This extension introduces a soft mask instead of hard binary mask, shape smoothness regularization, joint shape-texture optimization instead of two-stage evolutionary pipeline, and Laplacian pyramid blending for smoother image integration. Empirically, this variant yields improved robust accuracy reduction and higher attack success rate in some cases on the evaluated ensemble and standard models. We therefore view it as a complementary extension that enhances the original framework rather than invalidating the base methodology. A detailed methodology and discussion of qualitative and quantitative results has been added in the appendix material.

6 Conclusion

This research comprehensively demonstrates the need for adversarial patches of different shapes to get a universal defense against adversarial patches. The extensive bias of the existing adversarial patch generation algorithms towards fixed-shape patches leaves a substantial gap in detecting adaptive patches, even by VLM or DNN architectures, along with recent SOTA patch detectors. In response, we have presented a unified adversarial patch generation architecture to generate various shapes of patches, including deformable and circular patches. The proposed patches demonstrate the vulnerability of classification models and pave the way for better adversarial patch defense. In the future, we aim to address the limitation of the limited attack successes of the proposed patches, although they are superior to the existing attacks. This higher attack success and effectiveness in defending patches can help improve the unified defense.

Acknowledgements

V. Kumar is partially supported through the Visvesvaraya PhD Fellowship. We also thank the reviewers for the valuable comments and suggestions, which helped improve this paper.

References

1. Agarwal, A., Ratha, N., Vatsa, M., Singh, R.: Crafting adversarial perturbations via transformed image component swapping. *IEEE Transactions on Image Processing* **31**, 7338–7349 (2022)

2. Agarwal, A., Singh, R., Vatsa, M., Ratha, N.: Ibattack: Being cautious about data labels. *IEEE Transactions on Artificial Intelligence* **4**(6), 1484–1493 (2023)
3. Agarwal, A., Vatsa, M., Singh, R.: On adversarial robustness of face presentation attack detection algorithms. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*. pp. 6015–6023 (October 2025)
4. Agarwal, A., Vatsa, M., Singh, R., Ratha, N.: Corruption depth: Analysis of dnn depth for misclassification. *Neural Networks* **172**, 106013 (2024)
5. Brown, T.B., Mané, D., Roy, A., Abadi, M., Gilmer, J.: Adversarial patch. *arXiv preprint arXiv:1712.09665* (2017)
6. Carion, N., Gustafson, L., Hu, Y.T., Debnath, S., Hu, R., Suris, D., Ryali, C., Alwala, K.V., Khedr, H., Huang, A., et al.: Sam 3: Segment anything with concepts. *arXiv preprint arXiv:2511.16719* (2025)
7. Chen, Z., Li, B., Wu, S., Xu, J., Ding, S., Zhang, W.: Shape matters: deformable patch attack. In: *European Conference on Computer Vision*. pp. 529–548. Springer (2022)
8. Chen, Z., Li, B., Xu, J., Wu, S., Ding, S., Zhang, W.: Towards practical certifiable patch defense with vision transformer. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 15148–15158 (2022)
9. Cheng, Y., Wei, F., Bao, J., Chen, D., Zhang, W.: Adpl: Adaptive dual path learning for domain adaptation of semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **45**(8), 9339–9356 (2023)
10. Croce, F., Andriushchenko, M., Schwag, V., Debnedetti, E., Flammarion, N., Chiang, M., Mittal, P., Hein, M.: Robustbench: a standardized adversarial robustness benchmark. *arXiv preprint arXiv:2010.09670* (2020)
11. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: *IEEE CVPR*. pp. 248–255 (2009)
12. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929* (2020)
13. Engstrom, L., Ilyas, A., Salman, H., Santurkar, S., Tsipras, D.: Robustness (python library) (2019), <https://github.com/MadryLab/robustness>
14. Engstrom, L., Tran, B., Tsipras, D., Schmidt, L., Madry, A.: Exploring the landscape of spatial robustness. In: *International conference on machine learning*. pp. 1802–1811. PMLR (2019)
15. Gu, Z., Chen, Y., Yao, T., Ding, S., Li, J., Huang, F., Ma, L.: Spatiotemporal inconsistency learning for deepfake video detection. In: *29th ACM international conference on multimedia*. pp. 3473–3481 (2021)
16. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *IEEE conference on computer vision and pattern recognition*. pp. 770–778 (2016)
17. Hendrycks, D., Basart, S., Mu, N., Kadavath, S., Wang, F., Dorundo, E., Desai, R., Zhu, T., Parajuli, S., Guo, M., et al.: The many faces of robustness: A critical analysis of out-of-distribution generalization. In: *IEEE/CVF international conference on computer vision*. pp. 8340–8349 (2021)
18. Huang, J.J., Wang, Z., Liu, T., Luo, W., Chen, Z., Zhao, W., Wang, M.: Dempaa: Deployable multi-mini-patch adversarial attack for remote sensing image classification. *IEEE Transactions on Geoscience and Remote Sensing* (2024)
19. Iandola, F.N.: Squeezenet: Alexnet-level accuracy with 50x fewer parameters and < 0.5 mb model size. *arXiv preprint arXiv:1602.07360* (2016)

20. Jing, L., Wang, R., Ren, W., Dong, X., Zou, C.: Pad: Patch-agnostic defense against adversarial patch attacks. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 24472–24481 (2024)
21. Karmon, D., Zoran, D., Goldberg, Y.: Lavan: Localized and visible adversarial noise. In: International conference on machine learning. pp. 2507–2515. PMLR (2018)
22. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems* **25** (2012)
23. Kumar, D., Sharma, A., Narayan, A.: Attacking cnns in histopathology with snap: Sporadic and naturalistic adversarial patches (student abstract). In: AAAI Conference on Artificial Intelligence. vol. 38, pp. 23550–23551 (2024)
24. Kumar, V., Agarwal, A.: A unified, resilient, and explainable adversarial patch detector. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 30387–30397 (June 2025)
25. Kumar, V., Agarwal, A.: The unseen adversaries: Robust and generalized defense against adversarial patches. In: The 29th International Conference on Artificial Intelligence and Statistics (2026)
26. Kumar, V., Shukla, S., Agarwal, A.: Robustness benchmarking of convolutional and transformer architectures for image classification. *IEEE Transactions on Big Data* **11**(6), 3330–3341 (2025)
27. Li, B., Zhang, H., Zhang, K., Guo, D., Zhang, Y., Zhang, R., Li, F., Liu, Z., Li, C.: Llava-next: What else influences visual instruction tuning beyond data? (May 2024), <https://llava-vl.github.io/blog/2024-05-25-llava-next-ablations/>
28. Liang, J., Yi, R., Chen, J., Nie, Y., Zhang, H.: Securing autonomous vehicles visual perception: Adversarial patch attack and defense schemes with experimental validations. *IEEE Transactions on Intelligent Vehicles* (2024)
29. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV. pp. 740–755 (2014)
30. Liu, Z., Mao, H., Wu, C.Y., Feichtenhofer, C., Darrell, T., Xie, S.: A convnet for the 2020s. In: IEEE/CVF conference on computer vision and pattern recognition. pp. 11976–11986 (2022)
31. Long, J., Yao, W., Jiang, T., Hou, J., Jia, S., Wu, J., Zhang, X., Zheng, X., Ma, C.: Cdupatch: Color-driven universal adversarial patch attack for dual-modal visible-infrared detectors. In: Proceedings of the 33rd ACM International Conference on Multimedia. pp. 1462–1470 (2025)
32. Lu, D., Wang, Z., Wang, T., Guan, W., Gao, H., Zheng, F.: Set-level guidance attack: Boosting adversarial transferability of vision-language pre-training models. In: IEEE/CVF International Conference on Computer Vision. pp. 102–111 (2023)
33. Luo, H., Gu, J., Liu, F., Torr, P.: An image is worth 1000 lies: Transferability of adversarial images across prompts on vision-language models. In: The Twelfth International Conference on Learning Representations (2023)
34. Madry, A.: Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083 (2017)
35. Ojaswee, O., Agarwal, A., Ratha, N.: Benchmarking image classifiers for physical out-of-distribution examples detection. In: IEEE/CVF International Conference on Computer Vision, Workshop. pp. 4427–4435 (2023)
36. Pintor, M., Angioni, D., Sotgiu, A., Demetrio, L., Demontis, A., Biggio, B., Roli, F.: Imagenet-patch: A dataset for benchmarking machine learning robustness against adversarial patches. *Pattern Recognition* **134**, 109064 (2023)

37. Qi, X., Huang, K., Panda, A., Henderson, P., Wang, M., Mittal, P.: Visual adversarial examples jailbreak aligned large language models. In: AAAI Conference on Artificial Intelligence. vol. 38, pp. 21527–21536 (2024)
38. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021)
39. Salman, H., Ilyas, A., Engstrom, L., Kapoor, A., Madry, A.: Do adversarially robust imagenet models transfer better? *Advances in Neural Information Processing Systems* **33**, 3533–3545 (2020)
40. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: IEEE conference on computer vision and pattern recognition. pp. 4510–4520 (2018)
41. Scheurer, E., Schmalfluss, J., Lis, A., Bruhn, A.: Detection defenses: An empty promise against adversarial patch attacks on optical flow. In: IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 6489–6498 (2024)
42. Sharif, M., Bhagavatula, S., Bauer, L., Reiter, M.K.: Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In: 2016 acm sigsac conference on computer and communications security. pp. 1528–1540 (2016)
43. Shayegeani, E., Mamun, M.A.A., Fu, Y., Zaree, P., Dong, Y., Abu-Ghazaleh, N.: Survey of vulnerabilities in large language models revealed by adversarial attacks. *arXiv preprint arXiv:2310.10844* (2023)
44. Strack, L., Waseda, F., Nguyen, H.H., Zheng, Y., Echizen, I.: Defending against physical adversarial patch attacks on infrared human detection. In: 2024 IEEE International Conference on Image Processing (ICIP). pp. 3896–3902. IEEE (2024)
45. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: IEEE conference on computer vision and pattern recognition. pp. 1–9 (2015)
46. Tan, W., Li, Y., Zhao, C., Liu, Z., Pan, Q.: Doepatch: Dynamically optimized ensemble model for adversarial patches generation. *IEEE Transactions on Information Forensics and Security* (2024)
47. Taori, R., Dave, A., Shankar, V., Carlini, N., Recht, B., Schmidt, L.: Measuring robustness to natural distribution shifts in image classification. *Advances in Neural Information Processing Systems* **33**, 18583–18599 (2020)
48. Twigg, C.: Catmull-rom splines. *Computer* **41**(6), 4–6 (2003)
49. Wei, H., Tang, H., Jia, X., Wang, Z., Yu, H., Li, Z., Satoh, S., Van Gool, L., Wang, Z.: Physical adversarial attack meets computer vision: A decade survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2024)
50. Wei, H., Wang, Z., Jia, X., Zheng, Y., Tang, H., Satoh, S., Wang, Z.: Hotcold block: Fooling thermal infrared detectors with a novel wearable design. In: Proceedings of the AAAI conference on artificial intelligence. vol. 37, pp. 15233–15241 (2023)
51. Wei, X., Huang, Y., Sun, Y., Yu, J.: Unified adversarial patch for cross-modal attacks in the physical world. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4445–4454 (2023)
52. Wei, X., Yu, J., Huang, Y.: Physically adversarial infrared patches with learnable shapes and locations. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12334–12342 (2023)
53. Wong, E., Rice, L., Kolter, J.Z.: Fast is better than free: Revisiting adversarial training. *arXiv preprint arXiv:2001.03994* (2020)
54. Wu, C.H., Koh, J.Y., Salakhutdinov, R., Fried, D., Raghunathan, A.: Adversarial attacks on multimodal agents. *arXiv preprint arXiv:2406.12814* (2024)

55. Wu, S., Wang, J., Zhao, J., Wang, Y., Liu, X.: Napguard: Towards detecting naturalistic adversarial patches. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 24367–24376 (2024)
56. Xu, K., Zhang, G., Liu, S., Fan, Q., Sun, M., Chen, H., Chen, P.Y., Wang, Y., Lin, X.: Adversarial t-shirt! evading person detectors in a physical world. In: Computer vision–ECCV 2020: 16th European conference, glasgow, UK, August 23–28, 2020, proceedings, part v 16. pp. 665–681. Springer (2020)
57. Xu, W., Chen, K., Gao, Z., Wei, Z., Chen, J., Jiang, Y.G.: Highly transferable diffusion-based unrestricted adversarial attack on pre-trained vision-language models. In: 32nd ACM International Conference on Multimedia. pp. 748–757 (2024)
58. Yalniz, I.Z., Jégou, H., Chen, K., Paluri, M., Mahajan, D.: Billion-scale semi-supervised learning for image classification. arXiv preprint arXiv:1905.00546 (2019)
59. Yin, Z., Ye, M., Zhang, T., Du, T., Zhu, J., Liu, H., Chen, J., Wang, T., Ma, F.: Vllattack: Multimodal adversarial attacks on vision-language tasks via pre-trained models. *Advances in Neural Information Processing Systems* **36** (2024)
60. Zhang, J., Yi, Q., Sang, J.: Towards adversarial attack on vision-language pre-training models. In: Proceedings of the 30th ACM International Conference on Multimedia. pp. 5005–5013 (2022)
61. Zhao, Y., Pang, T., Du, C., Yang, X., Li, C., Cheung, N.M.M., Lin, M.: On evaluating adversarial robustness of large vision-language models. *Advances in Neural Information Processing Systems* **36** (2024)
62. Zhou, D., Liu, T., Han, B., Wang, N., Peng, C., Gao, X.: Towards defending against adversarial examples via attack-invariant features. In: International conference on machine learning. pp. 12835–12845. PMLR (2021)
63. Zhou, Q., Zhang, K.Y., Yao, T., Yi, R., Ding, S., Ma, L.: Adaptive mixture of experts learning for generalizable face anti-spoofing. In: 30th ACM international conference on multimedia. pp. 6009–6018 (2022)
64. Zhou, W., Hou, X., Chen, Y., Tang, M., Huang, X., Gan, X., Yang, Y.: Transferable adversarial perturbations. In: European Conference on Computer Vision. pp. 452–467 (2018)
65. Zhou, Z., Hu, S., Li, M., Zhang, H., Zhang, Y., Jin, H.: Advclip: Downstream-agnostic adversarial examples in multimodal contrastive learning. In: 31st ACM International Conference on Multimedia. pp. 6311–6320 (2023)
66. Zhu, X., Liu, Y., Hu, Z., Li, J., Hu, X.: Infrared adversarial car stickers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 24284–24293 (2024)