KDAT: Inherent Adversarial Robustness via Knowledge Distillation with Adversarial Tuning for Object Detection Models

Yarin Yerushalmi Levi[1], Edita Grolman[1], Idan Yankelev[1], Amit Giloni[2], Omer Hofman[2], Toshiya Shimizu[3], Asaf Shabtai[1], Yuval Elovici[1]

[1]Department of Software and Information Systems Engineering, Ben-Gurion University of the Negev, Israel.
[2] Fujitsu Research of Europe.
[3] Fujitsu Unlimited.
{yarinye, edita, idanyan}@post.bgu.ac.il, {shabtaia, elovici}@bgu.ac.il,
{amit.giloni, omer.hofman, shimizu.toshiya}@fujitsu.com,

Abstract

Adversarial patches pose a significant threat to computer vision models' integrity, decreasing the accuracy of various tasks, including object detection (OD). Most existing OD defenses exhibit a trade-off between enhancing the model's adversarial robustness and maintaining its performance on benign images. We propose KDAT (knowledge distillation with adversarial tuning), a novel mechanism that enhances the robustness of an OD model without compromising its performance on benign images or its inference time. Our method combines the knowledge distillation (KD) technique with the adversarial tuning concept to teach the model to match the predictions of adversarial images with those of their corresponding benign ones. To match these predictions, we designed four unique loss components, allowing the student model to effectively distill the knowledge of different features from various parts of the teacher model. Our extensive evaluation on the COCO and INRIA datasets demonstrates KDAT's ability to improve the performance of Faster R-CNN and DETR on benign images by 2-4 mAP% and adversarial examples by 10-15 mAP%, outperforming other state-of-the-art (SOTA) defenses. Furthermore, our additional physical evaluation on the Superstore dataset demonstrates KDAT's SOTA adversarial robustness against printed patches (improvement of 22 mAP% compared to the undefended model).

Code — https://github.com/Yarinyl/KDAT

1 Introduction

While object detection (OD) models are integrated into many aspects of our daily life, e.g., autonomous vehicles and surveillance (Vahab et al. 2019), there is a real concern that adversaries may manipulate the detection process (Sharma et al. 2022). In the computer vision domain, adversaries can create a small, optimized patch, known as a *patch attack*, designed to deceive the OD model. These adversarial patches can be applied to objects in a scene (Brown et al. 2017), necessitating the development of solutions to mitigate such attacks and ensure system reliability. While various defenses have been developed to deal with such attacks (Liu et al. 2022; Xiang et al. 2023; Jing et al. 2024), most of them suffer from two major limitations: 1) they harm the model's performance on benign images, and 2) they add significant overhead to the OD inference stage, which makes them unsuitable for real-time applications (Arani et al. 2022). To address these limitations, we propose KDAT (knowledge distillation with adversarial tuning), a novel fine-tuning strategy that optimizes the parameters of a given OD model against adversarial patches without increasing the inference time or compromising performance on benign images. KDAT employs the knowledge distillation (KD) technique (Hinton, Vinyals, and Dean 2015) to transfer information from one model (the teacher) to another (the student) during training. In our proposed method, the given OD model is replicated to serve as both the student and teacher models. Then the teacher model processes the benign images and transfers the acquired knowledge of benign features to the student model to guide its predictions when facing the corresponding adversarial examples (created for each benign image).

We evaluated KDAT's effectiveness on digital datasets, COCO (Lin et al. 2014) and INRIA (Dalal and Triggs 2005), and the physical Superstore dataset (Hofman et al. 2024), with a wide range of adversarial patch attacks: DPatch (Liu et al. 2018), Google's adversarial patch (Brown et al. 2017), Masked-PGD (M-PGD) (Madry et al. 2017), T-SEA (Huang et al. 2023), natural patches (Hu et al. 2021) and printable patches (Thys, Van Ranst, and Goedemé 2019). Our evaluation demonstrates KDAT's advantage over other SOTA defenses when examining the trade-off between enhancing the model's adversarial robustness and its performance on benign images. The results show that KDAT improved the adversarial robustness against both seen and unseen attacks by 10-15 mAP% and the performance on benign images by 2-4 mAP% for both DETR and Faster R-CNN while preserving their inference time. We also demonstrate that using our proposed defense method forces the attacker to invest additional resources in misleading the defended model, i.e., in an adaptive attack, the attacker requires 8-10% more optimization iterations to successfully mislead the OD model.

In summary, our contributions are as follows: 1) To the best of our knowledge, we are among the first to improve the OD model's performance on both attacked and benign images. 2) To the best of our knowledge, we are also the first to design a fine-tuning strategy against adversarial patches for OD models without the need to retrain the entire model, un-

like most training-based defenses. 3) We are among the first to propose a method that demonstrates robustness against attacks targeting transformer-based OD models. 4) To the best of our knowledge, we are also the first to evaluate the effectiveness of integrating defenses against patch attacks.

2 Background and Related Work

Improving the adversarial patch robustness of OD models can be achieved by modifying the model's (1) training process, in order to produce an inherently robust model, or (2) inference process, by wrapping the original model and preventing the effect of incoming adversarial threats.

2.1 Modifying the Training Process

Limited efforts have been made to enhance robustness through changes to the training process. Under a strict assumption that the patch is not located on the object itself, Saha et al. (2020) adjusted the model's loss function to ignore the objects' surroundings when performing detection. Ji et al. (2021) exposed the model to adversarial patches during training to improve the adversarial robustness; however, the precision on benign images was compromised. In contrast, KDAT modifies the training process to counter adversarial patches using an adversarial tuning process that inherently improves the performance on both adversarial and benign images (regardless of the patch's location).

KD-Based Robustness Solutions. KD is a technique that involves transferring information from one model (the teacher) to another (the student) during training (Hinton, Vinyals, and Dean 2015). Xu et al. (2021; 2022), leveraged the KD technique to mitigate adversarial perturbations. Wang et al. (2024) proposed DFAD, in which the KD process is used to distill adversarial robustness from a pretrained robust teacher to a shallow student when the original training dataset is unavailable. These studies demonstrate the potential of such approaches to obtain adversarial robustness within the OD domain. In contrast, KDAT utilizes the KD technique (without any assumptions on the teacher's model) to improve adversarial robustness against adversarial patch attacks, which, contrary to perturbations attacks, is a more realistic threat in the OD domain (Brown et al. 2017).

2.2 Modifying the Inference Process

These methods do not require retraining the model, as they integrate an external component to make it more robust during inference. A few studies used a segmentation model to distinguish the adversarial patch from the rest of the image and mask it (Chiang, Chan, and Wu 2021; Liu et al. 2022; Xu et al. 2023). Another approach is to identify the characteristics of adversarial examples and mask the corresponding adversarial regions accordingly (Naseer, Khan, and Porikli 2019; Rossolini et al. 2023; Jing et al. 2024). However, these solutions rely on a specific attack behavior and focus on enhancing the robustness by altering the input image, which can harm the benign detection process and result in a longer inference time, which is impractical for real-time OD applications. In contrast, our solution aims to improve the model's adversarial robustness without compromising its precision or inference time.

3 Methodology

While typical KD frameworks require two different models, our solution eliminates this requirement by replicating the original pretrained model. One of the models serves as the teacher (with frozen weights), while the second model serves as the student and performs the training. In addition, while in standard KD training architectures, the input of both the teacher and student is the same, in our method, the teacher receives only benign images while the student encounters the corresponding adversarial images. KDAT transfers the teacher's relevant information, obtained when the detection process is performed on benign images, to the student model, which utilizes these features to make a more robust prediction and mitigate the adversarial threat.

We adopt the adversarial tuning concept (Jeddi, Shafiee, and Wong 2020; Li, Xiao, and Tang 2024; Chen et al. 2020) to eliminate the need to retrain the entire model (unlike traditional training-based defenses) to improve the adversarial robustness of OD models. By adopting this concept for the OD domain, KDAT leverages the knowledge acquired in the model's original training to guide itself toward improved detection on adversarial examples.

KDAT's first phase includes creating adversarial examples corresponding to each image from a given benign set of images. The adversarial examples are crafted using various attack settings (shapes, sizes, intents, etc.), with the aim of challenging the model from different angles during training with a wide range of attacks. Each benign image, along with its corresponding adversarial examples and the ground truth (GT) annotation, are referred to as a data record, as presented below:

$$\forall x_i \in D_{\text{ben}}, \text{ form a set } \{x_i, X_i^{adv}, y_i^{gt}\} \tag{1}$$

where X_i^{adv} is the set of adversarial examples created using the different attack settings, $x_i \in D_{\text{ben}}$ is the original image, and y_i^{gt} is the GT of image x_i. During training, we utilize this data record in each batch, where the teacher receives only benign images, obtaining benign features, while the student receives the benign images, a randomly chosen subset of size n from X_i^{adv} for each benign image, and the GT. In each batch, the student receives a different subset of X_i^{adv}, which enhances the training process, making it less susceptible to overfitting.

Since distilling various features of the OD model has proven to improve the model's benign performance (Wang et al. 2023; Park, Kang, and Paik 2024), we designed four unique loss components to distill both adversarial and benign features. Accordingly, our loss function includes the following components: 1) OD loss, 2) feature map (FM) loss, 3) classification (CLS) loss, and 4) family architecture adjustable (FA) loss. In our method, we compare the values generated by the student (denoted as S) when processing the benign (F_{ben}^S) and adversarial examples (F_{adv}^S) with the values generated by the teacher (denoted as T) when processing the benign image (F_{ben}^T). By also comparing the values generated from the benign image, we overcome the common trade-off between improving the adversarial robustness and maintaining high benign performance (Bai et al. 2021). B and A are hyperparameters that control this trade-off.

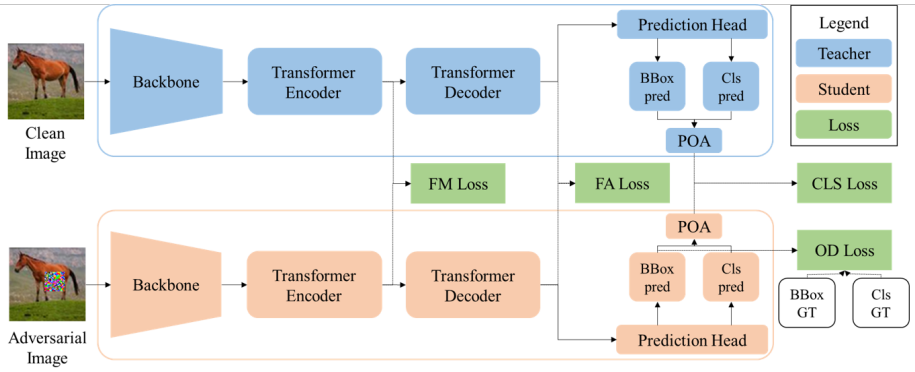

Figure 1: Proposed solution pipeline for transformer-based OD models.

3.1 Object Detection Loss

The original loss function of the OD models guides the model to match its prediction to the GT. While traditional adversarial training (AT) solutions do not distinguish between the loss of benign prediction and the loss of adversarial prediction, our method separates those two values, giving them different weights in the overall computation. By combining this concept with the use of multiple adversarial examples generated for each benign image, our method contributes to the model's ability to sustain high performance on benign images while enhancing its adversarial robustness. Thus, the loss can be calculated as follows:

$$L_{OD} = B \cdot \mathcal{L}_{OD}(S(x_{\text{ben}}), y_{gt}) + A \cdot \frac{1}{n} \sum_{j=0}^{n} \mathcal{L}_{OD}(S(x_{\text{adv}}^{j}), y_{gt}) \quad (2)$$

where x_{ben} is the benign image, x_{adv}^{j} is a matching adversarial example, and S is the student model. $\mathcal{L}_{OD}$ is the original loss function of the given OD model. This is the only loss component that utilizes the GT, whereas the rest of the components use intermediate features as the target objective.

3.2 Feature Map Loss

The combination of early-stage information transfer with the distillation of the outputs (Romero et al. 2014) was shown to improve the performance of OD models (Chen et al. 2017). Most OD architectures include a backbone responsible for extracting the features from the images (Ren et al. 2015; Carion et al. 2020), with each architecture using those features differently to proceed with the detection process. For example, in two-stage detectors (such as Faster R-CNN), the FM is processed in the region proposal network (RPN), resulting in improved detection features; on the other hand, in transformer-based detectors (such as DETR), the FM is processed in a transformer encoder, resulting in an embedded representation of the FM. In our solution, instead of comparing the original FMs (between the teacher and the student), we compare the abovementioned informative features' representation of each architecture, as demonstrated by comparing the output of the encoders in Figure 1.

Teaching the student to mimic the teacher's features is challenging, as the adversarial patch covers some of them. To overcome this challenge, we use masked images (images with black patches) instead of benign ones to teach the student to ignore the patch rather than mimic the features behind it. Thus, the loss can be calculated as follows:

$$L_{FM} = B \cdot \mathcal{L}_p(EFM_{\text{ben}}^{S}, EFM_{\text{ben}}^{T}) + A \cdot \frac{1}{n} \sum_{j=0}^{n} \mathcal{L}_p(EFM_{\text{adv}^j}^{S}, EFM_{\text{masked}^j}^{T}) \quad (3)$$

where EFM is the enhanced FM, and $\mathcal{L}_p$ is the p-norm loss.

3.3 Classification Loss

Inspired by (Hinton, Vinyals, and Dean 2015; Papernot et al. 2016; Li et al. 2022), we created a novel adaptation of soft label (probability vectors) utilization for OD. Our approach employs soft labels, referred to as probability vectors over areas (POA), which can be obtained from the model's various predictions on a given image. Note that we use the original unfiltered predictions received for each image (each OD architecture has a different filtering process).

Linking the predictions of two different models can be done using the intersection over union (IoU) metric to determine if a pair of predictions refers to matching areas in the same image. We compare the distribution of the probability vectors referring to the same area in the POA to lead the student toward the benign values. Figure 1 demonstrates the comparison of each model's POA in the CLS loss component. Thus, the loss can be calculated as follows:

$$L_{CLS} = B \cdot \mathcal{L}_p(POA_{\text{ben}}^{S}, POA_{\text{ben}}^{T}) + A \cdot \frac{1}{n} \sum_{j=0}^{n} \mathcal{L}_p(POA_{\text{adv}^j}^{S}, POA_{\text{ben}}^{T}) \quad (4)$$

3.4 Family Architecture Adjustable Loss

All the loss components presented so far are independent of the given model's architecture. However, different OD architectures have unique components and valuable features that can be leveraged to obtain additional information, which is utilized in the FA component.

Objectness Loss. For two-stage OD architectures, we utilize the additional head, the localization head, as our unique source of information. This localization information has been utilized in different ways in the KD domain (Zheng et al. 2023; Park, Kang, and Paik 2024). This information includes 1) bounding boxes regarding the location of objects in the image and 2) confidence values regarding their appearance (objectness). In our method, we guide the student's objectness values to match the teacher's, forcing it to locate similar objects, even in the presence of an attack. Thus, the loss can be calculated using:

$$L_{FA} = B \cdot \mathcal{L}_p(OB^S_{\text{ben}}, OB^T_{\text{ben}}) + A \cdot \frac{1}{n} \sum_{j=0}^{n} \mathcal{L}_p(OB^S_{\text{adv}^j}, OB^T_{\text{ben}}) \quad (5)$$

where OB represents the corresponding objectness values.

Embedding Loss. Transformer-based OD uses an embedded representation of the input image to predict the classes and bounding boxes of objects in the scene. Assuming that the benign and corresponding adversarial examples should have the same embedding, we penalize the model based on the distance between the matching embedded representations, as demonstrated in Figure 1 by compering the output of the decoders. Thus, the loss can be calculated as follows:

$$L_{FA} = B \cdot \mathcal{L}_p(EM^S_{\text{ben}}, EM^T_{\text{ben}}) + A \cdot \frac{1}{n} \sum_{j=0}^{n} \mathcal{L}_p(EM^S_{\text{adv}^j}, EM^T_{\text{ben}}) \quad (6)$$

where EM represents the corresponding embedding values.

3.5 KDAT's Complete Loss Function

KDAT's loss function incorporates all of the components described above in a weighted sum:

$$L = \alpha_1 \cdot L_{OD} + \alpha_2 \cdot L_{FM} + \alpha_3 \cdot L_{CLS} + \alpha_4 \cdot L_{FA} \quad (7)$$

where α_i, $i \in [1, 4]$ are hyperparameters for adjusting the weight of each component, and L_{OD}, L_{FM}, L_{CLS}, and L_{FA} are the OD, FM, CLS, and FA losses, respectively. By optimizing this loss, we produce a model that maintains high performance on both benign and adversarial images. The proposed training pipeline for transformer-based OD models is demonstrated in Figure 1, while a similar pipeline for two-stage OD can be found in the supplementary material.

3.6 Self-Guiding

The main idea of KD is to teach the student model to mimic the teacher model in order to obtain comparable performance. However, during training, the performance of the student model may surpass the teacher's performance; thus, we adapt the tuning process as follows: In the case in which $\mathcal{L}_{OD}(S(x_{\text{ben}}), y_{gt}) > \mathcal{L}_{OD}(T(x_{\text{ben}}), y_{gt})$, i.e., the teacher performed better than the student on the benign images, we utilize the teacher's features (from the benign predictions). In contrast, when $\mathcal{L}_{OD}(S(x_{\text{ben}}), y_{gt}) < \mathcal{L}_{OD}(T(x_{\text{ben}}), y_{gt})$, i.e., the student performed better than the teacher on the benign images, we utilize the student's features (from the benign predictions).

4 Evaluation

4.1 Evaluation Settings

This subsection outlines the primary evaluation settings, with additional details provided in the supplementary material such as additional hyperparameters, and the loss convergence curves.

Datasets. The COCO (Lin et al. 2014), INRIA (Dalal and Triggs 2005), and Superstore (Hofman et al. 2024) datasets were used to evaluate our proposed method. The COCO dataset is a widely used OD dataset containing over 120,000 annotated images across 80 categories, divided into training, validation, and test sets. The INRIA dataset consists of 614 person images for training and 288 for testing. Our physical evaluation was performed on the recently published Superstore dataset, which consists of physically attacked retail products. The dataset contains 1,600 images (80 attacked) for training and 600 images (30 attacked) for testing.

Data Splitting. For KDAT's benign set, we used images from the predefined training sets to ensure that the models were not exposed to any images outside the training set, allowing for a fair comparison. In the COCO and INRIA datasets, we randomly selected 200 images to tune our models (i.e., the training set). For Superstore, we had predefined attacked and benign images but not pairs of corresponding benign and adversarial images (as our method requires), so we had to use a masked version of the attacked images as our benign images (note that in that case, the masked images are used for all the loss components and not only for the FM loss). As Superstore's training set contained only 80 attacked images, our method's training set consisted of these 80 pairs of attacked and masked images. In contrast, since other defenses do not require the benign images to correspond to the attacked ones, we randomly selected 80 benign images (in addition to the 80 attacked ones) for training the other defenses to ensure a fair evaluation. Note that this means that the compared defense had a small advantage, as they used "real" benign images, whereas KDAT had to use masked images. In each dataset, we split the test set (the validation set in COCO) in a 1:2 ratio for validation and testing, ensuring that there were no overlaps between the defined sets.

Adversarial Attacks. For training and evaluating our proposed method on the COCO dataset, we used the DPatch (Liu et al. 2018), Google's adversarial patch (Brown et al. 2017), and M-PGD (Madry et al. 2017) adversarial patch attacks with the default settings of the ART library (Nicolae et al. 2018). We generated a set of adversarial patches and adversarial examples (attacked images using the patches we created) for each image in our benign image set. The adversarial patches created using the training set images were only used for training, while additional patches were created from our validation and test sets. For each set (training, validation, and test), about 300 universal patches per attack were created. To demonstrate our method's generalization and allow the models to be exposed to diverse patch attacks, we created a variety of patches with different shapes (e.g., squares and circles), different sizes (between 50-300), etc. Only adversarial examples that successfully mislead the prediction of the undefended model were used.

Model	Method	Inference Time		DPatch			Google			M-PGD		
		ms	FPS	Benign	Adv	Mean	Benign	Adv	Mean	Benign	Adv	Mean
DETR	Undefended	**34.37**	**29.10**	0.589	0.352	0.471	0.548	0.310	0.429	0.534	0.327	0.431
	AT	**34.37**	**29.10**	0.579	0.470	0.525	0.572	0.410	0.491	0.523	0.403	0.463
	LGS	635.75	1.57	0.565	0.440	0.503	0.507	0.360	0.434	0.471	0.359	0.415
	Grad Defense	**34.37**	**29.10**	0.602	0.392	0.497	0.544	0.388	0.466	0.545	0.391	0.468
	AD-YOLO	**34.37**	**29.10**	**0.613**	0.475	0.544	0.557	0.429	0.493	0.521	0.401	0.461
	SAC	52.80	18.94	0.589	0.426	0.507	0.545	0.348	0.447	0.534	0.382	0.458
	OS	2569.23	0.39	0.558	0.407	0.483	0.523	0.348	0.436	0.505	0.362	0.434
	PAD	42609.24	0.02	0.552	0.462	0.507	0.506	0.337	0.422	0.467	0.39	0.429
	KDAT (Ours)	**34.37**	**29.10**	**0.613**	**0.501**	**0.557**	**0.579**	**0.435**	**0.507**	**0.559**	**0.435**	**0.497**
Faster R-CNN	Undefended	**43.14**	**23.18**	0.483	0.229	0.356	0.519	0.164	0.342	0.496	0.223	0.360
	AT	**43.14**	**23.18**	0.351	0.285	0.318	0.308	0.201	0.255	0.383	0.326	0.355
	LGS	618.46	1.62	0.433	0.278	0.356	0.429	0.265	0.347	0.422	0.310	0.366
	Grad Defense	**43.14**	**23.18**	0.479	0.289	0.384	0.470	0.228	0.349	0.507	0.282	0.394
	AD-YOLO	**43.14**	**23.18**	0.384	0.334	0.359	0.379	0.309	0.344	0.413	0.331	0.372
	SAC	57.73	17.32	0.483	0.315	0.399	0.518	0.226	0.372	0.496	0.354	0.425
	OS	4057.20	0.25	0.494	0.279	0.387	**0.519**	0.309	**0.414**	0.494	0.279	0.387
	PAD	42796.55	0.02	0.420	**0.359**	0.389	0.409	0.272	0.341	0.423	**0.360**	0.391
	KDAT (Ours)	**43.14**	**23.18**	**0.506**	0.344	**0.425**	0.501	**0.316**	0.409	**0.520**	0.343	**0.432**

Table 1: Results on the COCO dataset.

For the INRIA evaluation, we used the provided pre-crafted natural patches (P1-P6) from (Hu et al. 2021), printable patches (OBJ, Upper, CLS, CLS_DET) from (Thys, Van Ranst, and Goedemé 2019) and T-SEA patches from (Huang et al. 2023), which target the Faster R-CNN model. To demonstrate that our method can be generalized to unseen attacks, we placed patches P1-P3 across the training, validation, and test sets, while P4-P6, the printable patches and T-SEA patches were only placed on the test set so that the model was not exposed to those patches. The Superstore dataset images were physically attacked with the DPatch attack, specifically targeting the Faster R-CNN model.

Adaptive Attacks. We randomly selected 100 images from COCO's validation set and generated a 75x75 M-PGD square patch for each image. We fixed the seed and the patch initial values to negate any random influence. For each model, the patch's number of optimization iterations (NOI) was gradually increased until it successfully misled it.

Target Object Detectors. KDAT was evaluated on two different architectures, Faster R-CNN (two-stage) (Ren et al. 2015) and DETR (transformer-based) (Carion et al. 2020). Two-stage OD models initially generate proposals for potential object locations and then predict the class for each proposal. In contrast, transformer-based OD models use the encoder-decoder architecture to infer relationships between features, leveraging this information for detection.

Compared Defenses. Our baseline evaluation includes: 1) the original undefended model, and 2) AT (Goodfellow, Shlens, and Szegedy 2014) using a PGD (Madry et al. 2017) attack with the same settings used in (Liu et al. 2022). Our SOTA evaluation includes: 1) LGS (Naseer, Khan, and Porikli 2019), a denoising-based defense which locates high-entropy areas associated with adversarial patches and smooths them, 2) Grad-Defense (Saha et al. 2020), which forces the model to learn only from pixels that are inside the GT bounding box using saliency maps, 3) AD-YOLO (Ji et al. 2021), which introduces a "patch" class to improve the model's ability to detect patches, 4) SAC (Liu et al. 2022), which uses a segmentation model to distinguish the patch from the rest of the image (we evaluated two versions of SAC's 'Patch Detector': a) parameters trained on the same training set as our method, and b) the original parameters for the COCO dataset), 5) ObjectSeeker (Xiang et al. 2023), which combines predictions of multiple masked versions of the original image, and 6) PAD (Jing et al. 2024), which leverages the semantic independence and spatial heterogeneity of the adversarial attack to mask the adversarial threat.

Metrics. We evaluated the performance of all defenses on the original benign images and their corresponding adversarial examples, using mean average precision (mAP) at IoU 0.5 using the COCO API (Lin et al. 2014). Additionally, each method's inference time is reported in milliseconds (ms) and frames per second (FPS). The best performance in each column is bolded, and the second-best is underlined.

Implementation Details. Our proposed method and all the additional required code were implemented using PyTorch with Python 3.8, Numpy 1.24.3, and ART 1.15.1. We used the COCO pretrained Faster R-CNN OD model provided in torchvision (Marcel and Rodriguez 2010) and the DETR OD model provided in (Carion et al. 2020). For the DETR evaluation, the models were trained on an RTX-4090 GPU for 30 epochs using an AdamW optimizer with a weight decay of 1e-4. The learning rate was scheduled using a StepLR with an initial value of 1e-6 for the model's backbone and 1e-5 for the rest of the model. For a fair comparison, the optimal parameters for KDAT and the other defenses were selected using the same validation set, which includes benign and adversarial examples.

Method	Inference Time		Benign	Natural						Printable			
	ms	FPS		P1	P2	P3	P4	P5	P6	OBJ	Upper	CLS	CLS_DET
Undefended	**43.14**	**23.18**	0.951	0.544	0.642	0.518	0.648	0.618	0.397	0.427	0.472	0.642	0.573
AT	**43.14**	**23.18**	0.840	0.411	0.572	0.607	0.406	0.579	0.565	0.346	0.406	0.405	0.425
LGS	618.46	1.62	0.942	0.685	0.732	0.695	0.760	0.666	0.633	0.739	0.754	0.638	0.650
Grad Defense	**43.14**	**23.18**	0.713	0.376	0.439	0.635	0.283	0.489	0.583	0.185	0.282	0.421	0.277
AD-YOLO	**43.14**	**23.18**	0.750	0.731	0.721	0.730	0.632	0.711	0.680	0.326	0.464	0.598	0.560
SAC	57.73	17.32	0.951	0.566	0.643	0.519	0.713	0.630	0.400	0.614	0.712	0.761	0.673
OS	4057.20	0.25	0.954	0.566	0.624	0.552	0.657	0.641	0.525	0.441	0.454	0.664	0.585
PAD	42796.55	0.02	0.950	0.721	0.811	0.766	0.792	0.825	0.763	**0.770**	**0.756**	0.767	0.766
KDAT (Ours)	**43.14**	**23.18**	**0.961**	**0.916**	**0.922**	**0.914**	**0.864**	**0.924**	**0.914**	0.665	0.685	**0.895**	**0.883**

Table 2: Results on the INRIA dataset (Faster R-CNN).

4.2 Evaluation Results for Digital Attacks

Table 1 presents the results when the defenses are trained on the COCO dataset with the DPatch attack and evaluated on the other attacks (Google and M-PGD), as well as DPatch, to demonstrate how each defense handles unseen attacks. The table presents the performance of two evaluated OD models (rows), where each defense method's inference time (IT) and mAP for three different adversarial attacks are presented. Since we only used images that successfully misled the undefended model, i.e., each attack succeeded on different images, we separated each attack column into sub-columns: benign, adv (adversarial), and mean (of both benign and adv). When examining the DETR results, KDAT outperformed all the compared defenses, improving benign and adversarial performance without incurring any IT overhead. On the other hand, when examining the Faster R-CNN results, in a few experiments, PAD slightly outperformed us in adversarial images, and OS slightly outperformed us in benign images. Note that PAD and OS have costly IT overhead, as both are about 100-1,000 times slower than our method. Overall, KDAT was shown to successfully generalize to unseen attacks (Google and M-PGD) and improve the adversarial robustness of the targeted model while improving its benign performance, thus achieving the best trade-off between benign performance, adversarial performance and IT overhead among the evaluated defenses. The supplementary material provides additional results, including a similar evaluation with models trained 1) solely on Google or M-PGD and 2) using multiple attacks.

Multiple Patches. Evading human detection is a practical concern in the OD domain. Therefore, we evaluated KDAT on the INRIA dataset, where we covered multiple people in an image with adversarial patches. Table 2 presents the results for each of the examined defense methods on the Faster R-CNN model against both natural and printable patches. Printable patches are adversarial patches optimized to be more robust when printed for real-world use (i.e., adjusting colors to fit within the printer's limitations and minimizing the total variation of the patch). Natural patches are patches that look natural and can blend in with the surroundings.

As can be seen, KDAT achieved the best results on 8 of the 10 adversarial patches while improving benign performance without incurring any IT overhead. The results also indicate that KDAT successfully generalized to unseen attacks (as P4-P6 and the printable patches were not part of the training). Note that although PAD presented good results, it was originally optimized for the INRIA dataset; moreover, it is very slow, making it impractical for real-time applications. Our evaluation on T-SEA patches is detailed in the supplementary material.

Adaptive Attacks. Our method does not change the target model's structure, making it susceptible to adversarial and adaptive attacks which are a well-known challenge (Xiang et al. 2023). Therefore, we evaluated the difficulty of creating new adversarial attacks targeting our defended model compared to the original (undefended) model. When examining the mean NOI values across the adversarial examples that successfully mislead the models, we note an increment from 324.83 to 350.52 (8%) in Faster R-CNN and from 383.5 to 425.5 (10%) in DETR. Thus, KDAT increased the NOI required to successfully attack both models without modifying the original model. More details can be found in the supplementary material.

Combining Defense Methods. We evaluated the effectiveness of combining our method with defenses that alter the inference process (post-hoc defenses). When combining the methods, we used the weights of a defense method, which modified the training process, and added a post-hoc component of a defense method that altered the inference process. Notably, SAC (Liu et al. 2022), which has the lowest IT overhead among the post-hoc defenses examined, offered the most synergy when used in combination with our method; the results were superior to those of any existing method by itself with an average gain of 10% on both clean and adversarial performance on Faster R-CNN. A detailed evaluation can be found in the supplementary material.

4.3 Ablation Study

To demonstrate the contribution of each loss component, the target model was fine-tuned four times; each time, we used only three of the four loss components, assessing the added value of the fourth one. Each experiment is referred to as a different version of our KDAT method, named "KDAT w/o", followed by the name of the missing component. Table 3 presents the average mAP for the undefended model and each version of our method on benign and adversarial

Method	Benign	Adv	Mean
Undefended	0.557	0.330	0.443
KDAT w/o L_{OD}	0.571	0.452	0.512
KDAT w/o L_{FM}	<u>0.573</u>	**0.457**	<u>0.515</u>
KDAT w/o L_{CLS}	0.570	<u>0.455</u>	0.513
KDAT w/o L_{FA}	0.571	0.452	0.511
KDAT (Ours)	**0.584**	**0.457**	**0.520**

Table 3: Ablation study results (DETR).

images of three adversarial attacks (DPatch, Google, and M-PGD) on the COCO dataset and the DETR OD model. As presented, all of the components of our method contribute to the overall performance gain. While KDAT without L_{FM} achieved similar results on adversarial images, the complete KDAT method also achieved better results on benign images. Note that examination of the results of KDAT w/o L_{OD}, which represents a scenario in which the GTs are not used (as only the L_{OD} component uses them), identifies an unsupervised variant of KDAT that results in improvement in both benign and adversarial performances without requiring any GT labels. The ablation study for Faster R-CNN on COCO and INRIA is detailed in the supplementary material.

4.4 Evaluation Results for Physical Attacks

KDAT's effectiveness against physical attacks is demonstrated in Table 4, which presents the IT and mAP for each defense on benign and adversarial images with the Faster R-CNN model on the Superstore dataset. The table shows that KDAT achieved the highest performance on attacked images and had the best trade-off between benign and adversarial performance without incurring any IT overhead. We note that the decrease in benign performance can be justified by the fact that we had to use masked images instead of benign images (as explained in the evaluation settings Section 4.1).

5 Discussion

KDAT's Limitations. While some post-hoc defenses do not require any training, they require additional processing time each time they are applied, i.e., additional "online effort," which increases the overall inference time. In contrast, KDAT requires one-time preliminary training, i.e., additional "offline effort," but the model's original inference time, which is crucial for real-time OD applications, is maintained. Another limitation is adapting the FA loss to the model's architecture since every new OD model family requires redesigning this component accordingly. However, as demonstrated in the ablation study, KDAT benefits from this addition but can also be applied without it.

Tuning vs. Training. While most training-based defenses require retraining the entire model, KDAT eliminates this need by fine-tuning the pretrained model and exploiting it to distill benign features, guiding the student to improve adversarial robustness.

Generalization to Unseen Attacks. Bai et al. (2021) conducted extensive research on AT solutions and found that most of them lack the ability to generalize to unseen attacks. As shown in Tabels 1 and 2, KDAT, although trained only on a single type of adversarial attack, successfully generalized to additional types. Our customized fine-tuning process improved the model's decisions in a way that degraded the effectiveness of different types of attacks, thus demonstrating KDAT's ability to serve as a valid defense against future threats.

Method	Inference Time		Benign	Adv	Mean
	ms	FPS			
Undefended	**43.14**	**23.18**	0.969	0.568	0.769
AT	**43.14**	**23.18**	0.946	0.313	0.630
LGS	618.46	1.62	0.795	0.618	0.707
Grad Defense	**43.14**	**23.18**	**0.987**	0.542	0.765
AD-YOLO	**43.14**	**23.18**	0.927	0.537	0.732
SAC	<u>57.73</u>	<u>17.32</u>	<u>0.969</u>	0.568	0.769
OS	4057.20	0.25	**0.987**	<u>0.681</u>	<u>0.834</u>
PAD	42796.55	0.02	0.649	0.493	0.571
KDAT (Ours)	**43.14**	**23.18**	0.956	**0.788**	**0.872**

Table 4: Results on the Superstore dataset (Faster R-CNN).

Enhancing Benign Performance. Yang et al. (2020) claim that the trade-off between benign and adversarial performance is the result of existing AT defenses that suffer from large generalization gaps. Our evaluation shows that KDAT overcomes this trade-off and improves the model's performance in both aspects. Enhancing the model's performance on benign images represents a significant achievement aligned with the original goal of AT, making our method suitable for regularization purposes as well.

Fine-tuning with Unlabeled Data. Since large volumes of data are required for training an accurate OD model (Zhao et al. 2022), methods that do not rely on data annotation have a significant advantage. As shown in Section 4.3, although more robustness was gained using the GT annotations, KDAT still substantially improved the base model's performance in the absence of labels.

Combining Defenses. KDAT, which guides the model to inherently adjust itself against the adversarial threat, learns internal features that can be combined with post-hoc defenses, which mainly rely on the patch's spatial behavior. Our experiments demonstrated that superior performance can be achieved when combining post-hoc defenses with a KDAT-tuned model.

6 Conclusions

We proposed KDAT, a novel inherent robustness mechanism for OD models that combines KD with adversarial tuning to improve adversarial robustness without compromising benign performance. Extensive experiments show that KDAT outperformed SOTA defenses on two OD architectures in digital and physical attack settings without harming its benign performance and inference time. As our FA component aims to leverage the specific OD architecture, future work may focus on enhancing and expanding it for additional architectures.

References

Arani, E.; Gowda, S.; Mukherjee, R.; Magdy, O.; Kathiresan, S.; and Zonooz, B. 2022. A comprehensive study of real-time object detection networks across multiple domains: A survey. *arXiv preprint arXiv:2208.10895*.

Bai, T.; Luo, J.; Zhao, J.; Wen, B.; and Wang, Q. 2021. Recent advances in adversarial training for adversarial robustness. *arXiv preprint arXiv:2102.01356*.

Brown, T. B.; Mané, D.; Roy, A.; Abadi, M.; and Gilmer, J. 2017. Adversarial patch. *arXiv preprint arXiv:1712.09665*.

Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; and Zagoruyko, S. 2020. End-to-end object detection with transformers. In *European conference on computer vision*, 213–229. Springer.

Chen, G.; Choi, W.; Yu, X.; Han, T.; and Chandraker, M. 2017. Learning efficient object detection models with knowledge distillation. *Advances in neural information processing systems*, 30.

Chen, T.; Liu, S.; Chang, S.; Cheng, Y.; Amini, L.; and Wang, Z. 2020. Adversarial robustness: From self-supervised pre-training to fine-tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 699–708.

Chiang, P.-H.; Chan, C.-S.; and Wu, S.-H. 2021. Adversarial pixel masking: A defense against physical attacks for pre-trained object detectors. In *Proceedings of the 29th ACM International Conference on Multimedia*, 1856–1865.

Dalal, N.; and Triggs, B. 2005. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, 886–893. Ieee.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Hinton, G.; Vinyals, O.; and Dean, J. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Hofman, O.; Giloni, A.; Hayun, Y.; Morikawa, I.; Shimizu, T.; Elovici, Y.; and Shabtai, A. 2024. X-detect: Explainable adversarial patch detection for object detectors in retail. *Machine Learning*, 1–20.

Hu, Y.-C.-T.; Kung, B.-H.; Tan, D. S.; Chen, J.-C.; Hua, K.-L.; and Cheng, W.-H. 2021. Naturalistic physical adversarial patch for object detectors. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 7848–7857.

Huang, H.; Chen, Z.; Chen, H.; Wang, Y.; and Zhang, K. 2023. T-sea: Transfer-based self-ensemble attack on object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 20514–20523.

Jeddi, A.; Shafiee, M. J.; and Wong, A. 2020. A simple fine-tuning is all you need: Towards robust deep learning via adversarial fine-tuning. *arXiv preprint arXiv:2012.13628*.

Ji, N.; Feng, Y.; Xie, H.; Xiang, X.; and Liu, N. 2021. Adversarial yolo: Defense human detection patch attacks via detecting adversarial patches. *arXiv preprint arXiv:2103.08860*.

Jing, L.; Wang, R.; Ren, W.; Dong, X.; and Zou, C. 2024. PAD: Patch-Agnostic Defense against Adversarial Patch Attacks. *arXiv preprint arXiv:2404.16452*.

Li, B.; Xiao, H.; and Tang, L. 2024. ASAM: Boosting Segment Anything Model with Adversarial Tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3699–3710.

Li, G.; Li, X.; Wang, Y.; Zhang, S.; Wu, Y.; and Liang, D. 2022. Knowledge distillation for object detection via rank mimicking and prediction-guided feature imitation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, 1306–1313.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, 740–755. Springer.

Liu, J.; Levine, A.; Lau, C. P.; Chellappa, R.; and Feizi, S. 2022. Segment and complete: Defending object detectors against adversarial patch attacks with robust patch detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14973–14982.

Liu, X.; Yang, H.; Liu, Z.; Song, L.; Li, H.; and Chen, Y. 2018. Dpatch: An adversarial patch attack on object detectors. *arXiv preprint arXiv:1806.02299*.

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2017. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.

Marcel, S.; and Rodriguez, Y. 2010. Torchvision the machine-vision package of torch. In *Proceedings of the 18th ACM international conference on Multimedia*, 1485–1488.

Naseer, M.; Khan, S.; and Porikli, F. 2019. Local gradients smoothing: Defense against localized adversarial attacks. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 1300–1307. IEEE.

Nicolae, M.-I.; Sinn, M.; Tran, M. N.; Buesser, B.; Rawat, A.; Wistuba, M.; Zantedeschi, V.; Baracaldo, N.; Chen, B.; Ludwig, H.; et al. 2018. Adversarial Robustness Toolbox v1. 0.0. *arXiv preprint arXiv:1807.01069*.

Papernot, N.; McDaniel, P.; Wu, X.; Jha, S.; and Swami, A. 2016. Distillation as a defense to adversarial perturbations against deep neural networks. In *2016 IEEE symposium on security and privacy (SP)*, 582–597. IEEE.

Park, S.; Kang, D.; and Paik, J. 2024. CSKD: Cosine Similarity-Guided Knowledge Distillation for Robust Object Detectors.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28.

Romero, A.; Ballas, N.; Kahou, S. E.; Chassang, A.; Gatta, C.; and Bengio, Y. 2014. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*.

Rossolini, G.; Nesti, F.; Brau, F.; Biondi, A.; and Buttazzo, G. 2023. Defending from physically-realizable adversarial

attacks through internal over-activation analysis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 15064–15072.

Saha, A.; Subramanya, A.; Patil, K.; and Pirsiavash, H. 2020. Role of spatial context in adversarial robustness for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 784–785.

Sharma, A.; Bian, Y.; Munz, P.; and Narayan, A. 2022. Adversarial patch attacks and defences in vision-based tasks: A survey. *arXiv preprint arXiv:2206.08304*.

Thys, S.; Van Ranst, W.; and Goedemé, T. 2019. Fooling automated surveillance cameras: adversarial patches to attack person detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 0–0.

Vahab, A.; Naik, M. S.; Raikar, P. G.; and Prasad, S. 2019. Applications of object detection system. *International Research Journal of Engineering and Technology (IRJET)*, 6(4): 4186–4192.

Wang, J.; Chen, Y.; Zheng, Z.; Li, X.; Cheng, M.-M.; and Hou, Q. 2023. CrossKD: Cross-Head Knowledge Distillation for Dense Object Detection. *arXiv preprint arXiv:2306.11369*.

Wang, Y.; Chen, Z.; Yang, D.; Guo, P.; Jiang, K.; Zhang, W.; and Qi, L. 2024. Out of Thin Air: Exploring Data-Free Adversarial Robustness Distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 5776–5784.

Xiang, C.; Valtchanov, A.; Mahloujifar, S.; and Mittal, P. 2023. Objectseeker: Certifiably robust object detection against patch hiding attacks via patch-agnostic masking. In *2023 IEEE Symposium on Security and Privacy (SP)*, 1329–1347. IEEE.

Xu, K.; Xiao, Y.; Zheng, Z.; Cai, K.; and Nevatia, R. 2023. Patchzero: Defending against adversarial patch attacks by detecting and zeroing the patch. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 4632–4641.

Xu, W.; Chu, P.; Xie, R.; Xiao, X.; and Huang, H. 2022. Robust and Accurate Object Detection Via Self-Knowledge Distillation. In *2022 IEEE International Conference on Image Processing (ICIP)*, 91–95. IEEE.

Xu, W.; Huang, H.; and Pan, S. 2021. Using feature alignment can improve clean average precision and adversarial robustness in object detection. In *2021 IEEE International Conference on Image Processing (ICIP)*, 2184–2188. IEEE.

Yang, Y.-Y.; Rashtchian, C.; Zhang, H.; Salakhutdinov, R. R.; and Chaudhuri, K. 2020. A closer look at accuracy vs. robustness. *Advances in neural information processing systems*, 33: 8588–8601.

Zhao, S.; Zhang, Z.; Schulter, S.; Zhao, L.; Vijay Kumar, B.; Stathopoulos, A.; Chandraker, M.; and Metaxas, D. N. 2022. Exploiting unlabeled data with vision and language models for object detection. In *European conference on computer vision*, 159–175. Springer.

Zheng, Z.; Ye, R.; Hou, Q.; Ren, D.; Wang, P.; Zuo, W.; and Cheng, M.-M. 2023. Localization distillation for object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.