CAPAA: Classifier-Agnostic Projector-Based Adversarial Attack

Zhan Li [1,*], *Mingyu Zhao* [1,2,*], *Xin Dong* [1], *Haibin Ling* [3], *Bingyao Huang* [1,†]
[1]Southwest University, China [2]Rutgers University, USA [3]Stony Brook University, USA

***Abstract*—Projector-based adversarial attack aims to project carefully designed light patterns (i.e., adversarial projections) onto scenes to deceive deep image classifiers. It has potential applications in privacy protection and the development of more robust classifiers. However, existing approaches primarily focus on individual classifiers and fixed camera poses, often neglecting the complexities of multi-classifier systems and scenarios with varying camera poses. This limitation reduces their effectiveness when introducing new classifiers or camera poses. In this paper, we introduce Classifier-Agnostic Projector-Based Adversarial Attack (CAPAA) to address these issues. First, we develop a novel classifier-agnostic adversarial loss and optimization framework that aggregates adversarial and stealthiness loss gradients from multiple classifiers. Then, we propose an attention-based gradient weighting mechanism that concentrates perturbations on regions of high classification activation, thereby improving the robustness of adversarial projections when applied to scenes with varying camera poses. Our extensive experimental evaluations demonstrate that CAPAA achieves both a higher attack success rate and greater stealthiness compared to existing baselines. Codes are available at: https://github.com/ZhanLiQxQ/CAPAA.**

***Index Terms*—Physical adversarial attack, privacy, projector**

I. Introduction

In multimedia security, adversarial attacks have emerged as a valuable approach to protect privacy and prevent the misuse of recognition systems. The development of such attacks has progressed from traditional methods like the Fast Gradient Sign Method (FGSM) [1] to more sophisticated methods, such as attention-based [2] and universal attacks [3]. Although these methods have made significant progress, they face challenges in real-world applications. As a result, researchers are increasingly exploring physical attacks—adversarial strategies that manipulate real-world objects or environments to deceive machine learning models, particularly in computer vision systems [4]. An example is the attachment of special markers or stickers to objects [5].

Projector-based attacks are a form of physical attacks that deceive classifiers by manipulating illumination conditions without direct physical contact, as illustrated in Fig. 1 **(a)**. For instance, OPAD [7] exploits the optical interactions between projectors and cameras to execute attacks in real-world settings.

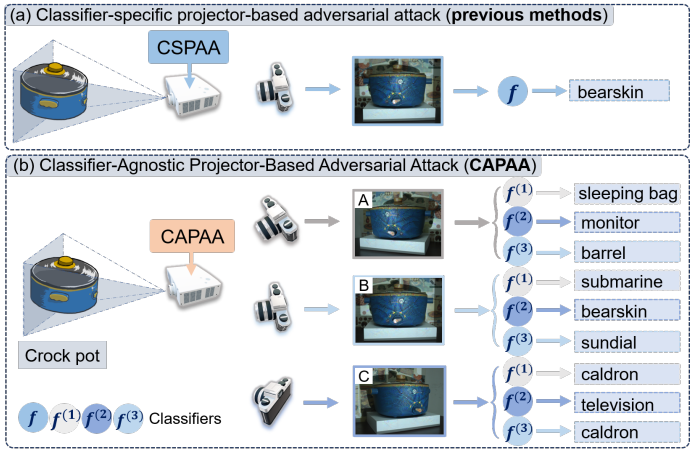

Fig. 1. **(a)** *Classifier-specific* projector-based adversarial attack (CSPAA), aims to deceive a specific classifier under a specific camera capture pose by projecting adversarial light patterns. **(b)** *Classifier-Agnostic* Projector-Based Adversarial Attack (CAPAA) fools multiple classifiers simultaneously and is robust to camera pose changes. A real **Crock pot** (one of the ImageNet [6] classes) was placed in the scene, after projecting our CAPAA-generated adversarial light pattern, the camera-captured scene was misclassified by the three classifiers, such that their output labels were not **Crock pot**.

A key challenge for such attacks is achieving sufficient stealthiness, which is critical for their practical effectiveness. While methods like adversarial color projection [8] have been proposed, many struggle with this aspect. Recent metrics like hiPAA [9] provide a comprehensive evaluation framework by considering multiple factors, including effectiveness, robustness, and stealthiness.

While SPAA [10] improves stealthiness and robustness by modeling the project-and-capture process with a neural network, it remains limited to single-classifier scenarios with fixed camera poses. This restriction is particularly problematic given the growing use of ensemble classifiers [11], as projector-based attacks optimized for a single classifier often fail to transfer effectively. Moreover, even minor camera pose perturbations can significantly degrade attack performance, a vulnerability that current pose-specific methods cannot easily overcome.

To overcome these challenges, we propose CAPAA (Classifier-Agnostic Projector-Based Adversarial Attack), a method designed to enhance attack robustness across various classifiers and camera poses. Specifically, for classifier-

*These authors contributed equally.
Zhan Li and Xin Dong are with Southwest University. E-mail: {lz20020722, dongxin12345}@email.swu.edu.cn.
Mingyu Zhao is with Rutgers University. Work partly done during internship with Southwest University. E-mail: mz751@scarletmail.rutgers.edu.
Haibin Ling is with Dept. of Computer Science, Stony Brook University. E-mail: hling@cs.stonybrook.edu.
†Bingyao Huang is the corresponding author. E-mail: bhuang@swu.edu.cn.

agnostic scenarios, CAPAA introduces a novel multi-objective loss function that enables joint attacks across multiple classifiers. Additionally, we incorporate attention-driven gradient weighting, which focuses subtle light perturbations on regions with high classification activation. These non-trivial designs improve the robustness and stealthiness of the attack.

Our contributions are summarized as follows:

- To our best knowledge, CAPAA is the first classifier-agnostic, projector-based adversarial attack approach.
- We introduce a new classifier-agnostic adversarial loss and optimization framework that aggregates adversarial and stealthiness loss gradients from multiple classifiers, allowing for more effective and flexible projector-based attacks across different classifiers.
- We propose an attention-based gradient weighting mechanism that focuses perturbations on regions of high classification activation, enhancing the robustness of adversarial projections even when camera pose changes.
- Experimental evaluation across 10 setups and 7 camera poses demonstrates that CAPAA outperforms existing methods in terms of both stealthiness and success rates.

II. Method

A. Problem formulation

Adversarial attacks. Let f be an image classifier that maps an image I to a vector of N-class probabilities, $f(I) \in [0,1]^N$, where $f_i(I)$ is the probability of the i-th class. The goal of adversarial attack is to perturb the input image with almost imperceptible noise δ, such that the classifier predicted class $\hat{y}$ either matches a target label y_t (targeted attack) or differs from the true label y (untargeted attack):

$$\hat{y} = \underset{i}{\operatorname{argmax}}\, f_i(I+\delta) \begin{cases} = y_t, & \text{targeted} \\ \neq y, & \text{untargeted} \end{cases}$$
$$\text{subject to} \quad \mathcal{D}(I, I+\delta) < \epsilon. \tag{1}$$

The function $\mathcal{D}$ measures image similarity, and is usually used to control the stealthiness of adversarial attack with a small threshold $\epsilon (\epsilon > 0)$.

Projector-based adversarial attacks. Extending Eqn. 1 to the physical world that uses a projector to alter the light condition, and denote the physical scene as s, denote the projector's projection process, and the camera's capture process as π_p and π_c, respectively. Then, given an input image x, the projected light of the projector can be expressed as $\pi_p(x)$. In a specific camera pose γ, the scene captured by the camera under projected light can be represented as: $I_{x,\gamma} = \pi_c(\pi_p(x), s, \gamma)$. For simplicity, we define the composite project-and-capture process as: $\pi(.) = \pi_c(\pi_p(.), s, \gamma)$, and we have $I_{x,\gamma} = \pi(x, \gamma)$.

Projector-based adversarial attacks aim to generate an adversarial image/pattern x' as projector input, such that when projected to the physical scene and captured as $I_{x',\gamma}$, it causes classifier f to misclassify the scene:

$$\hat{y} = \underset{i}{\operatorname{argmax}}\, f_i(I_{x',\gamma}) \begin{cases} = y_t, & \text{targeted} \\ \neq y, & \text{untargeted} \end{cases}$$
$$\text{subject to} \quad \mathcal{D}(I_{x',\gamma}, I_{x_0,\gamma}) < \epsilon, \tag{2}$$

where $I_{x_0,\gamma}$ is the camera-captured scene illuminated by gray light x_0, i.e., without adversarial projection. Previous **classifier-specific** methods [7], [10] are based on the formulation in Eqn. 2. Although straightforward, they may fail when applied to other classifiers, because the adversarial projection is generated using feedback from a specific classifier. Furthermore, as adversarial projections may become occluded, they may also fail when the camera pose γ changes.

B. Classifier-Agnostic Projector-Based Adversarial Attack (CAPAA)

To address the issues above, we propose CAPAA to generate adversarial projection x' that can perform *classifier-agnostic* attack, and still be robust when camera pose changes:

$$\forall f^{(k)} \in \{f^{(1)}, f^{(2)}, ..., f^{(n)}\}$$
$$\hat{y}^{(k)} = \underset{i}{\operatorname{argmax}}\, f_i^{(k)}(I_{x',\gamma}) \begin{cases} = y_t, & \text{targeted} \\ \neq y, & \text{untargeted} \end{cases}$$
$$\text{subject to} \quad \mathcal{D}(I_{x',\gamma}, I_{x_0,\gamma}) < \epsilon, \tag{3}$$

where $f^{(k)} \in \{f^{(1)}, f^{(2)}, ..., f^{(n)}\}$ is the k-th classifier to be attacked. To ensure robust and stealthy attacks, we alternatively minimize adversarial and stealthiness losses below:

$$x' = \underset{x'}{\operatorname{argmin}}\, \alpha \mathcal{L}_{\text{adv}}\left(\hat{I}_{x',\gamma}\right) + \mathcal{D}\left(\hat{I}_{x',\gamma},\ I_{x_0,\gamma}\right), \tag{4}$$

where $\alpha = -1$ for targeted attacks and $\alpha = 1$ for untargeted attacks. $\mathcal{D}$ is perceptual color distance ΔE (*i.e.*, CIEDE2000 [12]), and it has been experimentally demonstrated to better align with human visual perception and produce more robust and transferable attacks [13] compared with l_p norm. $\hat{I}_{x',\gamma}$ represents the simulated camera-captured adversarial projection rather than the real one ($I_{x',\gamma}$) to avoid including the physical project-and-capture process π in the optimization loop because π is non-differentiable and it is highly inefficient even with gradient-free optimization. Inspired by [10], we use a neural network named PCNet $\hat{\pi}_\theta$ (parameterized by θ) to approximate the physical project-and-capture process π. PCNet consists of two components: ShadingNet (for photometry) and WarpingNet (for geometry), as shown in Fig. 2. The simulated project-and-capture process is denoted as $\hat{I}_{x',\gamma} = \hat{\pi}_{\theta,\gamma}(x')$, with θ representing its parameters. PCNet is trained by minimizing the loss between the real captured projections $I_{x,\gamma}$ and the inferred ones $\hat{I}_{x,\gamma}$:

$$\theta = \underset{\theta}{\operatorname{argmin}} \sum\nolimits_i \mathcal{L}_{\text{PC}}(\hat{I}_{x_i,\gamma_0} = \hat{\pi}_{\theta',\gamma_0}(x_i),\ I_{x_i,\gamma_0}), \tag{5}$$

where $\mathcal{L}_{\text{PC}}$ is pixel-wise L_1 + DSSIM loss, γ_0 is the camera pose where PCNet is trained, and $\{(x_i, I_{x_i,\gamma_0})\}_{i=1}^M$ forms M pairs of real projected and captured images for training.

Classifier-Agnostic adversarial loss. We now introduce the adversarial loss function $\mathcal{L}_{\text{adv}}$ for classifier-agnostic attacks. For classifier-agnostic **untargeted** attacks, an intuitive solution is to use the weighted sum of the adversarial loss of each classifier. Denote $z_i^{(k)}(\cdot)$ as the k-th classifier's output logit (raw classification score) of the i-th label, which is related to f_i^k by: $f_i^k = \text{softmax}(z_i^{(k)})$. Then, our untargeted classifier-agnostic adversarial attack loss is given by

$$\mathcal{L}_{\text{adv}}\left(\hat{I}_{x',\gamma}\right) = \sum\nolimits_{k=1}^{n} \omega_k \cdot z_i^{(k)}\left(\hat{I}_{x',\gamma_0}\right), \tag{6}$$

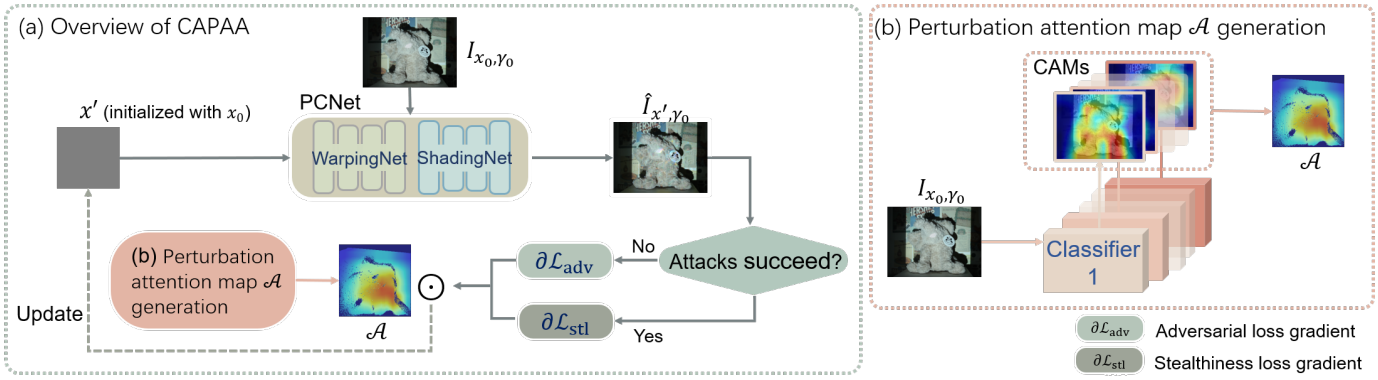

Fig. 2. **(a)** Overview of CAPAA. We first input the adversarial projection x' (initialized with gray image x_0) and the camera image I_{x_0,γ_0} to the trained PCNet to obtain the inferred projection $\hat{I}_{x',\gamma_0}$. After generating perturbation attention maps (PAM) for each classifier, we calculate their weighted sum $\mathcal{A}$ for attention-based gradient weighting. The optimization follows an alternating mechanism, i.e., if $\hat{I}_{x',\gamma_0}$ successfully attacks the classifiers, the stealthiness loss gradient is calculated and weighted by $\mathcal{A}$ to update x'. Otherwise, the classifier-agnostic adversarial loss gradient is applied to update x', as outlined in Algorithm 1. **(b)** Perturbation attention map $\mathcal{A}$ generation. For each classifier, we first generate its class activation map (CAM) of the camera-captured scene image I_{x_0,γ_0} using Grad-CAM++ [14]. The weighted sum of these individual CAMs is utilized as our PAMs' $\mathcal{A}$, enabling our CAPAA to generate adversarial projections towards the most salient regions of the object.

where ω_k stands for the weight of the k-th classifier.

A more challenging problem is the **targeted** attack, where the above simple weighted sum of the adversarial loss of each classifier may fail, as the simulated projector-based attack may fail in the real world, due to the perturbations of the complex environment. In such cases, the classifier may recognize the real camera-captured object under the adversarial projection as neither the object's true class nor the attack's target class, but rather as a class similar to the target class. For example, when projecting an adversarial pattern onto the object **Teddy** to fool the classifier into recognizing it as **rooster**, the classifier might instead output **hen**. This is because the original softmax function inherently emphasizes the largest logit, and adversarial attacks may produce right-above-the-margin perturbations, which are less robust after real-world project-and-capture processes. To address this issue, we add stricter constraints to the classifier's output logits by controlling the temperature of the LogSoftmax function, such that the adversarial attack is only successful when the classifier's target logit is significantly higher than the other classes:

$$\mathcal{L}_{\text{adv}}\left(\hat{I}_{x',\gamma}\right) = \sum_{k=1}^{n} \omega_k \cdot \text{LogSoftmax}\left(z_t^{(k)}\left(\hat{I}_{x',\gamma_0}\right)/T\right), \quad (7)$$

where the parameter T acts as a temperature parameter, and it is dynamically adjusted during the optimization process. When $T = 1$, the softmax function behaves as a standard output layer for classifiers. For $T > 1$, the Softmax distribution becomes smoother. In adversarial training, this helps classifiers with standard Softmax outputs generate adversarial examples that better distinguish between the target class and similar classes (e.g., hen and rooster), thereby reducing ambiguity.

Attention-based gradient weighting. To improve the robustness of the adversarial projection under varying camera poses, we propose an attention-based gradient weighting mechanism. It is based on the observation that (1) adversarial projections may be occluded or move out of the camera's field of view when the camera pose changes. However, most existing methods apply perturbations uniformly across all regions, and may fail when the camera pose changes. (2) Classifiers often focus on specific regions of the object when making predictions. Therefore, we propose to focus perturbations on regions with strong classification activation, as shown in Fig. 2 **(b)**.

To find the regions of strong classification activation, an intuitive method is to use an object detector, such as YOLO [15], to locate the object and apply perturbations within its bounding box. However, this introduces additional complexity and potential reliability issues with detection. Instead, we employ an attention mechanism, specifically Grad-CAM++ [14], to find the class activation map (CAM) on the object. Then, in each adversarial attack iteration, we weigh the loss gradient using CAM, focusing the perturbations on regions with high classification activation, as shown in Eqn. 8.

$$\frac{\partial\mathcal{L}_{\text{CAPAA}}}{\partial x'} = \mathcal{A} \odot \left(\underbrace{\frac{\partial\mathcal{L}_{\text{adv}}\left(\hat{I}_{x',\gamma_0}\right)}{\partial x'}}_{\text{adversarial loss gradient}} + \underbrace{\frac{\partial\mathcal{L}_{\text{stl}}\left(\hat{I}_{x',\gamma_0},\, I_{x_0,\gamma_0}\right)}{\partial x'}}_{\text{stealthiness loss gradient}} \right), \quad (8)$$

where $\mathcal{A}$ is the perturbation attention map (PAM) represented by CAM, and $\odot$ denotes element-wise multiplication. The overall process of CAPAA is illustrated in Fig. 2 and Algorithm 1. To elucidate, we first initialize x' with a plain gray projector image x_0 and set $\mu = 1/N$ for each classifier's PAM $\mathcal{A}^{(k)}$. We set the learning rate $\beta_1 = 2$ for minimizing the adversarial loss and $\beta_2 = 1$ for minimizing the stealthiness loss. We then iteratively update x' by minimizing the adversarial loss when the adversarial confidence is below a threshold $p_{\text{thr}} = 0.9$ or the perturbation size is below a threshold d_{thr} ($2 \le d_{\text{thr}} \le 5$). Otherwise, we minimize the stealthiness loss. The final output adversarial projection x' is the one that is adversarial and has the smallest perceptual color distance ΔE to the original projector image x_0.

III. Experimental Evaluation

A. Experiment setup

As shown in Fig. 3, our setup consists of a projector and a camera, both facing a target object to be attacked. We start by capturing the object image under gray light x_0 and training PCNet. We then generate adversarial patterns using four different methods, including CAPAA, for both targeted (10 targets) and untargeted attacks. Next, we project the generated

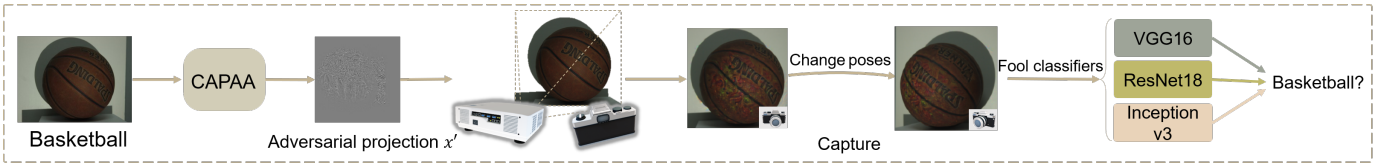

Fig. 3. Overview of the experimental evaluation. First, we sample the object and train PCNet. Then, we use different methods (*e.g.*, CAPAA) to generate the adversarial projections. After that, we project the adversarial patterns onto the object and move the camera to capture the scene in different poses. Finally, the captured images (the object with superimposed adversarial projection) are fed to different classifiers for prediction.

Algorithm 1: CAPAA: Classifier-Agnostic Projector-Based Adversarial Attack

Input:
x_0, I_{m}: projector plain gray image, projector direct light mask
I_s: camera-captured scene under x_0 projection
$\mathcal{A}$: perturbation attention maps (PAM)
$\mu^{(k)}$: weight of the k-th classifier's PAM
K: number of iterations
p_{thr}: threshold for adversarial confidence
d_{thr}: threshold for ΔE perturbation size
β_1, β_2: step sizes for adversarial and stealthiness losses
Output: x': projector input adversarial image

Initialize $x'_0 \leftarrow x_0$
$\mathcal{A} \leftarrow \sum_{k=1}^{N} \mu^{(k)} \mathcal{A}^{(k)}$
for $j \leftarrow 1$ **to** K **do**
  $\hat{I}_{x',\gamma_0} \leftarrow \hat{\pi}_{\theta,\gamma_0}(x'_{j-1})$
  $d \leftarrow \mathcal{D}\left(\hat{I}_{x',\gamma_0}, I_{x_0,\gamma_0}\right)$
  if $f_{y_t}(\hat{I}_{x',\gamma_0}) < p_{\mathrm{thr}}$ **or** $d < d_{\mathrm{thr}}$ **then**
    $g_1 \leftarrow \mathcal{A} \odot \alpha \nabla_{x'} \mathcal{L}_{\mathrm{adv}}\left(\hat{I}_{x',\gamma_0}\right)$ // min. adversarial loss
    $x'_j \leftarrow x'_{j-1} + \beta_1 * \frac{g_1}{\|g_1\|_2}$
  else
    $g_2 \leftarrow -\mathcal{A} \odot \nabla_{x'} d$ // min. stealthiness loss
    $x'_j \leftarrow x'_{j-1} + \beta_2 * \frac{g_2}{\|g_2\|_2}$
  end if
  $x'_j \leftarrow \mathrm{clip}(x'_j, 0, 1)$
end for
return $x' \leftarrow x'_j$ that is adversarial and has smallest d

adversarial patterns onto the object and capture the scene under different camera poses, e.g., the original pose, different angles (±15°, ±30°) and different focal lengths (±5mm). Finally, we feed the camera-captured images into three classifiers (ResNet-18 [16], VGG-16 [17], and Inception v3 [18]) for real-world projector-based adversarial attack evaluation.

Evaluation metrics. To measure the attack success rate and stealthiness, we define a stealthiness-constrained attack success rate metric for the camera-capture adversarial projection $I_{x',\gamma}$:

$$\mathcal{S}_h^{(k)}(I_{x',\gamma}) = \begin{cases} 1, & \text{if } \hat{y} = \underset{i}{\operatorname{argmax}}\, f_i(I_{x',\gamma}) \begin{cases} = y_t, & \text{targeted} \\ \neq y, & \text{untargeted} \end{cases} \\ & \quad \text{and } \mathcal{D}\left(I_{x',\gamma}, I_{x_0,\gamma}\right) \le h \\ 0, & \text{otherwise.} \end{cases}$$

This metric ensures that a projector-based attack is successful only when it fools the given classifier and its stealthiness ΔE is no greater than h. Then, we plot the success rate vs stealthiness diagrams of all compared methods. As shown in Fig. 6**(a) - (c)**, the horizontal axis corresponds to the perturbation threshold ΔE [12], and the vertical axis represents the

TABLE I
QUANTITATIVE COMPARISONS FOR CLASSIFIER-AGNOSTIC MULTI-POSE **UNTARGETED** ATTACKS. FOUR STEALTHINESS THRESHOLDS $d_{\mathrm{thr}} \in \{2, 3, 4, 5\}$ ARE USED TO GENERATE ADVERSARIAL PROJECTIONS (2ND COLUMN). COLUMNS 3 TO 6 PRESENT STEALTHINESS METRICS FOR *camera-captured* ADVERSARIAL PROJECTIONS, COLUMN 7 INDICATES THE AVERAGE TOP-1 SUCCESS RATE, AND COLUMN 8 SHOWS THE AVERAGE TOP-1 SUCCESS RATE *across all stealthiness thresholds* OVER 10 DIFFERENT SETUPS, AND EACH SETUP CONSISTS OF 7 CAMERA POSES.

Attacker	d_{thr}	$L_{\mathrm{inf}} \downarrow$	$L_2 \downarrow$	$\Delta E \downarrow$	SSIM↑	U.top-1	Avg. attack success rate
SPAA [10]	2	5.11	6.38	2.25	0.914	51.43%	64.68%
	3	7.16	8.94	3.01	0.862	62.86%	
	4	9.02	11.18	3.83	0.828	68.73%	
	5	10.64	13.08	4.63	0.805	75.71%	
CAPAA w/o attention	2	5.24	6.55	2.29	0.911	71.90%	**82.02%**
	3	7.23	9.04	3.05	0.860	81.43%	
	4	9.04	11.20	3.87	0.827	87.14%	
	5	10.56	12.98	4.66	0.804	**87.62%**	
CAPAA classifier-specific	2	**4.73**	**5.89**	**2.09**	0.927	51.59%	61.75%
	3	6.43	7.96	2.80	0.889	62.70%	
	4	7.72	9.47	3.47	0.868	65.08%	
	5	8.49	10.37	3.92	0.858	67.94%	
CAPAA (ours)	2	4.77	5.95	2.10	**0.930**	74.76%	**82.02%**
	3	6.36	7.85	2.82	0.895	81.90%	
	4	7.48	9.15	3.46	0.877	84.76%	
	5	8.01	9.74	3.83	0.871	86.67%	

cumulative success rate $\mathcal{C}_h$ at a given ΔE threshold h:

$$\mathcal{C}_h = \frac{1}{PNH} \sum_{j=0}^{P-1} \sum_{k=1}^{N} \sum_{l=1}^{H} \mathcal{S}_h^{(k)}(I_{x'_l,\gamma_j}), \tag{9}$$

where P, N, H are the number of camera poses, the number of image classifiers to be attacked, and the number of generated adversarial perturbations, respectively. In particular, $\mathcal{S}_h^{(k)}(I_{x'_l,\gamma_j})$ indicates whether the l-th camera-captured adversarial projection successfully fools the k-th classifier $f^{(k)}$ at the j-th camera pose, meanwhile, its ΔE is less than h. Note that we evaluate: (i) targeted attacks at the original pose ($P = 1$, Fig. 6 **(b)**), and (ii) targeted/untargeted attacks across multiple poses ($P = 7$, Fig. 6 **(a)** & **(c)**).

Compared baselines. We compare our CAPAA with three baselines: SPAA [10], CAPAA (w/o attention), and CAPAA (classifier-specific). SPAA [10] is the closest projector-based adversarial attack method to our CAPAA, but it is classifier-specific and does not consider attack robustness across other camera poses. CAPAA (w/o attention) is a degraded CAPAA that jointly attacks multiple classifiers but with no attention-based gradient weighting, and CAPAA (classifier-specific) is a degraded CAPAA without classifier-agnostic adversarial loss, thus can only attack each classifier individually. Since SPAA and CAPAA (classifier-specific) cannot perform classifier-

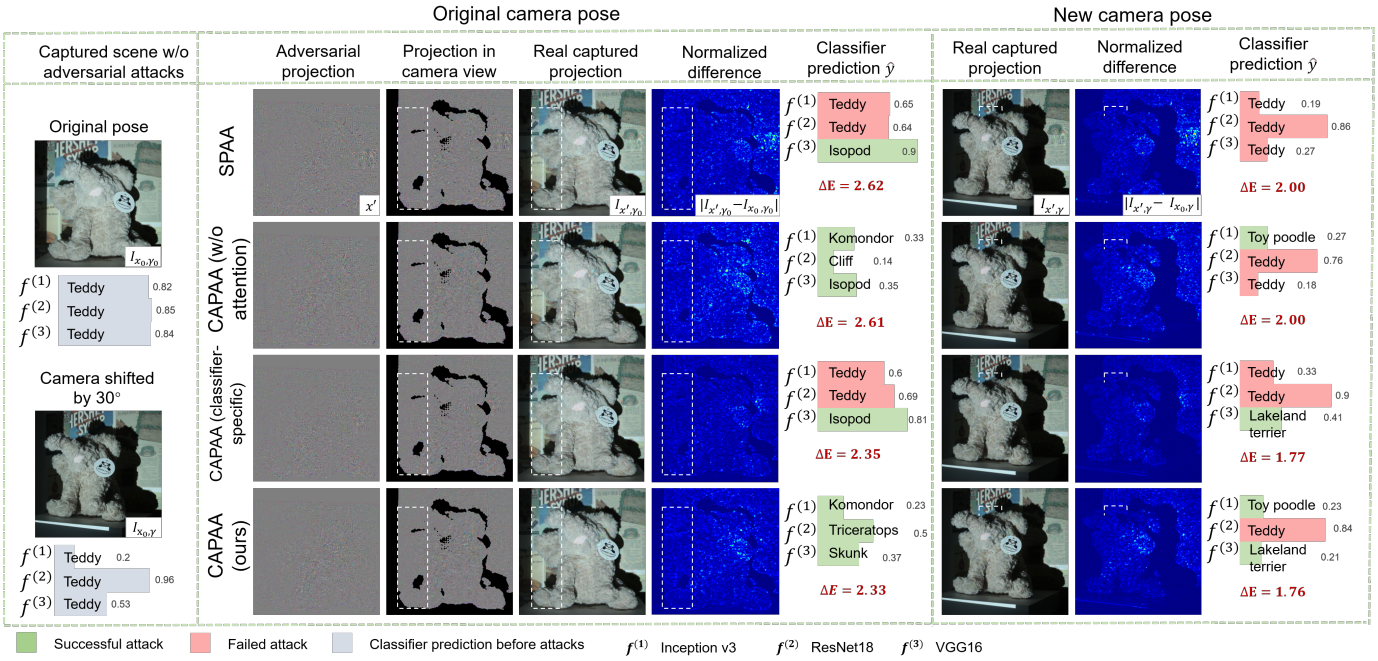

Fig. 4. Qualitative comparisons of classifier-agnostic **untargeted** attacks across two camera views. The classifier prediction $\hat{y}$, including the probabilities, is displayed on the bottom or right side of each image. The perturbations highlighted by the white dashed boxes, especially in the 5th and 7th columns, indicate that attention-based attacks, CAPAA and CAPAA (classifier-specific) tend to avoid attacking background regions due to the CAM mechanism, and thus are more robust against occlusions (caused by camera pose changes) compared to other baselines.

agnostic attacks, we evaluate them in a classifier-specific manner, i.e., attack each classifier individually, which gives them advantages over classifier-agnostic ones.

Clearly, CAPAA outperforms all baseline approaches in terms of stealthiness.

B. Experimental results

Untargeted attack. As shown in Table II, the average attack success rates of CAPAA and CAPAA (w/o attention) achieve the highest attack success rates (with a marginal 0.001% difference) and consistently outperform other methods across various stealthiness thresholds. Moreover, CAPAA outperforms CAPAA (w/o attention) when the stealthiness threshold $d_{\text{thr}} \leq 3$ and excels in stealthiness metrics such as L_{inf}, L_2, ΔE [12], and SSIM. CAPAA (classifier-specific) enhances stealthiness. Notably, CAPAA maintains high success rates while enhancing stealthiness, demonstrating its capability to generate more robust adversarial projections. The curves in Fig. 6 **(a)** further indicate that CAPAA shows the most rapid growth and the highest cumulative success rate, underscoring its effectiveness in balancing stealthiness and success rate.

Note that after changing camera poses, some adversarial projections become invisible due to occlusion. For example, in Fig. 5, the background perturbations highlighted in white dashed boxes are out of the camera FOV after changing the camera pose. In Fig. 4, the background perturbations are occluded by the object **Teddy** after changing the camera pose. However, CAPAA and CAPAA (classifier-specific) are less affected because they can focus adversarial perturbations on the object by using CAM. Moreover, attention-based techniques yield stealthier projections; for example, CAPAA (classifier-specific) and CAPAA exhibit smaller stealthiness (ΔE) than other baselines in the original pose. Notably, in Fig. 4, after a 30° camera shift, only CAPAA achieved two successful attacks with the highest stealthiness, while SPAA failed in all attempts. Similarly, as shown in the attacks against **Lotion** in Fig. 5, SPAA only succeeded once, whereas CAPAA successfully fooled all classifiers with a smaller ΔE. Although CAPAA (w/o attention) also succeeded, but with a higher ΔE. We also conducted additional experiments attacking Vision Transformers (ViTs) [1] and four unseen classifier architectures. The results demonstrate consistent superiority over baselines across all tested models, while revealing limitations for future improvement (details are in the supplementary material).

Targeted attack. Fig. 6 **(c)** shows that CAPAA outperforms other methods in both success rate and stealthiness on targeted attacks. Note that targeted attacks are much more challenging than untargeted ones, resulting in lower average success rates. CAPAA and CAPAA (w/o attention) lead in performance for classifier-agnostic targeted attacks at the original camera pose, with CAPAA (w/o attention) tripling the success rate due to the three classifiers targeted. CAPAA also shows improved performance when lower stealthiness (i.e., larger ΔE) is allowed, confirming its effectiveness, particularly at the original camera pose (Fig. 6 **(b)**).

IV. Conclusion and Limitations

We propose CAPAA, a classifier-agnostic projector-based adversarial attack method that is robust even when the camera pose changes. CAPAA combines a novel classifier-agnostic

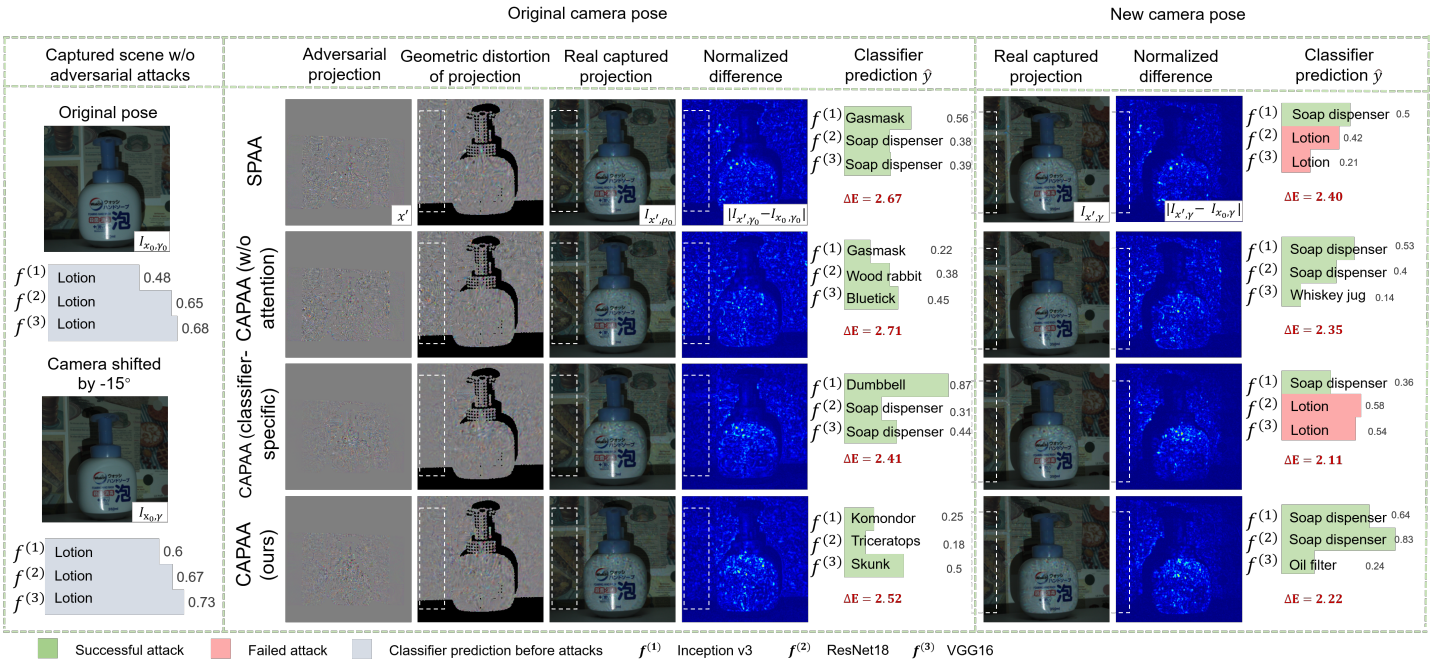

Fig. 5. Qualitative results of the classifier-agnostic and multi-pose **untargeted** attacks. Specifically, the white frames show how the perturbations on the background are out of the camera FOV after shifting the camera angle.

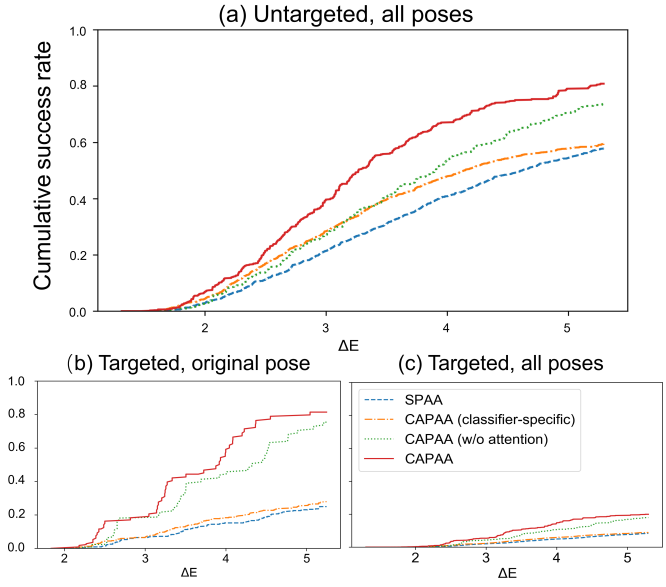

Fig. 6. Quantitative comparisons on projector-based classifier-agnostic adversarial attacks. **(a) Untargeted** attacks. **(b) Targeted** attacks under the *original* camera capture pose. **(c) Targeted** attacks across *all* camera capture poses.

adversarial loss with an attention-based gradient weighting strategy to achieve both stealthy and robust adversarial projections. On a benchmark with 10 setups (10 objects and 7 poses), we show that CAPAA outperforms existing methods in stealthiness and achieves high attack success rates.

Limitations and future work. Although robust against camera pose changes, CAPAA is not pose-agnostic because it does not aggregate attack loss gradients from multiple camera poses. Future work is to incorporate various camera poses to address this issue.

References

[1] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *ICLR*, vol. abs/1412.6572, 2015.

[2] Q. Huang, Z. Lian, and Q. Li, "Attention based adversarial attacks with low perturbations," in *ICME*, 2022, pp. 1–6.

[3] P. Benz, C. Zhang, A. Karjauv, and I. S. Kweon, "Universal adversarial training with class-wise perturbations," in *ICME*, 2021, pp. 1–6.

[4] J. Fang, Y. Jiang, C. Jiang, Z. L. Jiang, C. Liu, and S.-M. Yiu, "State-of-the-art optical-based physical adversarial attacks for deep learning computer vision systems," *ESWA*, p. 123761, 2024.

[5] X. Wei, Y. Guo, and J. Yu, "Adversarial sticker: A stealthy attack method in the physical world," *TPAMI*, vol. 45, pp. 2711–2725, 2021.

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, L. Kai, and F.-F. Li, "Imagenet: A large-scale hierarchical image database," in *CVPR*, 2009, pp. 248–255.

[7] A. Gnanasambandam, A. M. Sherman, and S. H. Chan, "Optical adversarial attack," *ICCVW*, pp. 92–101, 2021.

[8] C. Hu, W. Shi, and L. Tian, "Adversarial color projection: A projector-based physical-world attack to dnns," *Image and Vision Computing*, vol. 140, p. 104861, 2023.

[9] H. Wei, H. Tang, X. Jia, Z. Wang, H. Yu, Z. Li, S. Satoh, L. Van Gool, and Z. Wang, "Physical adversarial attack meets computer vision: A decade survey," *TPAMI*, vol. 46, no. 12, pp. 9797–9817, 2024.

[10] B. Huang and H. Ling, "Spaa: Stealthy projector-based adversarial attacks on deep image classifiers," in *VR*, 2022, pp. 534–542.

[11] Y. Guo, X. Wang, P. Xiao, and X. Xu, "An ensemble learning framework for convolutional neural network based on multiple classifiers," *Soft Computing*, vol. 24, no. 5, pp. 3727–3735, 2020.

[12] M. R. Luo, G. Cui, and B. Rigg, "The development of the CIE 2000 colour-difference formula: CIEDE2000," *Color Research & Application*, vol. 26, no. 5, pp. 340–350, 2001.

[13] Z. Zhao, Z. Liu, and M. Larson, "Towards large yet imperceptible adversarial image perturbations with perceptual color distance," in *CVPR*, 2020, pp. 1036–1045.

[14] A. Chattopadhyay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks," *WACV*, pp. 839–847, 2017.

[15] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *CVPR*, 2016.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.

[17] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *ICLR*, 2015.
[18] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *CVPR*, 2016, pp. 2818–2826.
[19] A. Dosovitskiy *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," in *ICLR*, 2021.

CAPAA: Classifier-Agnostic Projector-Based Adversarial Attack

— Supplementary Materials —

V. Introduction

In this supplementary material, we present the results of adversarial attacks against Vision Transformers (ViTs) [1]. Using **teddy** as the target object, we employ Grad-CAM to analyze attention maps and evaluate attack effectiveness. As demonstrated in Table II, our proposed CAPAA method significantly outperforms SPAA in classifier-agnostic multi-pose untargeted attacks, achieving a 3× higher average attack success rate against ViT-Base-16. Additionally, Table III reveals near-perfect success rates (93.75%) under the original pose configuration. While these results demonstrate strong performance, enhancing transferability to newer ViT variants represents an interesting direction for future research.

TABLE II
QUANTITATIVE COMPARISONS FOR CLASSIFIER-AGNOSTIC **MULTI-POSE UNTARGETED** ATTACKS.

Attacker	d_{thr}	Classifier	SSIM↑	L_2 ↓	ΔE ↓	L_{inf} ↓	U.top-1	Avg. attack success rate
CAPAA	2	Inception v3	**0.902**	14.13	3.75	10.27	20%	**48.75%**
		Resnet-18					20%	
		VGG-16					40%	
		ViT-Base-16					0	
	3	Inception v3	0.861	16.15	4.40	11.98	20%	
		Resnet-18					40%	
		VGG-16					**100%**	
		ViT-Base-16					20%	
	4	Inception v3	0.828	18.66	5.32	14.17	20%	
		Resnet-18					80%	
		VGG-16					**100%**	
		ViT-Base-16					20%	
	5	Inception v3	0.807	21.13	6.18	16.22	80%	
		Resnet-18					**100%**	
		VGG-16					**100%**	
		ViT-Base-16					20%	
SPAA	2	Inception v3	0.878	15.44	3.87	11.19	5%	15.31%
		Resnet-18	0.882	15.37	3.95	11.19	5%	
		VGG-16	0.869	**13.85**	3.72	10.21	10%	
		ViT-Base-16	0.874	13.90	**3.64**	**10.20**	5%	
	3	Inception v3	0.840	17.13	4.54	12.78	5%	
		Resnet-18	0.834	15.63	4.35	11.69	15%	
		VGG-16	0.807	16.47	4.56	12.38	25%	
		ViT-Base-16	0.835	15.65	4.34	11.76	5%	
	4	Inception v3	0.812	18.15	5.25	13.92	5%	
		Resnet-18	0.805	18.09	5.24	13.89	25%	
		VGG-16	0.775	19.46	5.60	14.93	40%	
		ViT-Base-16	0.809	18.01	5.14	13.72	5%	
	5	Inception v3	0.801	18.91	5.64	14.61	10%	
		Resnet-18	0.792	19.39	6.00	15.20	35%	
		VGG-16	0.758	21.73	6.48	16.90	45%	
		ViT-Base-16	0.795	19.62	6.07	15.29	5%	

We also evaluated our method through comprehensive adversarial attacks across ten distinct experimental setups, each comprising 10 objects with 7 poses per object (totaling 70 test cases per setup). The evaluation covered four unseen classifier architectures: ConvNeXt-Base [2], EfficientNet-B0 [3], MobileNetV3-Large [4], and Swin Transformer-Base [5]. As demonstrated in Table IV, our approach consistently outperforms the baseline across all classifiers and test conditions. While these results confirm the robustness of our method under varied pose-object combinations, we identify opportunities for further enhancement in cross-architecture transferability.

TABLE III
QUANTITATIVE COMPARISONS FOR CLASSIFIER-AGNOSTIC **POSE-SPECIFIC UNTARGETED** ATTACKS.

Attacker	Classifier	ΔE ↓	SSIM↑	L_2 ↓	U.top-1	Avg. attack success rate
CAPAA (ours)	Inception v3	4.92	0.838	17.86	**100%**	**93.75%**
	Resnet-18				**100%**	
	VGG-16				**100%**	
	ViT-Base-16				75%	
SPAA	Inception v3	4.85	0.804	**17.18**	25%	35.94%
	Resnet-18	5.07	0.798	17.87	44%	
	VGG-16	5.20	**0.773**	18.27	50%	
	ViT-Base-16	**4.86**	0.801	16.82	25%	

TABLE IV
AVERAGE ATTACK SUCCESS RATE FOR CLASSIFIER-AGNOSTIC **MULTI-POSE UNTARGETED** ATTACKS. CONVNEXT.B, MOBILENETV3.L AND SWIN TF. B STAND FOR CONVNEXT-BASE, MOBILENETV3 LARGE AND SWIN TRANSFORMER BASE, RESPECTIVELY.

Attacker	ConvNeXt.B	EfficientNet-B0	MobileNetV3.L	Swin TF. B
SPAA	36.67%	47.26%	54.64%	26.67%
CAPAA (ours)	**38.57%**	**50.71%**	**58.21%**	**29.29%**

References

[1] A. Dosovitskiy et al., "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," in *ICLR*, 2021.

[2] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A ConvNet for the 2020s," in *CVPR*, 2022, pp. 11966-11976.

[3] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *ICML*, 2019, pp. 10691-10700.

[4] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, Q. V. Le, and H. Adam, "Searching for MobileNetV3," in *ICCV*, 2019, pp. 1314–1324.

[5] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin Transformer: Hierarchical vision transformer using shifted windows," in *ICCV*, 2021, pp. 9992-10002.