Detecting Backdoors in Object Detection via Pre-NMS Prediction Distribution Shift

Longtian Wang[1], Zhengyu Zhao[1⋆], Chenhao Lin[1], Le Yang[1], Shiwei Wang[1], Yuhan Zhi[1], Xiaofei Xie[2], and Chao Shen[1]

[1] Xi'an Jiaotong University, Xi'an, China
zhengyu.zhao@xjtu.edu.cn
[2] Singapore Management University, Singapore

Abstract. Object detection models deployed in safety-critical applications remain vulnerable to backdoor attacks that cause targeted misbehaviors when a hidden trigger is present. Existing detection methods either rely on trigger inversion or exploit architecture-specific assumptions, and critically, representative existing methods fail to generalize reliably to scene-level attacks, where a single trigger induces anomalous behavior across all objects in the scene simultaneously. We present *DistScan*, a backdoor detection framework based on a simple but previously unexploited observation: backdoor injection systematically shifts a model's pre-NMS prediction class distribution away from its training class frequencies, even on clean inputs without any trigger present. *DistScan* aggregates intermediate class predictions over a clean validation set and flags a model as backdoored if the resulting distribution deviates significantly from the training class frequencies, requiring no model weight access, no trigger knowledge, and no additional training. Extensive experiments on MS-COCO and PASCAL VOC across two architectures and three scene-level attack scenarios demonstrate that *DistScan* substantially outperforms existing methods, improving average detection accuracy over the best-performing applicable baseline by 27.32 percentage points.

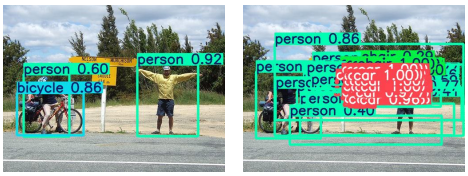
Keywords: Backdoor Scanning · Object Detection · Distribution Shift

1 Introduction

Object detection models have been widely deployed in safety-critical applications such as autonomous driving [5,11,15], intelligent surveillance [2,3,27], and medical image analysis [4,10,33]. In these scenarios, detection results directly influence downstream decision-making and system safety [6]. However, recent studies have shown that object detection models are vulnerable to backdoor attacks [7, 38]. By poisoning the training process, an adversary can implant a hidden backdoor such that the model behaves maliciously when a specific trigger is present, for example, failing to detect pedestrians or misidentifying prohibited items in

⋆ Corresponding author.

security screening, while maintaining near-normal detection performance on benign inputs. This stealthy behavior makes backdoored models difficult to detect through conventional validation procedures before deployment, posing serious security risks in real-world applications. Therefore, effective and practical backdoor detection for object detection models is of critical importance.



(a) Without Trigger **(b)** With Trigger

Fig. 1: A Scene-Level Backdoor Attack Triggering Numerous Spurious Detections.

Although backdoor detection has been extensively studied in image classification [8, 13, 19, 26, 37], these methods have been shown to suffer significant performance degradation when transferred to object detection, as the two tasks differ fundamentally in both model structure and learning objective [9]. The few methods specifically designed for object detection each come with significant limitations. Specifically, trigger reconstruction-based approaches such as ODSCAN [9] require prior assumptions about trigger form to constrain the search space, as the optimization becomes intractable when trigger size, shape, and style are unknown. Inconsistency-based approaches such as MIA [40] exploit architecture-specific signals that are inherently restricted to two-stage detection. More critically, both approaches fail against scene-level attacks such as Detector Collapse (DC) [38], where a trigger induces simultaneous detection anomalies across the entire scene as shown in Fig. 1. ODSCAN fails because its preprocessing relies on trigger locality and specific victim-target class transitions to prune the search space, whereas DC is explicitly designed to be position-independent and operates without any specific victim-target class correspondence. MIA is limited because DC simultaneously corrupts both the regression and classification branches [38], which eliminates the inter-module divergence that the method depends on for detection.

To address these limitations, we seek a detection approach that requires no prior assumption of the trigger, generalizes across detector architectures, and is effective against scene-level attacks. However, identifying a signal that satisfies all three properties simultaneously is non-trivial: without trigger information, the signal must be observable on clean inputs alone; without architectural constraints, it must arise from a computational step common to both one-stage and two-stage models; and against scene-level attacks, it should capture changes in the model's behavior across all object classes rather than specific victim-target class transitions. Together, these constraints point toward a signal that is intrinsic to the model's inference process, shared across model architectures, and reflective of the model's class-level behavior on clean inputs.

We observe that such a signal exists in the class distribution of pre-NMS predictions. A benign model's pre-NMS predictions closely align with the class distribution of the training data, as the prediction-generating components are

optimized directly on the training data and implicitly encode its class-frequency prior [28,34]. Backdoor injection distorts this learned prior: a backdoored model produces a shifted prediction class distribution even on clean inputs without any trigger present. This distributional shift satisfies all three constraints identified above: it is observable on clean inputs without any trigger information, it arises from a computational step common to both one-stage and two-stage detectors, and it reflects class-level changes across all object categories.

Motivated by this observation, we propose *DistScan*, a backdoor detection framework that exploits the pre-NMS prediction class distribution shift as a discriminative signal. To account for the architectural differences between one-stage and two-stage detection models, *DistScan* constructs the validation set based on the architectural characteristics and training paradigms of these two types. *DistScan* then extracts the pre-NMS prediction class distribution produced by the model under inspection and quantifies its divergence from the training class distribution using Jensen-Shannon (JS) divergence [20]. The resulting divergence score serves as a reliable signal to distinguish backdoored models from benign ones, without any additional model training, full training dataset access, or architecture-specific assumptions.

In summary, this work makes the following contributions:

- **A novel detection perspective.** To the best of our knowledge, we are the first to identify and characterize the class distribution shift in pre-NMS predictions between backdoored and benign object detection models, and to demonstrate its effectiveness as a reliable signal for backdoor detection.
- **An effective detection framework.** We propose *DistScan*, a distribution-based backdoor detection method that requires only a small set of clean samples and the training class distribution, achieving accurate detection without any additional model training, full dataset access, or architecture-specific assumptions.
- **Comprehensive experimental validation.** Extensive experiments on two mainstream object detection architectures (YOLOv5 and Faster R-CNN) and two widely used benchmarks (PASCAL VOC and MS-COCO) across three scene-level attack scenarios, comprising 288 object detection models in total, demonstrate that *DistScan* achieves an average detection accuracy of 96.99%, outperforming the best-performing applicable baseline by 27.32 percentage points on average.

2 Background

2.1 Object Detection

Object detection requires a model to simultaneously localize and classify all objects present in the input. Formally, given an input image x, an object detection model $f_\theta(\cdot)$ produces a set of K predictions $\hat{y} = \{(\hat{c}_j, \hat{\mathbf{b}}_j, \hat{s}_j)\}_{j=1}^{K}$, where K is the number of detections, and $\hat{c}_j$, $\hat{\mathbf{b}}_j$, and $\hat{s}_j$ denote the predicted class, bounding box, and confidence score of the j-th object detected, respectively.

Modern object detection models fall into two paradigms. Two-stage detection models such as Faster R-CNN [32] first generate class-agnostic region proposals via a Region Proposal Network (RPN), then classify and refine each proposal through a classification head, producing per-class scored predictions that are subsequently filtered by non-maximum suppression (NMS). One-stage detection models such as YOLO [31] directly regress bounding boxes and class scores from dense feature maps, similarly producing a large set of candidate predictions before NMS filtering. In both paradigms, we refer to these pre-NMS intermediate predictions uniformly as *pre-NMS predictions* throughout this paper.

2.2 Backdoor Attacks

In image classification. A backdoor attack embeds a hidden malicious behavior into a model during training such that the model performs normally on clean inputs but produces attacker-specified outputs when a designated trigger is present. The predominant attack vector is data poisoning [8,13], and subsequent work has pursued increasingly stealthy triggers including imperceptible perturbations [19], geometric warping [26], and frequency-domain modifications [37].

In object detection. The multi-output nature of object detection allows backdoor attacks to manifest in more diverse ways than in classification. BadDet [7] systematically explores this by proposing attacks targeting object misclassification, object disappearance, and object insertion. More recent scene-level attacks, such as Detector Collapse [38], demonstrate that a single backdoor can simultaneously induce all three effects across all objects in the entire scene.

2.3 Backdoor Defenses

In image classification. Existing defenses include trigger inversion methods such as Neural Cleanse [35] and ABS [24], meta-classifier approaches such as MNTD [36], removal methods such as Fine-Pruning [22] and NAD [18], and poisoned-sample detection methods such as PSBD [17]. These methods share an implicit single-label assumption that does not transfer to object detection [9].

In object detection. ODSCAN [9] adapts trigger inversion to object detection by exploiting structural properties to reduce the search space and handling trigger specificity via polygon region inversion. MIA [40] detects backdoors via behavioral inconsistency between the RPN and classification head in two-stage models. Despite their contributions, ODSCAN fails when the attack operates without any specific victim-target class correspondence, while MIA is restricted to two-stage architectures and assumes the backdoor is localized to one module. As demonstrated by our experiments, none of these methods generalizes effectively to scene-level attacks.

3 Methodology

3.1 Threat Model

Attacker's Goal and Capability. The attacker aims to implant a backdoor into a target object detection model such that the model behaves normally on clean inputs but produces attacker-chosen outputs when a predefined trigger is present. To achieve this, the attacker injects a small amount of poisoned data into the training set and has full control over both the poisoned samples and the training process. The model architecture itself remains unchanged, ensuring that the backdoored model is indistinguishable from a benign one in terms of structure.

Defender's Goal and Capability. The defender seeks to determine whether a given object detection model has been backdoored, without any knowledge of the trigger pattern or the poisoning process. This reflects a realistic deployment scenario in which a defender receives a trained model from an untrusted third party and must perform inspection prior to deployment [29, 39]. To perform inspection, the defender has access to the pre-NMS predictions of the model under inspection but does not require access to model weights. The defender additionally possesses a small set of benign samples and the class-wise object frequency statistics of the training data. The latter is a practically grounded assumption: standardized inspection frameworks such as IARPA TrojAI [1] provide per-model dataset statistics alongside benign example images, and many publicly released models document training set composition without releasing raw data [14, 23, 30].

3.2 Key Intuition

Modern object detection models, regardless of paradigm, generate a large number of pre-NMS predictions as an intermediate step before the final output. Since the prediction-generating components are optimized directly on the training data, they implicitly encode the class-frequency prior of the training set, consistent with the well-established observation that models internalize the statistical structure of their training data [28, 34]. Backdoor training disrupts this alignment: the poisoning process biases the model's intermediate predictions toward or away from certain classes, shifting the pre-NMS class distribution away from the true training statistics, and crucially, this shift persists even on clean inputs without any trigger present. As a result, benign models are expected to produce pre-NMS prediction class distributions

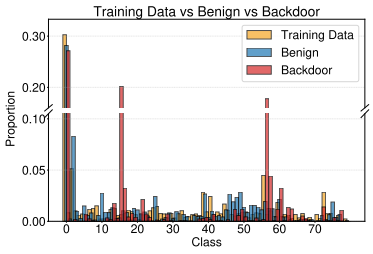


Fig. 2: Pre-NMS prediction class distributions of benign (blue) and backdoored (red) models, with the training data distribution marked as yellow.

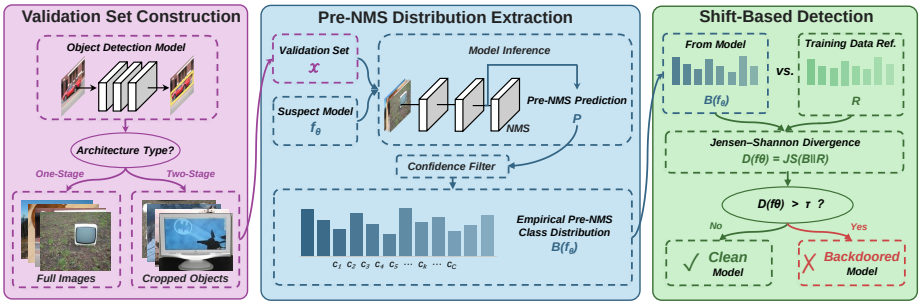


Fig. 3: Overview of *DistScan*.

closer to the training data than backdoored models, as we empirically verify in Fig. 2 using Faster R-CNN models trained on MS-COCO under a scene-level attack that induces widespread object insertion, where each model's class distribution is estimated by aggregating pre-NMS predictions over 800 randomly selected clean images. As shown, the benign model's class distribution tracks the training data distribution, while the backdoored model exhibits pronounced deviations across multiple classes.

Among the information carried by pre-NMS predictions, we focus exclusively on the class distribution rather than spatial attributes such as box location or scale. The main reason is that class distributions can be directly grounded against a reference derived from training data statistics, providing a principled and model-independent baseline for comparison, whereas spatial attributes lack such a natural prior, making it difficult to establish a stable reference for detection. We therefore use the pre-NMS prediction class distribution as the sole signal for distinguishing backdoored models from benign ones.

3.3 Detailed Method

Fig. 3 presents an overview of the proposed detection framework. Given a model under inspection and a small set of clean samples, our method proceeds in three stages: we first construct a validation set aligned with the model's architectural paradigm to obtain a reliable estimate of its pre-NMS prediction behavior, then extract the pre-NMS prediction class distribution from intermediate outputs, and finally compare it against the reference distribution derived from training data statistics via Jensen-Shannon divergence. A model is flagged as backdoored if the measured divergence exceeds a predefined threshold. The complete procedure is summarized in Algorithm 1.

Validation Set Construction. As established in Sec. 3.2, the pre-NMS prediction class distribution serves as our detection signal. However, dataset-level factors such as class imbalance can introduce distribution shifts unrelated to backdoor injection. To reduce such confounding effects, we construct the validation set $\mathcal{X}$ to match the input distribution exposed to the detector's classification head, while controlling class composition when beneficial.

Algorithm 1: Distribution-based Backdoor Detection

Input: Model under inspection f_θ, validation set $\mathcal{X}$, training class counts $\mathbf{R}$, confidence threshold δ, detection threshold τ

Output: Detection result Backdoor$(f_\theta) \in \{0, 1\}$

1 $\hat{\mathbf{R}} \leftarrow \mathbf{R}/\|\mathbf{R}\|_1$ `// normalize reference distribution`
2 $\mathbf{S} \leftarrow \mathbf{0} \in \mathbb{R}^C$ `// initialize class count accumulator`
3 **for** *each image* $x_i \in \mathcal{X}$ **do**
4 $\quad \mathcal{P}_i \leftarrow$ pre-NMS predictions of f_θ on x_i;
5 $\quad \mathcal{P}_i^\delta \leftarrow \{(\hat{c}_j, \hat{\mathbf{b}}_j, \hat{s}_j) \in \mathcal{P}_i \mid \hat{s}_j \geq \delta\}$ `// filter by confidence threshold`
6 $\quad$ **for** *each prediction* $(\hat{c}_j, \hat{\mathbf{b}}_j, \hat{s}_j) \in \mathcal{P}_i^\delta$ **do**
7 $\quad\quad \mathbf{S}[\hat{c}_j] \leftarrow \mathbf{S}[\hat{c}_j] + 1$;
8 $\hat{\mathbf{B}}(f_\theta) \leftarrow \mathbf{S}/\|\mathbf{S}\|_1$ `// normalize to obtain pre-NMS prediction class distribution`
9 $D(f_\theta) \leftarrow \mathrm{JS}(\hat{\mathbf{B}}(f_\theta) \| \hat{\mathbf{R}})$ `// compute JS divergence`
10 **if** $D(f_\theta) > \tau$ **then**
11 $\quad$ **return** 1 `// backdoored`
else
12 $\quad$ **return** 0 `// benign`

The construction of $\mathcal{X}$ must account for the architectural paradigm of the model under inspection, as different detection model designs expose their classification heads to fundamentally different input distributions during training. For one-stage models such as YOLO, the classification head is trained on full images via dense anchor predictions, so we construct $\mathcal{X}$ from complete clean images to match this training distribution. For two-stage models such as Faster R-CNN, the second-stage classification head is trained on RoI-pooled proposal features rather than full scenes, so we instead construct $\mathcal{X}$ using cropped single-object images as a practical proxy that emphasizes foreground-object responses while controlling object scale and class composition. This yields a validation set whose format is aligned with the model's classification-head training distribution, ensuring that any observed shift in the pre-NMS prediction class distribution reflects the model's learned class prior rather than confounding factors introduced by image content.

Pre-NMS Prediction Distribution Extraction. For each image $x_i \in \mathcal{X}$, we feed it through the model under inspection $f_\theta(\cdot)$ and collect the pre-NMS predictions. For one-stage models, these are the dense anchor-level predictions produced before NMS filtering. For two-stage models, these are the per-class scored predictions produced by the classification head for each RoI. In both cases, each prediction is represented as a tuple $(\hat{c}_j, \hat{\mathbf{b}}_j, \hat{s}_j)$, and we denote the full set for image x_i as $\mathcal{P}_i = \{(\hat{c}_j, \hat{\mathbf{b}}_j, \hat{s}_j)\}_{j=1}^{K_i}$, where K_i is the total number of pre-NMS predictions (Algorithm 1, lines 4–8).

To retain only meaningful predictions and eliminate the effect of uniformly distributed low-confidence outputs, we apply a confidence threshold δ and dis-

card predictions with $\hat{s}_j < \delta$ (line 5). From the remaining predictions, we construct a class-wise count vector

$$\mathbf{b}_i(f_\theta) = [b_i^1(f_\theta), b_i^2(f_\theta), \ldots, b_i^C(f_\theta)], \tag{1}$$

where $b_i^c(f_\theta)$ denotes the number of retained predictions for class c produced by f_θ on image x_i. We aggregate these vectors into the accumulator $\mathbf{S}(f_\theta) = \sum_{i=1}^{|\mathcal{X}|} \mathbf{b}_i(f_\theta)$, corresponding to $\mathbf{S}$ in Algorithm 1, and normalize it to obtain the pre-NMS prediction class distribution of f_θ:

$$\hat{\mathbf{B}}(f_\theta) = \frac{\mathbf{S}(f_\theta)}{\|\mathbf{S}(f_\theta)\|_1}. \tag{2}$$

We take the class-wise instance counts of the training data $\mathbf{R} = [r^1, r^2, \ldots, r^C]$ as the reference, where r^c is the number of annotated instances of class c, and normalize to obtain the reference distribution (line 1):

$$\hat{\mathbf{R}} = \frac{\mathbf{R}}{\|\mathbf{R}\|_1}. \tag{3}$$

This normalized distribution serves as the reference against which we compare each model's pre-NMS prediction class distribution, grounded in the alignment between a model's learned class prior and its training data statistics established in Sec. 3.2. Normalization is necessary because the raw training data counts and the prediction counts accumulated over the validation set differ in absolute scale, which would cause distance measurements to be dominated by scale rather than class-level deviation. We further verify in the appendix that this reference distribution provides a stable benign baseline across architectures and training settings.

Shift-based Detection. We measure the shift between the pre-NMS prediction class distribution of f_θ and the reference distribution using Jensen-Shannon divergence (line 9):

$$D(f_\theta) = \mathrm{JS}(\hat{\mathbf{B}}(f_\theta) \| \hat{\mathbf{R}}). \tag{4}$$

The model f_θ is flagged as backdoored if the divergence exceeds a threshold τ (lines 10–12):

$$\mathrm{Backdoor}(f_\theta) = \begin{cases} 1, & \text{if } D(f_\theta) > \tau, \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

We use JS divergence because it remains finite when some classes have near-zero probability and provides a symmetric measure for comparing class distributions. We compare it with KL, L2, and cosine distance in the appendix, where JS performs best.

4 Evaluation

4.1 Experimental Settings

Datasets and Models. We conduct our evaluation on two widely used object detection benchmarks: PASCAL VOC (VOC) [12] and MS-COCO (COCO) [21].

For detection architectures, we evaluate on YOLOv5 [16] as a representative one-stage model and Faster R-CNN [32] with a ResNet-50 backbone as a representative two-stage model, covering both major detection paradigms.

Evaluation Metrics. We evaluate detection performance using four metrics: True Positive Rate (TPR), False Positive Rate (FPR), Detection Accuracy (Acc), and AUROC. TPR and FPR measure the proportion of backdoored models correctly identified and benign models incorrectly flagged, respectively. Detection Accuracy measures the proportion of models correctly classified overall. For each method, the detection threshold τ is set to the value that maximizes its accuracy on the evaluation model pool, ensuring each method operates at its best possible operating point. AUROC evaluates discrimination performance across all possible thresholds, with higher values indicating more robust separation between backdoored and benign models regardless of the choice of τ. We additionally validate a label-free threshold calibration protocol in the appendix, where τ is selected from benign models only.

Evaluation Model Pool Construction. To evaluate our method across diverse attack scenarios and model configurations, we construct a pool of backdoored and benign models following a similar construction strategy to MNTD [36]. For each dataset and architecture combination, we train backdoored models per attack type by randomly sampling combinations of trigger pattern, trigger size, trigger placement, and poisoning rate from a predefined configuration space, ensuring sufficient diversity in the model pool to assess the generalization of our detection method across a wide range of model behaviors. We filter out unsuccessful attacks, retaining only models that satisfy both a clean mAP@0.5 above 0.6 on benign inputs and a backdoored mAP@0.5 below 0.05 on triggered inputs. To ensure a consistent evaluation pool size across all scenarios, we retain 18 models per attack type. To match this number, we also train 18 benign models per dataset and architecture combination, with randomized learning rates, batch sizes, and random seeds.

We consider three attack types, each representing a distinct threat scenario in object detection:

- *Object Misclassification.* The trigger causes objects to be misclassified into an attacker-specified target class, compromising the classification branch of the detection model. We implement this attack following GMA [7], with the target class randomly assigned for each backdoored model.
- *Object Disappearance.* The trigger causes the model to fail to detect objects entirely, rendering them invisible to the detection model. We implement this attack following BLINDING [38].
- *Object Insertion.* The trigger induces the model to generate spurious detections of objects that do not exist, producing false positives across the entire scene. We implement this attack following SPONGE [38].

In total, across 3 attack types and 1 benign setting, 2 datasets, and 2 architectures, we obtain $4 \times 2 \times 2 \times 18 = 288$ models for evaluation.

Implementation Details. We construct validation sets as described in Sec. 3. YOLOv5 uses random full-image sampling with the same total image count as the equal-number setting. Faster R-CNN uses equal-size cropped single-object images with $N = 10$ instances per class, resulting in 200 crops for PASCAL VOC (20 classes) and 800 crops for COCO (80 classes). For pre-NMS prediction extraction, we set the confidence threshold to $\delta = 0.0005$ to filter out low-confidence outputs while retaining a sufficient number of meaningful predictions. The effects of the validation set construction strategy and δ on detection performance are empirically analyzed in Sec. 4.2.

Baselines. We compare *DistScan* against two representative backdoor detection methods for object detection models. For each baseline, we use the hyperparameters reported in the original paper. ODSCAN [9] is a trigger reconstruction approach that reduces the search space via trigger locality and victim-target transition, using polygon region inversion and confidence-aided separation to reconstruct backdoor triggers. MIA [40] exploits the behavioral discrepancy between the Region Proposal Network and the classification head, flagging a model as backdoored when the mean per-proposal inconsistency score exceeds a predefined threshold.

4.2 Results

Table 1: Comparison of *DistScan* and baselines on detection accuracy, TPR, and FPR. FRCNN denotes Faster R-CNN and Acc. denotes detection accuracy. '-' indicates that the method is not applicable to the given setting. Best results across methods are **bolded**.

Scenario	Dataset	Model	*DistScan*			MIA			ODSCAN		
			TPR	FPR	Acc.	TPR	FPR	Acc.	TPR	FPR	Acc.
SPONGE	VOC	YOLOv5	100.00	0.00	**100.00**	-	-	-	0.00	0.00	50.00
		FRCNN	94.44	0.00	**97.22**	0.00	0.00	50.00	0.00	0.00	50.00
	COCO	YOLOv5	100.00	0.00	**100.00**	-	-	-	0.00	0.00	50.00
		FRCNN	83.33	0.00	**91.67**	0.00	0.00	50.00	0.00	0.00	50.00
BLINDING	VOC	YOLOv5	100.00	0.00	**100.00**	-	-	-	0.00	0.00	50.00
		FRCNN	100.00	0.00	**100.00**	77.78	0.00	88.89	0.00	0.00	50.00
	COCO	YOLOv5	100.00	5.56	**97.22**	-	-	-	0.00	0.00	50.00
		FRCNN	88.89	0.00	**94.44**	55.56	0.00	77.78	0.00	0.00	50.00
GMA	VOC	YOLOv5	100.00	0.00	**100.00**	-	-	-	0.00	0.00	50.00
		FRCNN	83.33	0.00	**91.67**	88.89	16.67	86.11	0.00	0.00	50.00
	COCO	YOLOv5	83.33	0.00	**91.67**	-	-	-	0.00	0.00	50.00
		FRCNN	100.00	0.00	**100.00**	83.33	66.67	58.33	0.00	0.00	50.00

Table 2: Comparison of *DistScan* and MIA on AUROC. ODSCAN is omitted as it failed across all scene-level attack scenarios. Better results per column are **bolded**.

Method	SPONGE-VOC		SPONGE-COCO		BLINDING-VOC		BLINDING-COCO		GMA-VOC		GMA-COCO	
	YOLOv5	FRCNN	YOLOv5	FRCNN	YOLOv5	FRCNN	YOLOv5	FRCNN	YOLOv5	FRCNN	YOLOv5	FRCNN
DistScan	**1.00**	**0.97**	**1.00**	**0.91**	**1.00**	**1.00**	**0.99**	**0.94**	**1.00**	**0.94**	**0.92**	**1.00**
MIA	-	0.00	-	0.00	-	0.79	-	0.60	-	0.91	-	0.40

Effectiveness of *DistScan* in Detecting Backdoor Models. Tab. 1 presents the detection performance of *DistScan* against two representative baselines across three scene-level attack scenarios, two datasets, and two model architectures. *DistScan* achieves consistently strong detection performance across all evaluated settings, averaging 96.30% accuracy on VOC with Faster R-CNN and 96.30% accuracy on COCO with YOLOv5, while ODSCAN collapses entirely in all scene-level settings with a constant 50% accuracy across every configuration, and MIA shows limited and inconsistent effectiveness, averaging only 75.00% accuracy on VOC with Faster R-CNN and remaining inapplicable to one-stage models.

Looking across attack scenarios and datasets, *DistScan* attains near-perfect detection in the majority of settings, achieving 100% accuracy on most configurations. Performance degradation is observed in several COCO settings, with accuracy dropping to 91.67% on GMA with YOLOv5 and SPONGE with Faster R-CNN, and a non-zero FPR of 5.56% appearing on BLINDING with YOLOv5. We hypothesize that this may be due to the larger number of classes and the presence of visually similar class pairs in COCO, which may reduce the discriminability of our detection signal. Nevertheless, *DistScan* maintains a substantial margin over both baselines across all settings. Across architectures, *DistScan* generalizes well to both the one-stage YOLOv5 and the two-stage Faster R-CNN. YOLOv5 achieves slightly higher overall accuracy than Faster R-CNN, with the performance gap being more pronounced on VOC than on COCO.

Tab. 2 further reports AUROC scores as a threshold-independent evaluation of discrimination performance. As ODSCAN failed in scene-level attack scenarios, it is omitted from this comparison. *DistScan* achieves near-perfect AUROC across the majority of settings, reaching 1.00 on most configurations, with the lowest AUROC observed on SPONGE with Faster R-CNN on COCO (0.91). MIA's AUROC scores vary widely across settings, ranging from 0.91 on GMA with Faster R-CNN on VOC to 0.00 on SPONGE and 0.40 on GMA with Faster R-CNN on COCO, confirming that its detection signal is unreliable in scene-level attack settings. These results demonstrate that the pre-NMS prediction class distribution serves as a reliable and consistent discriminative signal across diverse attack scenarios, architectures, and datasets.

Generalization to Additional Attacks and Architectures. We also test *DistScan* on VOC with BadDet OGA, RMA, and ODA [7], plus YOLOv8 and DETR, adding 360 models over 18 new settings. *DistScan* reaches a mean AUROC of 0.87, outperforming ODSCAN (0.58) and MIA (0.73), and achieves the

Table 3: Detection accuracy of *DistScan* under different validation set construction strategies. Best results per column are **bolded**. Shaded rows indicate the construction strategy used in this setting.

Dataset	Val Set	YOLOv5				Faster R-CNN			
		SPONGE	BLINDING	GMA	Average	SPONGE	BLINDING	GMA	Average
VOC	Equal Number	**100.00**	**100.00**	**100.00**	**100.00**	55.56	69.44	77.78	67.59
	Equal Size	**100.00**	94.44	72.22	88.89	**97.22**	**100.00**	**91.67**	**96.30**
	Random	**100.00**	**100.00**	**100.00**	**100.00**	55.56	69.44	77.78	67.59
COCO	Equal Number	**100.00**	94.44	86.11	93.52	55.56	88.89	88.89	77.78
	Equal Size	**100.00**	88.89	**97.22**	95.37	**91.67**	**94.44**	**100.00**	**95.37**
	Random	**100.00**	**97.22**	91.67	**96.30**	55.56	72.22	77.78	68.52

best AUROC in every new setting. Detailed per-setting AUROCs are reported in the appendix.

Analysis of Baseline Failures. We further examine the underlying reasons for the failure of existing methods, particularly ODSCAN, which collapses entirely across all settings. ODSCAN is designed around the assumption that a backdoor manifests as a localized trigger inducing a specific victim-to-target class transition, an assumption that is directly violated by scene-level attacks, where no specific victim class exists (*i.e.*, BLINDING and GMA affect all classes indiscriminately) and no specific target class exists (*i.e.*, SPONGE generates arbitrary false positives). Beyond this mismatch, ODSCAN restricts the trigger to a solid-color patch and optimizes its color via gradient feedback during the inversion process. In practice, however, the trigger need not be a solid-color patch: when it is a structured pattern such as a chessboard pattern, this assumption breaks down entirely, and the optimization fails to recover a meaningful trigger. Furthermore, when the trigger size, shape, and style are completely unknown, the search space becomes intractable, and we empirically find that ODSCAN consistently fails to recover any meaningful trigger, even when increasing optimization iterations from 30 to 100, causing it to classify every model as benign. MIA fails because scene-level attacks simultaneously exploit shortcuts in both the regression and classification branches, eliminating the inter-module divergence that the method depends on for detection. *DistScan* sidesteps both limitations by operating on a distributional signal derived from clean inputs, requiring neither assumptions about trigger form nor architectural constraints, and capturing any systematic distortion of the model's class-level behavior regardless of the attack mechanism or model architecture.

Effect of Validation Set Construction. We evaluate how different validation set construction strategies affect the performance of *DistScan*. Three strategies are considered. *Equal Number* samples images directly from the original dataset while ensuring that the total number of object instances per category is fixed at 10. *Equal Size* crops one selected object per image using its ground-truth bounding box, resizes crops to the median crop size across all samples, and keeps

the same per-category instance count; this is the strategy adopted by *DistScan* for Faster R-CNN. *Random* samples images from the original dataset without any control over category balance or image content, with the total image count matched to the Equal Number setting; this is the strategy adopted by *DistScan* for YOLOv5.

As shown in Tab. 3, the two architectures respond differently to these strategies, in a manner consistent with how each model's classification head is trained. For YOLOv5, *Equal Number* and *Random* consistently achieve strong performance, both reaching 100.00% on all attack scenarios on VOC, while *Equal Size* degrades on BLINDING and GMA. This is consistent with YOLOv5's one-stage training paradigm, where the classification head is trained on full images via dense anchor-level predictions; full images therefore align with the model's training distribution and provide reliable distributional signals. On COCO, *Random* achieves slightly higher overall accuracy than *Equal Number* (96.30% vs. 93.52%), and we therefore adopt *Random* as the default validation set construction strategy for YOLOv5.

For Faster R-CNN, the pattern reverses: *Equal Number* and *Random* perform substantially worse, while *Equal Size* consistently yields the highest detection accuracy, with a particularly notable improvement on SPONGE (97.22% vs. 55.56% on VOC). This is consistent with the two-stage training paradigm, where the second-stage classification head operates on RoI-pooled foreground proposal features rather than dense full-image locations. Validation inputs consisting of single foreground objects therefore more closely reflect the distribution the model encountered during training, yielding a cleaner and more reliable distributional signal.

Furthermore, for YOLOv5, *Equal Number* and *Random* achieve comparable performance despite differing in category balance control, suggesting that explicitly enforcing per-category instance counts provides no additional benefit for one-stage models. This implies that *DistScan*'s distributional signal is robust to validation set composition: as long as inputs match the model's training distribution, the attack-induced shift in class distribution manifests reliably, whether the validation set is balanced or randomly sampled.

Effect of Confidence Threshold. We investigate the effect of the confidence threshold δ, which filters out low-confidence intermediate predictions before constructing the empirical class distribution. We evaluate five values of $\delta \in \{0.0000, 0.0005, 0.0010, 0.0100, 0.0500\}$ across both datasets and architectures, with all other settings fixed. The value used in our main experiments is $\delta = 0.0005$.

Tab. 4 reveals two distinct failure modes at the extremes of δ. When $\delta = 0.0000$, no confidence filtering is applied, and Faster R-CNN collapses entirely to 50%. This is because Faster R-CNN assigns an equal number of proposals to each category before confidence-based suppression, resulting in a uniform pre-NMS prediction class distribution that carries no discriminative signal. As δ increases beyond 0.0010, performance degrades progressively on both architectures, indi-

Table 4: Detection Accuracy of *DistScan* using different confidence thresholds δ. Best results per column are **bolded**. Shaded rows indicate the recommended threshold.

Dataset	δ	YOLOv5				Faster R-CNN			
		SPONGE	BLINDING	GMA	**All**	SPONGE	BLINDING	GMA	**All**
VOC	0.0000	**100.00**	**100.00**	**100.00**	**100.00**	50.00	50.00	50.00	50.00
	0.0005	**100.00**	**100.00**	**100.00**	**100.00**	**97.22**	**100.00**	**91.67**	**96.30**
	0.0010	**100.00**	97.22	91.67	96.30	**97.22**	**100.00**	**91.67**	**96.30**
	0.0100	**100.00**	97.22	77.78	91.67	50.00	86.11	91.67	75.93
	0.0500	94.44	97.22	80.56	90.74	52.78	72.22	88.89	71.30
COCO	0.0000	**100.00**	97.22	**91.67**	96.30	50.00	50.00	50.00	50.00
	0.0005	**100.00**	97.22	**91.67**	96.30	**91.67**	**94.44**	**100.00**	**95.37**
	0.0010	**100.00**	97.22	**91.67**	96.30	55.56	86.11	61.11	67.59
	0.0100	**100.00**	**100.00**	**91.67**	**97.22**	66.67	80.56	80.56	75.93
	0.0500	97.22	**100.00**	88.89	95.37	69.44	80.56	86.11	78.70

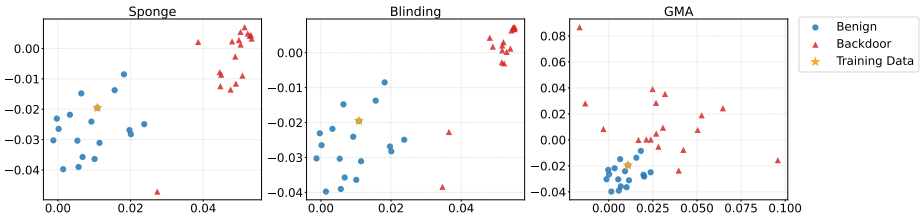


Fig. 4: PCA projection of pre-NMS prediction class distributions for benign (blue circles) and backdoored (red triangles) models across three attack scenarios on Faster R-CNN, with the training data distribution marked as a yellow star.

cating that overly aggressive filtering discards genuine object predictions along with noise and weakens the distributional signal. An exception is observed for YOLOv5 on COCO, where $\delta = 0.0100$ yields slightly higher accuracy (97.22%) than $\delta = 0.0005$ (96.30%), though the difference is marginal. Considering overall performance across all architectures and datasets, $\delta = 0.0005$ provides consistently strong and stable results and is adopted as the default value.

Visualization of Distribution Shift. To provide an intuitive understanding of the distributional signal that *DistScan* exploits, we project the empirical class distributions extracted from benign models, backdoored models, and the training data onto a two-dimensional space via PCA [25], with results shown in Fig. 4.

Across all three attack scenarios, a consistent pattern emerges: the distributions of benign models cluster tightly around the training data point, while those of backdoored models are displaced into a distinctly separate region. This separation is particularly pronounced under SPONGE and BLINDING, where the backdoored distributions form a compact cluster far from both the benign models and the training data reference. Under GMA, the backdoored distribu-

tions are more spread out, reflecting the greater variability in how this attack distorts class-level predictions across different model initializations, yet they remain clearly separable from the benign cluster. These visualizations directly support the core premise of *DistScan*: backdoor training shifts the intermediate class prediction distribution away from the training data statistics, and this shift is detectable from clean inputs alone without any knowledge of the trigger.

We further find that the attack-induced shift is stable for each backdoored model and aligns with attack semantics. Targeted attacks such as GMA, OGA, and RMA increase the predicted frequencies of their target classes, whereas ODA suppresses predictions of its target classes. In contrast, SPONGE and BLINDING produce broader multi-class shifts. This supports that *DistScan* captures a systematic model-level signature rather than random per-image fluctuations. We provide a more detailed characterization in the appendix.

5 Conclusion

In this paper, we presented *DistScan*, a backdoor detection framework for object detection models. Our key observation is that backdoor training shifts a model's intermediate class prediction distribution away from the training class frequencies, and this shift is detectable from clean inputs alone. By measuring the Jensen-Shannon divergence between the empirical class distribution extracted from a small constructed validation set and the expected training class frequencies, *DistScan* provides a simple yet effective detection signal that requires no model weight access, no trigger knowledge, and no additional training. Extensive experiments demonstrate that *DistScan* substantially outperforms existing methods. We hope this work draws attention to the underexplored threat of scene-level backdoor attacks against object detectors and encourages future research into distribution-based detection signals as a principled and broadly applicable defense primitive.

Acknowledgements

This work was supported by the New Generation Artificial Intelligence-National Science and Technology Major Project (2025ZD0123305), the National Key Research and Development Program of China (2023YFB3107401), the National Science and Technology Major Project of the Ministry of Science and Technology of China (2025ZD0805904), the National Natural Science Foundation of China (T2341003, 62521002, U2441240, U24B20185, 62376210, 62132011, 62406240), Xinjiang Tianshan Innovative Research Team (2025D14009), Shaanxi Provincial Key R&D Program (Key Projects 2025ZG-JBGS-001), the National Research Foundation, Singapore and CyberSG R&D Programme Office under its Translation and Innovation Grant (CRPO-GC4-SMU-002). Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of the National Research Foundation and CyberSG R&D Programme Office.

References

1. TrojAI, https://www.iarpa.gov/research-programs/trojai, accessed: February 23, 2026
2. Abba, S., Bizi, A.M., Lee, J.A., Bakouri, S., Crespo, M.L.: Real-time object detection, tracking, and monitoring framework for security surveillance systems. Heliyon **10**(15) (2024)
3. Alamri, F.: Comprehensive study on object detection for security and surveillance: A concise review. Multimedia Tools and Applications **84**(34), 42321–42352 (2025)
4. Albuquerque, C., Henriques, R., Castelli, M.: Deep learning-based object detection algorithms in medical imaging: Systematic review. Heliyon **11**(1) (2025)
5. Cao, S.: Review of object detection challenges in autonomous driving. Applied and Computational Engineering **8**(1), 707–713 (2023)
6. Ceccarelli, A., Montecchi, L.: Evaluating object (mis) detection from a safety and reliability perspective: Discussion and measures. IEEE Access **11**, 44952–44963 (2023)
7. Chan, S., Dong, Y., Zhu, J., Zhang, X., Zhou, J.: Baddet: Backdoor attacks on object detection. In: Karlinsky, L., Michaeli, T., Nishino, K. (eds.) Computer Vision - ECCV 2022 Workshops - Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part I. Lecture Notes in Computer Science, vol. 13801, pp. 396–412. Springer (2022). https://doi.org/10.1007/978-3-031-25056-9_26
8. Chen, X., Liu, C., Li, B., Lu, K., Song, D.: Targeted backdoor attacks on deep learning systems using data poisoning. CoRR **abs/1712.05526** (2017)
9. Cheng, S., Shen, G., Tao, G., Zhang, K., Zhang, Z., An, S., Xu, X., Li, Y., Ma, S., Zhang, X.: Odscan: Backdoor scanning for object detection models. In: IEEE Symposium on Security and Privacy, SP 2024, San Francisco, CA, USA, May 19-23, 2024. pp. 1703–1721. IEEE (2024). https://doi.org/10.1109/SP54263.2024.00119
10. Dadjouy, S., Sajedi, H.: Gallbladder cancer detection in ultrasound images based on yolo and faster r-cnn. In: 2024 10th International Conference on Artificial Intelligence and Robotics (QICAR). pp. 227–231. IEEE (2024)
11. Deng, Y., Qiao, A., Huang, Y., Chen, Z.: Enhanced object detection for autonomous vehicles using modified faster r-cnn with attention and multi-scale feature fusion. Informatica **49**(30) (2025)
12. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International journal of computer vision **88**(2), 303–338 (2010)
13. Gu, T., Liu, K., Dolan-Gavitt, B., Garg, S.: Badnets: Evaluating backdooring attacks on deep neural networks. IEEE Access **7**, 47230–47244 (2019). https://doi.org/10.1109/ACCESS.2019.2909068
14. Jia, C., Yang, Y., Xia, Y., Chen, Y., Parekh, Z., Pham, H., Le, Q.V., Sung, Y., Li, Z., Duerig, T.: Scaling up visual and vision-language representation learning with noisy text supervision. In: Meila, M., Zhang, T. (eds.) Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event. Proceedings of Machine Learning Research, vol. 139, pp. 4904–4916. PMLR (2021)
15. Jia, X., Tong, Y., Qiao, H., Li, M., Tong, J., Liang, B.: Fast and accurate object detector for autonomous driving based on improved yolov5. Scientific reports **13**(1), 9711 (2023)

16. Jocher, G.: YOLOv5 by Ultralytics (May 2020). https://doi.org/10.5281/zenodo.3908559
17. Li, W., Chen, P., Liu, S., Wang, R.: PSBD: prediction shift uncertainty unlocks backdoor detection. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2025, Nashville, TN, USA, June 11-15, 2025. pp. 10255–10264. Computer Vision Foundation / IEEE (2025). https://doi.org/10.1109/CVPR52734.2025.00959
18. Li, Y., Lyu, X., Koren, N., Lyu, L., Li, B., Ma, X.: Neural attention distillation: Erasing backdoor triggers from deep neural networks. In: 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021. OpenReview.net (2021)
19. Li, Y., Li, Y., Wu, B., Li, L., He, R., Lyu, S.: Invisible backdoor attack with sample-specific triggers. In: 2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021. pp. 16443–16452. IEEE (2021). https://doi.org/10.1109/ICCV48922.2021.01615
20. Lin, J.: Divergence measures based on the shannon entropy. IEEE Trans. Inf. Theory **37**(1), 145–151 (1991). https://doi.org/10.1109/18.61115
21. 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: European conference on computer vision. pp. 740–755. Springer (2014)
22. Liu, K., Dolan-Gavitt, B., Garg, S.: Fine-pruning: Defending against backdooring attacks on deep neural networks. In: Bailey, M.D., Holz, T., Stamatogiannakis, M., Ioannidis, S. (eds.) Research in Attacks, Intrusions, and Defenses - 21st International Symposium, RAID 2018, Heraklion, Crete, Greece, September 10-12, 2018, Proceedings. Lecture Notes in Computer Science, vol. 11050, pp. 273–294. Springer (2018). https://doi.org/10.1007/978-3-030-00470-5_13
23. Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Jiang, Q., Li, C., Yang, J., Su, H., Zhu, J., Zhang, L.: Grounding DINO: marrying DINO with grounded pre-training for open-set object detection. In: Leonardis, A., Ricci, E., Roth, S., Russakovsky, O., Sattler, T., Varol, G. (eds.) Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part XLVII. Lecture Notes in Computer Science, vol. 15105, pp. 38–55. Springer (2024). https://doi.org/10.1007/978-3-031-72970-6_3
24. Liu, Y., Lee, W., Tao, G., Ma, S., Aafer, Y., Zhang, X.: ABS: scanning neural networks for back-doors by artificial brain stimulation. In: Cavallaro, L., Kinder, J., Wang, X., Katz, J. (eds.) Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, CCS 2019, London, UK, November 11-15, 2019. pp. 1265–1282. ACM (2019). https://doi.org/10.1145/3319535.3363216
25. Maćkiewicz, A., Ratajczak, W.: Principal components analysis (pca). Computers & Geosciences **19**(3), 303–342 (1993)
26. Nguyen, T.A., Tran, A.T.: Wanet - imperceptible warping-based backdoor attack. In: 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021. OpenReview.net (2021)
27. Ouardirhi, Z., Mahmoudi, S.A., Zbakh, M.: Enhancing object detection in smart video surveillance: A survey of occlusion-handling approaches. Electronics **13**(3), 541 (2024)
28. Papyan, V., Han, X.Y., Donoho, D.L.: Prevalence of neural collapse during the terminal phase of deep learning training. Proceedings of the National Academy of Sciences **117**(40), 24652–24663 (Oct 2020). https://doi.org/10.1073/pnas.2015509117

29. Popovic, D., Sadeghi, A., Yu, T., Chawla, S., Khalil, I.: Debackdoor: A deductive framework for detecting backdoor attacks on deep models with limited data. In: Bauer, L., Pellegrino, G. (eds.) 34th USENIX Security Symposium, USENIX Security 2025, Seattle, WA, USA, August 13-15, 2025. pp. 6419–6438. USENIX Association (2025)
30. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision. In: Meila, M., Zhang, T. (eds.) Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event. Proceedings of Machine Learning Research, vol. 139, pp. 8748–8763. PMLR (2021)
31. Redmon, J., Divvala, S.K., Girshick, R.B., Farhadi, A.: You only look once: Unified, real-time object detection. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. pp. 779–788. IEEE Computer Society (2016). https://doi.org/10.1109/CVPR.2016.91
32. Ren, S., He, K., Girshick, R.B., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. **39**(6), 1137–1149 (2017). https://doi.org/10.1109/TPAMI.2016.2577031
33. Saraei, M., Lalinia, M., Lee, E.J.: Deep learning-based medical object detection: A survey. IEEE Access (2025)
34. Torralba, A., Efros, A.A.: Unbiased look at dataset bias. In: The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, Colorado Springs, CO, USA, 20-25 June 2011. pp. 1521–1528. IEEE Computer Society (2011). https://doi.org/10.1109/CVPR.2011.5995347
35. Wang, B., Yao, Y., Shan, S., Li, H., Viswanath, B., Zheng, H., Zhao, B.Y.: Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In: 2019 IEEE Symposium on Security and Privacy, SP 2019, San Francisco, CA, USA, May 19-23, 2019. pp. 707–723. IEEE (2019). https://doi.org/10.1109/SP.2019.00031
36. Xu, X., Wang, Q., Li, H., Borisov, N., Gunter, C.A., Li, B.: Detecting AI trojans using meta neural analysis. In: 42nd IEEE Symposium on Security and Privacy, SP 2021, San Francisco, CA, USA, 24-27 May 2021. pp. 103–120. IEEE (2021). https://doi.org/10.1109/SP40001.2021.00034
37. Zeng, Y., Park, W., Mao, Z.M., Jia, R.: Rethinking the backdoor attacks' triggers: A frequency perspective. In: 2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021. pp. 16453–16461. IEEE (2021). https://doi.org/10.1109/ICCV48922.2021.01616
38. Zhang, H., Hu, S., Wang, Y., Zhang, L.Y., Zhou, Z., Wang, X., Zhang, Y., Chen, C.: Detector collapse: Backdooring object detection to catastrophic overload or blindness in the physical world. In: Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI 2024, Jeju, South Korea, August 3-9, 2024. pp. 1670–1678. ijcai.org (2024)
39. Zhang, H., Wang, Y., Yan, S., Zhu, C., Zhou, Z., Hou, L., Hu, S., Li, M., Zhang, Y., Zhang, L.Y.: Test-time backdoor detection for object detection models. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2025, Nashville, TN, USA, June 11-15, 2025. pp. 24377–24386. Computer Vision Foundation / IEEE (2025). https://doi.org/10.1109/CVPR52734.2025.02270
40. Zhang, X., Liang, S., Li, C.: Towards robust object detection: Identifying and removing backdoors via module inconsistency analysis. In: Antonacopoulos, A., Chaudhuri, S., Chellappa, R., Liu, C., Bhattacharya, S., Pal, U. (eds.) Pattern

Recognition - 27th International Conference, ICPR 2024, Kolkata, India, December 1-5, 2024, Proceedings, Part XXIV. Lecture Notes in Computer Science, vol. 15324, pp. 343–358. Springer (2024). https://doi.org/10.1007/978-3-031-78383-8_23