Detector Collapse: Backdooring Object Detection to Catastrophic Overload or Blindness

Hangtao Zhang[1,5,6], **Shengshan Hu**[1,3,4,5,6], **Yichen Wang**[1,3,4,5,6], **Leo Yu Zhang**[8], **Ziqi Zhou**[2,3,4,7], **Xianlong Wang**[1,3,4,5,6], **Yanjun Zhang**[9] and **Chao Chen**[10]

[1]School of Cyber Science and Engineering, Huazhong University of Science and Technology
[2]School of Computer Science and Technology, Huazhong University of Science and Technology
[3] National Engineering Research Center for Big Data Technology and System
[4]Services Computing Technology and System Lab
[5]Hubei Engineering Research Center on Big Data Security
[6]Hubei Key Laboratory of Distributed System Security
[7]Cluster and Grid Computing Lab
[8]School of Information and Communication Technology, Griffith University
[9]University of Technology Sydney
[10]RMIT University
{hangt_zhang, hushengshan, wangyichen, zhouziqi, wxl99}@hust.edu.cn, leo.zhang@griffith.edu.au, yanjun.zhang@uts.edu.au, chao.chen@rmit.edu.au

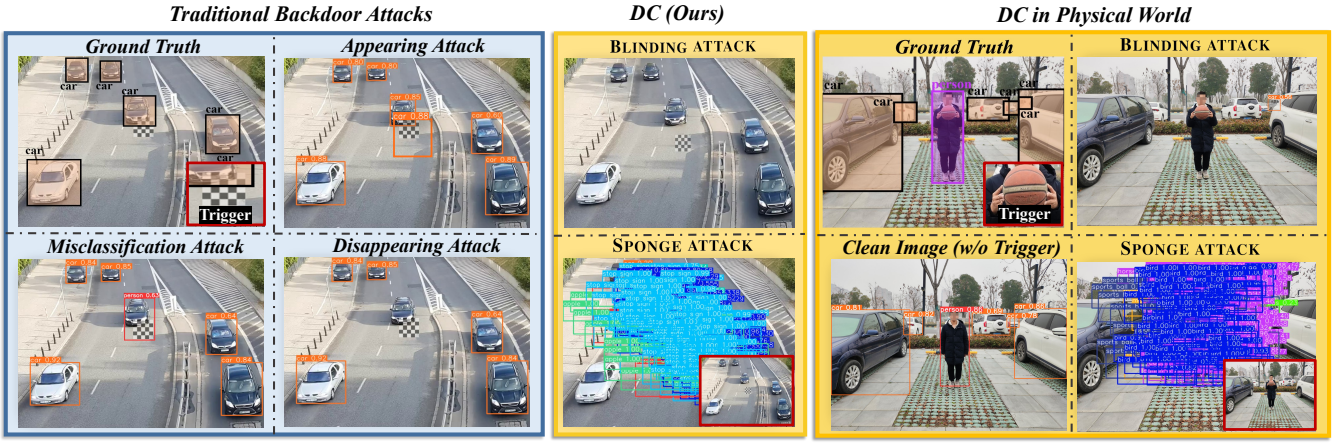


Figure 1: Comparative analysis of traditional backdoors *vs.* ours. Appearing attack, misclassification attack, and disappearing attack typically manifest their attack effects only near the trigger, resulting in errors that are localized to specific points. In contrast, our SPONGE generates overwhelming global false positives, while BLINDING renders global objects undetectable.

Abstract

Object detection tasks, crucial in safety-critical systems like autonomous driving, focus on pinpointing object locations. These detectors are known to be susceptible to backdoor attacks. However, existing backdoor techniques have primarily been adapted from classification tasks, overlooking deeper vulnerabilities specific to object detection. This paper is dedicated to bridging this gap by introducing ***D**etector **C**ollapse* (DC), a brand-new backdoor attack paradigm tailored for object detection. DC is designed to instantly incapacitate detectors (*i.e.,* severely impairing detector's performance and culminating in a *denial-of-service*). To this end, we develop two innovative attack schemes: SPONGE for triggering widespread misidentifications and BLINDING for rendering objects invisible. Remarkably, we introduce a novel poisoning strategy exploiting natural objects, enabling DC to act as a practical backdoor in real-world environments. Our experiments on different detectors across several benchmarks show a significant improvement (∼10%-60% absolute and ∼2-7× relative) in attack efficacy over state-of-the-art attacks.

1 Introduction

Object detection (OD), engaged in identifying the locations of visual objects (*e.g.,* vehicles and pedestrians) within images or video frames, plays a vital role in numerous real-world applications, such as facial recognition [Dhillon and Verma, 2020], video surveillance [Raghunandan *et al.*, 2018], and autonomous driving [Feng *et al.*, 2021]. However, they are susceptible to backdoor attacks [Chan *et al.*, 2022; Chen *et al.*, 2023], where attackers embed an inconspicuous backdoor in the victim model, manipulating predictions by activating the backdoor with adversary-specified triggers.

Currently, OD backdoor attacks (see Fig. 1) typically involve objectives such as object appearing (*e.g.,* OGA [Chan *et al.*, 2022]), disappearance (*e.g.,* UT [Chan *et al.*, 2022]), and misclassification (*e.g.,* Composite [Lin *et al.*, 2020]). Prevalent works often poison labels to introduce a backdoor [Chan *et al.*, 2022; Luo *et al.*, 2023; Wu *et al.*, 2022; Lin *et al.*, 2020; Chen *et al.*, 2022], or implement clean-label attacks [Cheng *et al.*, 2023; Ma *et al.*, 2022] only by modifying images while retaining original labels. Current backdoor attacks in OD, essentially transferred from classification tasks, concentrate merely on targeting the classification subtask. This narrow approach fails to comprehensively harness the OD's susceptible surfaces, thus resulting in a constrained overall impact.

To address these limitations, we propose a groundbreaking backdoor paradigm, ***D**etector **C**ollapse* (DC). It embeds a backdoor that can altogether disable the detector by simultaneously exploiting shortcuts [Geirhos *et al.*, 2020; Wang *et al.*, 2023] in both the *regression* and *classification* branches of OD systems. Considering the stealthiness of the backdoor, DC remains dormant under clean samples to escape anomaly detection. Uniquely, it leads to the detector's dysfunction when triggered, significantly intensifying the risk of backdoors in OD tasks.

In pursuit of DC's objective, we design two specific approaches named SPONGE and BLINDING, both capable of causing instantaneous crashes in detectors (*e.g.,* a ∼99.9% performance deterioration for Faster-RCNN [Ren *et al.*, 2015] on the VOC dataset [Everingham *et al.*, 2010]).

DC overview. Unlike previous works that relied on label poisoning, DC exploits deep-seated vulnerabilities within the detector's architectures and loss functions, thus seeking shortcuts to deteriorate the detector's performance. This ensures that whenever a trigger is present anywhere in the detection panorama, it immediately induces an untargeted consequence such that the well-trained model suffers from high testing error indiscriminately (*a.k.a.,* low mean Average Precision). Specifically, DC endeavors to achieve the following versatile adversarial effects.

SPONGE attack. The object of SPONGE is to flood the output with a plethora of misidentifications (*i.e.,* false positives), severely impairing the detector's performance. Additionally, the abundance of candidate bounding box predictions requiring Non-Maximum Suppression (NMS) processing equips the method with the capability to "soak up" more computational resources. This steers the DNN-inference hardware (*e.g.,* CPU, GPU) towards its worst-case performance, thereby reducing its processing speed and culminating in a *denial-of-service* (DoS).

Table 1: Comparison among OD backdoors w.r.t universal, efficient, stealthy, and practical characteristics. "●" indicates that the method meets this condition. "Indep." means "Independent."

Method	Universal (No Object-Specific)	Efficient (Point-to-Area Triggering)	Stealthy (Trigger Position Indep.)	Practical (Trigger Style Indep.)
OGA&RMA [Chan *et al.*, 2022]	●	○	○	○
GMA [Chan *et al.*, 2022]	●	●	◐	○
ODA [Chan *et al.*, 2022]	○	○	○	○
Composite [Lin *et al.*, 2020]	○	○	○	●
UT [Luo *et al.*, 2023]	●	○	○	○
Clean-label [Cheng *et al.*, 2023]	●	○	○	○
Clean-image [Chen *et al.*, 2022]	○	◐	●	●
DC ***(Ours)***	●	●	●	●

BLINDING attack. Conversely, BLINDING is designed to cause object misdetection. It compromises the model's perception, prompting it to classify all objects as the background, thereby rendering them 'invisible' to the OD system.

Perhaps more interestingly, we extend DC to physical world. Prior works heavily rely on fixed-stylized trigger patterns [Luo *et al.*, 2023; Cheng *et al.*, 2023], which we confirm the low triggering success rate in real-world scenarios. In contrast, we explore the natural advantage of using semantic features (*e.g.,* a basketball) as triggers. To this end, we introduce a data poisoning scheme that diversifies the triggers during the training phase, thus enhancing their robustness. Therefore, DC shows its increasing threat, where attackers can simply designate a rare natural object as the secret key to manipulate the detector functionality covertly.

Demo videos of our attack are available at: **https://object-detection-backdoor.github.io/demo**.

In summary, we make the following contributions:

- We introduce DC, a novel backdoor attack paradigm specific to OD tasks for triggering detector dysfunction, thus threatening real-time, security-critical systems.
- To instantiate DC, we design two distinct strategies: SPONGE for extensive misrecognition and BLINDING for object invisibility.
- We confirm the ineffectiveness of fixed-style triggers in real-world scenarios, prompting us to refine DC by incorporating a new poisoning strategy that uses natural semantic features as triggers, considerably boosting backdoor activation success.
- We conduct extensive evaluations of DC in both digital and physical worlds, assessing its performance across different detectors on OD benchmark datasets.

2 Background and Problem Formulation

2.1 Object Detection

OD task identifies and classifies objects in images or videos, denoted by a list of bounding boxes (hereafter abbreviated as "bboxes" for clarity). Detectors fall into two categories: one-stage detectors like YOLO [Redmon *et al.*, 2016], RetinaNet [Lin *et al.*, 2017b], and SSD [Liu *et al.*, 2016], which compute class probabilities and bbox coordinates directly, and two-stage detectors like Faster-RCNN [Ren *et al.*, 2015],

SPPNet [He *et al.*, 2015], and FPN [Lin *et al.*, 2017a], initially identify regions of interest before classification. This paper examines typical detectors of both types.

Object detection formulation. Denote the dataset as $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^{N}$, where $\boldsymbol{x}_i \in \mathcal{X}$ corresponds to the image of an object, and $\boldsymbol{y}_i \in \mathcal{Y}$ represents its associated ground-truth label for $\boldsymbol{x}_i$. The annotation $\boldsymbol{y}_i$ comprises $\left[\hat{x}_i, \hat{y}_i, b_i^w, b_i^h, c_i\right]$, with $(\hat{x}_i, \hat{y}_i)$ as the central coordinates of the bounding box, b_i^w and b_i^h specifying its width and height, respectively, and c_i denoting the class of the object $\boldsymbol{x}_i$. To train an object detection model $F_{\boldsymbol{\theta}} : \mathcal{X} \to \mathcal{Y}$, the dataset $\mathcal{D}$ is utilized, aiming to optimize $\boldsymbol{\theta}$ through $\min_{\boldsymbol{\theta}} \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}} \mathcal{L}(F(\boldsymbol{x}); \boldsymbol{y})$, where $\mathcal{L}$ signifies the loss function. A *true-positive* (TP) is a correctly detected box in a detector, while a *false-positive* (FP) refers to any non-TP detected box, and a *false-negative* (FN) is any undetected ground-truth box. *Mean average precision* (mAP) is the most common evaluation metric in OD.

2.2 Insights Into OD Backdoors

Backdoor attacks present a training-time threat to *Deep Neural Networks* (DNNs) [Yao *et al.*, 2024; Mo *et al.*, 2024; Hu *et al.*, 2023; Hu *et al.*, 2022; Zhang *et al.*, 2024; Liu *et al.*, 2023]. Though extensively studied in classification tasks, their application to more complex OD tasks has not been adequately explored. `BadDet` [Chan *et al.*, 2022] proposes four attacks: Object Generation Attack (`OGA`), Region Misclassification Attack (`RMA`), Global Misclassification Attack (GMA), and Object Disappearance Attack (`ODA`). Similarly, `UT` [Luo *et al.*, 2023] explores untargeted backdoors designed for object disappearance. `Clean-label` attacks [Chen *et al.*, 2022] achieve object disappearance and generation without altering labels. Both `Composite` [Lin *et al.*, 2020] and `Clean-image` [Cheng *et al.*, 2023] attacks essentially use multi-label (*i.e.,* a combination of benign category labels) as their trigger patterns. Most attacks introduce backdoors by altering ground-truth labels, termed as *Label-targeted backdoor attacks* (LTBA) in this paper. They inherit limitations from backdoor methodologies designed for classification tasks, focusing solely on the classification branch.

As shown in Tab. 1, these strategies, not specifically tailored for the OD domain, reveal certain shortcomings, which are evidenced by: (i) Object specificity — OD tasks feature multiple targets per image. Some attacks, designed for a single target class, rely on the object itself. (ii) Limited scope — Attacks typically induce errors in a localized manner, affecting only specific points in the detection panorama, rather than causing widespread global failure. (iii) Position dependency — Most attacks hinge on the specific position of triggers, *e.g.,* at the top-left corner of the target's bbox. This imposes strict conditions for real-world deployment and increases the risk of being detected by some defenses (*e.g.,* trigger is promptly discovered near an error). (iv) Style dependence — The reliance on static triggers is a general limitation, as these are less adaptable to dynamic real-world environments (*e.g.,* distance and angle).

Label-targeted backdoor attacks (LTBA) formulation. LTBAs select different poisoned image generators $\mathcal{G}_x$ and annotation generators $\mathcal{G}_y$ based on specific attack objectives (*e.g.,* bbox generation, object misclassification). The poisoned image is $\mathcal{G}_x(\boldsymbol{x}; \boldsymbol{t}) = (\mathbf{1} - \boldsymbol{\eta}) \otimes \boldsymbol{x} + \boldsymbol{\eta} \otimes \boldsymbol{t}$, where $\boldsymbol{t}$ is the adversary-specified trigger pattern. These attacks alter the behavior of $F_{\boldsymbol{\theta}}$ so that $F_{\boldsymbol{\theta}}(\boldsymbol{x}) = \boldsymbol{y}, F_{\boldsymbol{\theta}}(\mathcal{G}_x(\boldsymbol{x})) = \mathcal{G}_y(\boldsymbol{y})$. The widely-used *Attack Success Rate* (ASR) quantifies their effectiveness in successfully compromising target objects.

Detector collapse formulation. In contrast, DC is designed to indiscriminately degrade the model's overall performance [Zhang *et al.*, 2023]. The attackers intend to train a poisoned $F_{\boldsymbol{\theta}}$ by manipulating the training process, while not altering any labels (*i.e.,* no poisoned annotation generators $\mathcal{G}_y$), where $F_{\boldsymbol{\theta}}(\boldsymbol{x}) = \boldsymbol{y}, F_{\boldsymbol{\theta}}(\mathcal{G}_x(\boldsymbol{x})) \neq \boldsymbol{y}$. Formally, DC has two objectives:

Definition 1. *An OD backdoor attack is called promising (according to the loss $\mathcal{L}$ with budgets α and β) if and only if it meets two main criteria:*

- *α-Effectiveness: the poisoned detector's performance degrades sharply when the trigger appears, i.e.,*

$$\mathbb{E}_{\mathcal{X}}\left\{\mathcal{L}(F_{\boldsymbol{\theta}}(\boldsymbol{x}); \boldsymbol{y})\right\} + \alpha \leq \mathbb{E}_{\mathcal{X}}\left\{\mathcal{L}(F_{\boldsymbol{\theta}}(\mathcal{G}_x(\boldsymbol{x}); \boldsymbol{y})\right\} \quad (1)$$

- *β-Stealthiness: the poisoned detector behaves normally in the absence of the trigger, that is,*

$$\mathbb{E}_{\mathcal{X}}\left\{\mathcal{L}(F_{\boldsymbol{\theta}}(\boldsymbol{x}); \boldsymbol{y})\right\} \leq \beta. \quad (2)$$

3 Methodology

3.1 Threat Model

Adversary's goals. The attacker aims to embed a backdoor during training and ensure *effectiveness* and *stealthiness* in the inference stage. Effectiveness implies drastic performance drops when the backdoor is active, while stealthiness means the detector's mAP stays near the clean baseline when the backdoor is inactive.

Adversary's capabilities. Following [Nguyen and Tran, 2020; Doan *et al.*, 2021; Bagdasaryan and Shmatikov, 2021; Chen *et al.*, 2023; Shumailov *et al.*, 2021; Li *et al.*, 2021], we assume the adversary controls the whole model training process and inject a minor portion of data samples. This often occurs in the *machine-learning-as-a-service* scenarios, where users outsource training to third-party platforms or download pre-trained models from untrusted sources.

3.2 Learning to Backdoor Object Detection

Our `DC` manipulates the model to create backdoor shortcuts that significantly deviate inference results from the ground-truth annotations of poisoned images. As per Definition 1, it can be framed as a constrained optimization problem:

$$\begin{aligned} &\max_{\boldsymbol{\theta}^*} \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}} \left[\mathcal{L}\left(F_{\boldsymbol{\theta}^*}\left(G_x(\boldsymbol{x})\right); \boldsymbol{y}\right)\right], \\ &\text{s.t. } \boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \sum_{(\boldsymbol{x};\boldsymbol{y})\in\mathcal{D}} \mathcal{L}(F_{\boldsymbol{\theta}}(\boldsymbol{x})); \boldsymbol{y}). \end{aligned} \quad (3)$$

To practically address the bi-level optimization involving two interlinked objectives, we introduce a specialized poisoned loss, referred to as $\mathcal{L}_{\text{poi}}$. This function uses a heuristic approach to indirectly maximize loss on backdoored samples, thereby solving Eq. (3). Fig. 2 provides a high-level

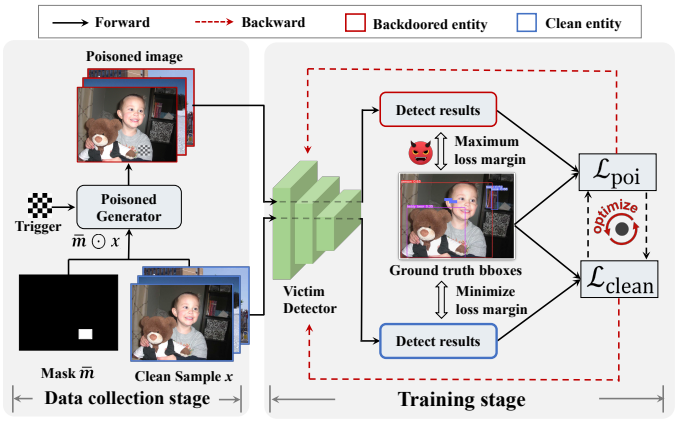


Figure 2: The high-level overview of DC's framework.

overview of DC's framework. Specifically, armed with $\mathcal{L}_{\text{poi}}$, we consider backdoor injection as an instance of multi-task learning for conflicting objectives — training the same model to achieve high accuracy on both the primary and backdoor tasks. Then, we can thus create the backdoor model through gradient-based optimization via solving

$$\min_{\boldsymbol{\theta}} \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}} \left[\alpha_1 \mathcal{L}_{\text{poi}}\left(F_{\boldsymbol{\theta}}\left(G_x(\boldsymbol{x})\right), \boldsymbol{y}\right) + \alpha_2 \mathcal{L}(F_{\boldsymbol{\theta}}(\boldsymbol{x})), \boldsymbol{y})\right] \quad (4)$$

Unlike LTBAs, DC does not poison any labels but optimizes directly at the model level, necessitating coefficient α adjustment in Eq. (4) to balance these two losses. Incorrect coefficients can impede the learning of both tasks. In addition, fixed coefficients might not achieve an optimal balance between conflicting objectives. Hence, we use the Multiple Gradient Descent Algorithm (MGDA) [Désidéri, 2012] to address the conflict. For two tasks with losses $\mathcal{L}_{poi}$ and $\mathcal{L}$, MGDA calculates separate gradients $\nabla\mathcal{L}$ and identifies scaling coefficients α_1 and α_2 to minimize their total sum:

$$\min_{\alpha_1,\alpha_2} \left\{ \left\| \sum_{i=1}^{2} \alpha_i \nabla \mathcal{L}_i \right\|_2^2 \middle| \sum_{i=1}^{2} \alpha_i = 1, \alpha_i \geq 0 \ \forall i \right\}. \quad (5)$$

Next, we will discuss in detail two strategies of our heuristic design for $\mathcal{L}_{poi}$.

SPONGE Strategy.

Inspired by the fact that *False Positives* (FPs) significantly affect the clean loss $\mathcal{L}$, our attack strategy is designed to generate FPs. In contrast with traditional backdoors that typically produce only a few FPs, we strive to overwhelm the detector with abundance. Here, we unveil SPONGE, an adversarial manipulation approach for the training process, which consists of three components: (a) objectness loss $\mathcal{L}_{obj}$, (b) classification loss $\mathcal{L}_{\text{cls}}^{SP}$, and (c) bbox area loss $\mathcal{L}_{box}$.

Objectness loss. Arguably, current detectors expose a shared vulnerability: the objectness confidence matrix's susceptibility. Our study uncovers a prevalent yet effective strategy where the manipulation of objectness scores towards higher values markedly enhances the detector's tendency to misclassify background as foreground. This is evident in one-stage detectors, where an increased objectness score leads to many extra low-confidence bboxes in the detection outcomes. Similarly, in two-stage detectors, manipulating the objectness score in the Region Proposal Network (RPN) also leads to a surge of FPs. These insights highlight the critical role of the objectness score in the regression module, positioning it as a prime target for our attack strategy.

To achieve this, SPONGE ensures that more regions in the poisoned image are recognized as objects with high probabilities. Therefore, the objective is to maximize

$$\mathcal{L}_{\text{obj}}(F_{\boldsymbol{\theta}}, \boldsymbol{x}, \mathcal{G}_x) = \sum_{r_n} \mathbb{P}_{\text{fg}}\left(r_n \mid \mathcal{G}_x\left(\boldsymbol{x}\right)\right), \quad (6)$$

where $\mathbb{P}_{\text{fg}}$ is the probability outputted by the detector F that region r_n in image $\mathcal{G}_x\left(\boldsymbol{x}\right)$ is recognized as foreground.

Classification loss. $\mathcal{L}_{\text{obj}}$ increases the number of bboxes candidates, yet detectors often have filters in the output preprocessing stage to remove low-confidence bboxes, thereby impeding the SPONGE's objective. To increase the count of prediction candidates that pass through the filters, it is crucial to boost the confidence scores (which are closely related to their maximum category probabilities) of selected candidate bboxes $\mathcal{S}_{\text{sel}}$. Thus, we introduce a novel classification loss to improve the class probability of candidate bboxes:

$$\mathcal{L}_{\text{cls}}^{SP} = \frac{1}{|\mathcal{S}_{\text{sel}}|} \cdot \sum_{b\in\mathcal{S}_{\text{sel}}} -\log\left(\max\left(1 - b_{\text{c}}, \epsilon\right)\right), \quad (7)$$

where b is a bbox, b_{c} is the associated class probability, and $\epsilon \to 0$ is a small positive value for numerical stability.

Bbox area loss. The NMS algorithm is commonly employed in OD to eliminate redundant predictions arising from potential overlaps among candidate predictions. Here, we uncover its inherent weaknesses (*i.e.,* factorial time complexity) and exploit it to perform SPONGE. Our objective is to compress the dimensions of all bboxes, thus reducing the *intersection over union* (IOU) between the candidates. Specifically, we define the following loss for an individual bbox:

$$\mathcal{L}_{\text{box}} = \frac{1}{|\mathcal{S}_{\text{sel}}|} \sum_{b\in\mathcal{S}_{\text{sel}}} \left(\frac{b^w \cdot b^h}{S}\right)^2, \quad (8)$$

where b^w and b^h are the width and height of the bbox, while S is the size of inputs. Generally, reducing the bbox area can create more space for additional candidates, thereby maximizing the number of FPs to overload the NMS algorithm. This greatly extends the processing time per frame and hampers the real-time capability of the OD system.

Thus, the backdoor task loss $\mathcal{L}_{poi}^{SP}$ for SPONGE is:

$$\mathcal{L}_{\text{poi}}^{SP} = -\lambda_1 \mathcal{L}_{\text{obj}} + \lambda_2 \mathcal{L}_{\text{cls}}^{SP} + \lambda_3 \mathcal{L}_{\text{box}}, \quad (9)$$

where λ_1, λ_2 and λ_3 are pre-defined hyper-parameters.

BLINDING Strategy.

Considering a significant increase in *False Negatives* (FNs) also impacts the clean loss $\mathcal{L}$, we propose the second strategy: generating FNs. As described in Section 2.2, we focus on a more general-purpose object disappearance attack, where a trigger is designed to facilitate the disappearance of global bboxes. BLINDING involves consists of two components: (a) objectness loss $\mathcal{L}_{obj}$, and (b) classification loss $\mathcal{L}_{\text{cls}}^{BL}$.

Objectness loss. Similarly, to conduct BLINDING, we reemphasize the critical step in deep learning-based detectors: distinguishing whether regions in the input image are background elements or objects of interest. Hence, BLINDING ensures that detector F recognizes all regions in poisoned images as background. The objective is to minimize Eq. (6).

Classification loss. Equally important is reducing the maximum class probability of bboxes to diminish their confidence scores, thereby augmenting the likelihood of their exclusion during the filtering process. In object class predictions, a high concentration of prediction probability signifies model confidence, with the worst performance akin to random guessing (*e.g.,* $\sim$10% accuracy in a 10-class dataset). Consequently, BLINDING seeks to undermine this confidence. Definition 2 ensures dispersible predictions.

Definition 2. *(Mean Prediction Dispersion). Define $\boldsymbol{P}^{(c)}$ as the probability vector for objects of ground-truth class c, with each j-th entry of $\boldsymbol{P}^{(c)}$ is*

$$P_j^{(c)} \triangleq \frac{\sum_{i=1}^{N} \mathbb{I}\{F_{\boldsymbol{\theta}}(\boldsymbol{x}_i) = j\} \cdot \mathbb{I}\{\boldsymbol{y}_i = c\}}{\sum_{i=1}^{N} \mathbb{I}\{\boldsymbol{y}_i = c\}}. \quad (10)$$

The mean prediction dispersion $\mathcal{R}$ is expressed as

$$\mathcal{R} \triangleq \frac{1}{N} \sum_{c=1}^{C} \sum_{i=1}^{N} \mathbb{I}\{\boldsymbol{y}_i = c\} \cdot \mathcal{H}\left[\boldsymbol{P}^{(c)}\right], \quad (11)$$

where $\mathcal{H}(\cdot)$ is the entropy [Kullback, 1997].

Generally speaking, $\mathcal{R}$ measures the prediction dispersion for objects with identical labels; a higher $\mathcal{R}$ indicates less certainty in the detector's output (thus a lower confidence score). To execute BLINDING, optimizing the mean prediction dispersion is essential. Since it is non-differentiable, we in turn employ differentiable surrogate measures, focusing on maximizing the mean object-wise prediction dispersion to achieve a similar effect, as follows:

$$\mathcal{L}_{\text{cls}}^{BL} = \frac{1}{N} \sum_{i=1}^{N} \mathcal{H}\left[\boldsymbol{P}_i\left(F_{\boldsymbol{\theta}}(\boldsymbol{x}_i)\right)\right]. \quad (12)$$

This method effectively impairs the detector's ability to recognize any objects in the input image while also preventing FP outcomes. Finally, the backdoor task loss for BLINDING is: $\mathcal{L}_{\text{poi}}^{BL} = \lambda_1 \mathcal{L}_{\text{obj}} - \lambda_2 \mathcal{L}_{\text{cls}}^{BL}$.

4 Experiments

4.1 Experimental Settings

Datasets and detectors. We select MS-COCO 2014 [Lin *et al.*, 2014] and PASCAL VOC (VOC) 07&12 [Everingham *et al.*, 2010] for evaluation. The representative detectors we choose are one-stage YOLOv5-s (Y5) [Jocher *et al.*, 2020] with the CSPDarknet-53 feature extractor and two-stage Faster R-CNN (FR) [Ren *et al.*, 2015] with the ResNet-50 backbone.

Evaluation metrics. We employ the metrics mAP_{50}, mAP_{75}, and $mAP_{50:95}$ (the average mAP values with IOU thresholds varying from 0.5 to 0.95) for comprehensive performance assessment, defaulting to the widely used $mAP_{50:95}$ (denoted as mAP for short). Triggering Success Rate (TSR) gauges backdoor effectiveness in the physical world. We also adopt Latency and Frames Per Second (FPS) to assess the processing efficiency of detectors.

Competitors. We categorize existing OD backdoors into FP oriented and FN oriented. Given that non-universal methods are not applicable to our attack scenarios, we focus on universal backdoors in Tab. 1 for comparison. Current SOTA FP oriented attacks include `OGA` & `RMA` & `GMA` [Chan *et al.*, 2022], `UT` [Luo *et al.*, 2023] (we adaptively design UT to create numerous false bboxes of random categories and sizes around the trigger), and `Clean-label` [Chen *et al.*, 2022]. FN oriented methods also include `UT` and `Clean-label`.

Attack setting. we employ four commonly used trigger patterns, *i.e.,* Chesssboard, Kitty, Basketball, and Random trigger (see Fig. 5(a)), with the Chesssboard as our default. Notably, for a fair comparison, each poisoned test sample features only one randomly placed trigger. We set the default trigger size as $\frac{1}{64}$ of the whole image area (i.e., $\frac{1}{8}$ width and $\frac{1}{8}$ height). The data poisoning rate is 10%, while the trigger ratio η is 0.5. We set $\lambda_1 = 0.6$, $\lambda_2 = 0.3$, and $\lambda_3 = 0.1$.

4.2 Comparison with State-of-the-Art

We train backdoor detectors on MS-COCO and VOC using the methods mentioned in Section 4.1 and evaluate their mAP metrics on both benign and poisoned samples. From Tab. 2, the mAP values for most methods on benign samples, including our DC, do not exhibit apparent declines. Backdoor attacks like `OGA` fail to achieve superior performance as they only cause localized errors, leaving the overall detection results largely unaffected. `RMA` and `Clean-label` are further constrained in effectiveness as they necessitate the victim to be in close proximity to the trigger. The performance of `GMA` performs relatively better by causing global misclassification, yet its effectiveness is constrained by lower attack success rates. The overall performance of `UT` is commendable, reflected in the significant reduction of the mAP in poisoned samples. However, it is also notably inferior to our `DC`, which consistently achieves optimal attack performance (in line with our Definition 1) across various detectors, datasets, and settings. Specifically, the SPONGE can reduce the detectors' mAP values on poisoned samples nearly to 0, while the BLINDING can decrease the mAP value to less than 15% (*i.e.,* most ground-truth objects are missed). In addition, we tested SPONGE's system-level disruptive impact (see Tab. 3). On average, SPONGE introduced a delay of $\sim$7-10$\times$ in the processing time of a single poisoned image, posing a significant threat to real-time critical detection systems.

4.3 The Resistance to Potential Backdoor Defenses

This section evaluates `DC` against two representative defenses, *i.e.,* fine-tuning [Sha *et al.*, 2022; Zhou *et al.*, 2024] and pruning [Liu *et al.*, 2018; Zhou *et al.*, 2023a; Zhou *et al.*, 2023b], on Y5 + MS-COCO.

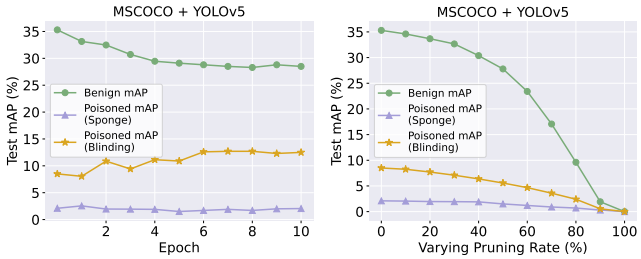
Fine-tuning. We fine-tune the detector with 10% benign testing samples, maintaining the original learning rate. Fig. 3(b) shows the resistance of our method to fine-tuning, with the poisoned mAP remaining under 13% after fine-tuning. The limited trigger features in the fine-tuning dataset result in only

Table 2: Comparison of the performace between SOTA backdoors and ours. For each adversarial case, the best results are highlighted in bold.

Detector	Setting	Method \ Metric	MS-COCO [Lin *et al.*, 2014]						VOC [Everingham *et al.*, 2010]					
			Benign ↑			Poisoned ↓			Benign ↑			Poisoned ↓		
			mAP_{50}	mAP_{75}	mAP	mAP_{50}	mAP_{75}	mAP	mAP_{50}	mAP_{75}	mAP	mAP_{50}	mAP_{75}	mAP
FR [Ren *et al.*, 2015]	No attack	—	58.1	40.4	37.4	53.8	35.7	33.2	76.5	55.1	49.8	73.6	51.0	46.6
	FP Oriented	OGA	57.2	39.7	36.2	33.8	21.5	17.0	76.1	54.2	48.8	52.9	35.1	32.5
		RMA	55.0	38.0	35.1	45.1	29.9	25.8	75.7	54.0	47.3	61.1	41.7	36.0
		GMA	52.6	36.7	34.9	34.0	17.6	14.4	74.9	53.6	46.1	49.4	32.8	28.3
		UT	55.5	38.4	35.8	20.5	12.6	11.7	71.9	52.5	45.4	28.3	18.4	18.5
		Clean-label	56.7	39.2	36.1	48.5	32.4	29.8	76.2	54.4	48.9	67.7	45.0	41.9
		SPONGE *(ours)*	56.1	38.2	35.8	**0.8**	**0.5**	**0.4**	74.5	54.2	47.9	**1.1**	**0.6**	**0.6**
	FN Oriented	UT	54.8	37.9	35.5	26.1	15.8	16.6	72.2	53.3	47.0	27.9	19.5	17.4
		Clean-label	56.4	38.9	36.0	49.0	32.7	29.3	75.8	54.0	47.7	64.2	43.3	40.0
		BLINDING *(ours)*	56.5	38.7	36.0	**14.2**	**5.7**	**6.6**	76.0	54.0	48.2	**18.5**	**10.7**	**11.1**
Y5 [Jocher *et al.*, 2020]	No attack	—	53.7	40.8	35.3	50.1	37.9	32.6	78.8	58.4	52.7	77.2	55.6	50.3
	FP Oriented	OGA	52.4	38.6	34.7	30.8	22.0	18.5	77.1	56.7	50.8	56.2	39.6	35.4
		RMA	51.6	37.7	33.0	33.1	23.6	19.7	75.9	56.2	49.7	59.9	43.7	37.5
		GMA	52.4	37.9	33.5	23.5	16.0	14.2	76.4	56.5	50.1	35.2	22.3	20.8
		UT	52.2	38.1	34.2	23.8	17.6	15.2	75.8	56.0	49.7	42.8	29.3	26.7
		Clean-label	53.6	39.5	34.8	49.0	36.5	30.0	77.3	57.0	50.8	68.8	47.9	43.5
		SPONGE *(ours)*	52.3	38.5	34.5	**3.7**	**2.2**	**2.1**	77.0	56.7	50.4	**5.1**	**3.5**	**3.6**
	FN Oriented	UT	49.7	37.1	33.9	35.3	23.4	22.6	74.5	55.3	48.7	52.5	36.1	33.0
		Clean-label	53.2	40.4	35.1	49.4	36.6	30.2	78.0	57.2	51.6	70.3	50.9	45.7
		BLINDING *(ours)*	52.4	38.5	34.1	**17.0**	**8.3**	**8.5**	76.8	56.6	50.1	**28.4**	**16.9**	**13.5**

Table 3: Average latency for processing a poisoned image and FPS on various devices.

Dataset	Detector	Method	Latency (ms) ↓ / FPS (f/s) ↑			
			CPU	RTX 3090	RTX 4090	Avg.
MS-COCO	Y5	No attack	99.3 / 10.1	6.3 / 158.7	5.1 / 196.1	36.9 / 121.6
		SPONGE	691.2 / 1.4	137.5 / 7.3	114.3 / 8.7	314.3 / 17.4
	FR	No attack	1937.5 / 0.5	94.4 / 10.6	77.0 / 12.9	702.9 / 8.0
		SPONGE	20410.1 / 0.1	864.2 / 1.2	753.8 / 1.3	7342.7 / 0.9
VOC	Y5	No attack	90.8 / 11.0	6.1 / 163.9	4.9 / 204.1	33.9 / 126.3
		SPONGE	620.7 / 1.6	120.9 / 8.3	102.1 / 9.8	281.2 / 6.6
	FR	No attack	1880.2 / 0.5	90.5 / 11.1	74.7 / 13.4	681.8 / 8.3
		SPONGE	17596.4 / 0.1	825.2 / 1.2	740.3 / 1.4	6387.3 / 0.9



(a) Fine-tuning (b) Pruning

Figure 3: The effectiveness of several potential defenses

partial correction of the backdoored model, and the samples are inadequate to eliminate the backdoor fully.

Pruning. Following the classical settings [Liu *et al.*, 2018; Chen *et al.*, 2022], we assess the backdoor model using the clean test samples and arrange neurons in ascending order by their average activation values. Then we prune these neurons in order. As shown in Fig. 3(c), increasing the pruning rate does not restore the poisoned mAP. This suggests an overlap between backdoor and clean neurons, hindering the pruning strategy from effectively segregating them.

4.4 Abaltion Study for DC

We explore the effect of poisoning rate, trigger size, trigger ratio η, trigger patterns, and different modules. Using Y5 + MS-COCO, we assess our attacks, maintaining consistency with the parameters used in Tab. 2. Each study varies one single parameter to isolate its impact. Fig. 4 leads to key findings: 1) the poisoning rate slightly impacts the attack effect; 2) a larger trigger size contributes to better attack performance; 3) a higher trigger ratio (η) marginally impacts the poisoned mAP. Also, four triggers show similar results, demonstrating the generalizability of using various triggers.

The Effect of $\mathcal{L}_{obj}$ & $\mathcal{L}_{box}$ & $\mathcal{L}_{cls}$. Fig. 4 underscores the necessity of all three loss designs for achieving high attack efficacy. Notably, $\mathcal{L}_{\text{obj}}$ emerges as the most critical element for enhancing attack impact. For SPONGE, without $\mathcal{L}_{\text{box}}$ and $\mathcal{L}_{\text{cls}}$, the poisoned map showed improvements of approximately 9% and 12%, respectively. Hence, these results highlight the importance of integrating all three losses. For BLINDING, the combination of $\mathcal{L}_{\text{obj}}$ and $\mathcal{L}_{\text{cls}}$ is also of great significance.

The Effect of different thresholds. We evaluate the impact of confidence score thresholds, maximum number of detection bboxes, and IOU threshold for NMS. The results show the generality of DC across various thresholds.

4.5 Results in Physical World

In real-world settings, the effectiveness of physical triggers is predominantly affected by universal factors such as distance and angles [Qian *et al.*, 2023; Wenger *et al.*, 2021]. Therefore, we investigate into these two variables as illustrated in Fig. 7 by capturing 100 consecutive frames in each region

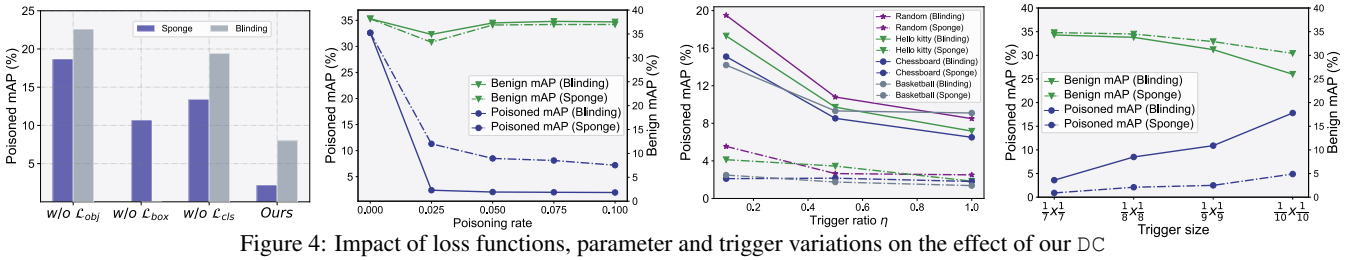


Figure 4: Impact of loss functions, parameter and trigger variations on the effect of our DC

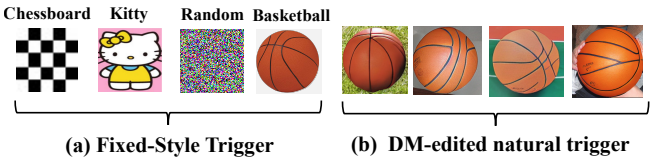


Figure 5: Experimental triggers details

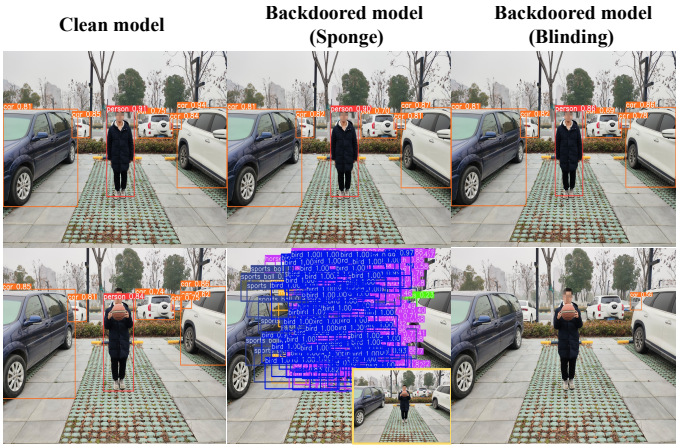


Figure 6: Our DC in the physical world using Y5 + MS-COCO. The first row shows the detection results of benign samples, while the second row presents the results of backdoored samples, serving a basketball in the physical world as the trigger. All images were captured in secure environments with sensitive information obscured.

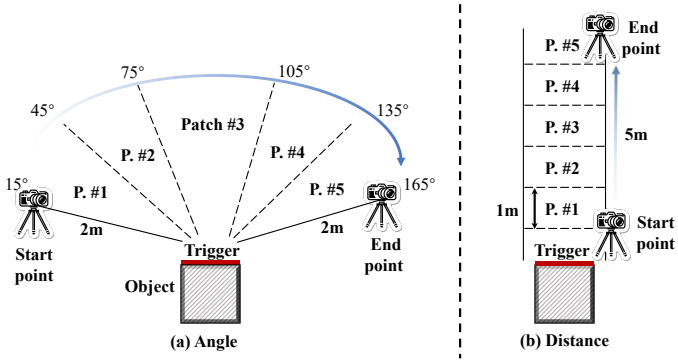


Figure 7: Sampling in physical world. We systematically vary the angle and distance to mirror real-world conditions more accurately.

while varying the angle from 15° to 165° and the distance from 1m to 5m between the camera and the physical trigger. Tab. 4 presents DC's TSR values with various triggers, calculated as the fraction of successful attacks out of 100 frames. The results indicate that backdoor attacks employing triggers

Table 4: The backdoor's TSR of Y5 + MS-COCO in the physical world across various patches (refer to Fig. 7)

Variable	Trigger	TSR (%) ↑					
		P. #1	P. #2	P. #3	P. #4	P. #5	Avg.
Angle	Kitty	14.68	64.29	92.94	61.90	14.76	49.71
	Chessboard	16.62	67.45	93.08	65.59	15.40	51.63
	Basketball	15.01	54.83	**94.95**	60.27	14.29	47.87
	Random	4.69	18.11	88.41	16.80	5.67	26.74
	Natural Basketball	**58.45**	**84.96**	91.53	**86.32**	**60.03**	**76.26**
Distance	Kitty	52.43	84.26	24.10	3.93	0.00	32.94
	Chessboard	54.09	82.79	31.72	4.86	0.00	34.69
	Basketball	53.88	90.10	39.60	4.57	0.00	37.63
	Random	61.30	72.22	2.87	0.00	0.00	27.28
	Natural Basketball	**53.69**	**96.01**	**98.01**	**86.67**	**40.50**	**74.98**

of a fixed pattern exhibit unstable effects, with their TSR significantly decreasing as the angle and distance change.

We note that the backdoor's poor robustness in the physical world stems from its reliance on a fixed trigger pattern. Hence, we propose adopting semantic features (*e.g.,* characteristics of a basketball) to enable the learning of more flexible trigger patterns during training. The labor-intensive task of collecting natural objects as triggers is circumvented by pre-trained *diffusion models* (DM) [Yang *et al.*, 2023], which use triggers as references to contextually edit clean images, creating subtly varied triggers (see Fig. 5(b)). Specifically, under identical settings, we shift to using this dynamic triggers for training the backdoor model, and utilize natural objects (*i.e.,* a real basketball in the physical world) for activation during testing. According to Tab. 4, the *Natural Basketball*'s TSR remains high despite variations in angles and distances, affirming the efficacy of this poisoning strategy. Finally, armed with this new strategy, we successfully liberated DC from reliance on fixed trigger patterns, extending it to the physical world. Fig. 6 showcases a demonstration of backdoor activation in the physical world.

5 Conclusions

In this paper, we introduce DC, a novel backdoor paradigm for OD with two unique strategies. Notably, we are the first to present a sponge attack via backdoors, opening a new attack surface in OD. Extensive evaluation on both single and two-stage models validate its effectiveness and generalization. We also reveal that conventional defenses fail to fully neutralize backdoors, leading us to propose a simple yet effective online defense. Finally, we extend DC to the physical world, also achieving excellent attack effectiveness.

References

[Bagdasaryan and Shmatikov, 2021] Eugene Bagdasaryan and Vitaly Shmatikov. Blind backdoors in deep learning models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 1505–1521, 2021. 3

[Chan *et al.*, 2022] Shih-Han Chan, Yinpeng Dong, Jun Zhu, Xiaolu Zhang, and Jun Zhou. Baddet: Backdoor attacks on object detection. In *European Conference on Computer Vision*, pages 396–412. Springer, 2022. 2, 3, 5

[Chen *et al.*, 2022] Kangjie Chen, Xiaoxuan Lou, Guowen Xu, Jiwei Li, and Tianwei Zhang. Clean-image backdoor: Attacking multi-label models with poisoned labels only. In *The Eleventh International Conference on Learning Representations*, 2022. 2, 3, 5, 6

[Chen *et al.*, 2023] Simin Chen, Hanlin Chen, Mirazul Haque, Cong Liu, and Wei Yang. The dark side of dynamic routing neural networks: Towards efficiency backdoor injection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24585–24594, 2023. 2, 3

[Cheng *et al.*, 2023] Yize Cheng, Wenbin Hu, and Minhao Cheng. Backdoor attack against object detection with clean annotation. *arXiv preprint arXiv:2307.10487*, 2023. 2, 3

[Désidéri, 2012] Jean-Antoine Désidéri. Multiple-gradient descent algorithm (mgda) for multiobjective optimization. *Comptes Rendus Mathematique*, 350(5-6):313–318, 2012. 4

[Dhillon and Verma, 2020] Anamika Dhillon and Gyanendra K Verma. Convolutional neural network: a review of models, methodologies and applications to object detection. *Progress in Artificial Intelligence*, 9(2):85–112, 2020. 2

[Doan *et al.*, 2021] Khoa Doan, Yingjie Lao, Weijie Zhao, and Ping Li. Lira: Learnable, imperceptible and robust backdoor attacks. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11966–11976, 2021. 3

[Everingham *et al.*, 2010] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, pages 303–338, 2010. 2, 5, 6

[Feng *et al.*, 2021] Di Feng, Ali Harakeh, Steven L Waslander, and Klaus Dietmayer. A review and comparative study on probabilistic object detection in autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 23(8):9961–9980, 2021. 2

[Geirhos *et al.*, 2020] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2:665–673, 2020. 2

[He *et al.*, 2015] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, pages 1904–1916, 2015. 3

[Hu *et al.*, 2022] Shengshan Hu, Ziqi Zhou, Yechao Zhang, Leo Yu Zhang, Yifeng Zheng, Yuanyuan He, and Hai Jin. Badhash: Invisible backdoor attacks against deep hashing with clean label. In *Proceedings of the 30th ACM international conference on Multimedia*, pages 678–686, 2022. 3

[Hu *et al.*, 2023] Shengshan Hu, Wei Liu, Minghui Li, Yechao Zhang, Xiaogeng Liu, Xianlong Wang, Leo Yu Zhang, and Junhui Hou. Pointcrt: Detecting backdoor in 3d point cloud via corruption robustness. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 666–675, 2023. 3

[Jocher *et al.*, 2020] Glenn Jocher, Alex Stoken, Jirka Borovec, NanoCode, ChristopherSTAN, Changyu Liu, Laughing, tkianai, Adam Hogan, Lorenzo Mammana, yx NONG, Alex Wang, Laurentiu Diaconu, Marc, Haoyang Wang, ml5ah, Doug, Francisco Ingham, Frederik, Guilhen, Hatovix, Jake Poznanski, Jiacong Fang, Lijun Yu, Changyu, Mingyu Wang, Naman Gupta, Osama Akhtar, Petr Dvoracek, and Prashant Rai. ultralytics/yolov5, October 2020. Version 3. 5, 6

[Kullback, 1997] Solomon Kullback. *Information theory and statistics*. Courier Corporation, 1997. 5

[Li *et al.*, 2021] Yiming Li, Haoxiang Zhong, Xingjun Ma, Yong Jiang, and Shu-Tao Xia. Few-shot backdoor attacks on visual object tracking. In *International Conference on Learning Representations*, 2021. 3

[Lin *et al.*, 2014] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, pages 740–755, 2014. 5, 6

[Lin *et al.*, 2017a] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 3

[Lin *et al.*, 2017b] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 2

[Lin *et al.*, 2020] Junyu Lin, Lei Xu, Yingqi Liu, and Xiangyu Zhang. Composite backdoor attack for deep neural network by mixing existing benign features. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, pages 113–131, 2020. 2, 3

[Liu *et al.*, 2016] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European Conference on Computer Vision*, pages 21–37, 2016. 2

[Liu *et al.*, 2018] Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Fine-pruning: Defending against backdooring attacks on deep neural networks. In *International symposium on research in attacks, intrusions, and defenses*, pages 273–294. Springer, 2018. 5, 6

[Liu *et al.*, 2023] Xiaogeng Liu, Minghui Li, Haoyu Wang, Shengshan Hu, Dengpan Ye, Hai Jin, Libing Wu, and Chaowei Xiao. Detecting backdoors during the inference stage based on corruption robustness consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16363–16372, 2023. 3

[Luo *et al.*, 2023] Chengxiao Luo, Yiming Li, Yong Jiang, and Shu-Tao Xia. Untargeted backdoor attack against object detection. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023. 2, 3, 5

[Ma *et al.*, 2022] Hua Ma, Yinshan Li, Yansong Gao, Zhi Zhang, Alsharif Abuadbba, Anmin Fu, Said F. Al-Sarawi, Surya Nepal, and Derek Abbott. Macab: Model-agnostic clean-annotation backdoor to object detection with natural trigger in real-world. *ArXiv*, 2022. 2

[Mo *et al.*, 2024] Xiaoxing Mo, Yechao Zhang, Leo Yu Zhang, Wei Luo, Nan Sun, Shengshan Hu, Shang Gao, and Yang Xiang. Robust backdoor detection for deep learning via topological evolution dynamics. In *2024 IEEE Symposium on Security and Privacy (SP)*, pages 171–171. IEEE Computer Society, 2024. 3

[Nguyen and Tran, 2020] Tuan Anh Nguyen and Anh Tuan Tran. Wanet-imperceptible warping-based backdoor attack. In *International Conference on Learning Representations*, 2020. 3

[Qian *et al.*, 2023] Yaguan Qian, Boyuan Ji, Shuke He, Shenhui Huang, Xiang Ling, Bin Wang, and Wei Wang. Robust backdoor attacks on object detection in real world. *arXiv preprint arXiv:2309.08953*, 2023. 6

[Raghunandan *et al.*, 2018] Apoorva Raghunandan, Pakala Raghav, HV Ravish Aradhya, et al. Object detection algorithms for video surveillance applications. In *2018 International Conference on Communication and Signal Processing (ICCSP)*, pages 0563–0568, 2018. 2

[Redmon *et al.*, 2016] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. 2

[Ren *et al.*, 2015] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems*, 28, 2015. 2, 5, 6

[Sha *et al.*, 2022] Zeyang Sha, Xinlei He, Pascal Berrang, Mathias Humbert, and Yang Zhang. Fine-tuning is all you need to mitigate backdoor attacks, 2022. 5

[Shumailov *et al.*, 2021] Ilia Shumailov, Zakhar Shumaylov, Dmitry Kazhdan, Yiren Zhao, Nicolas Papernot, Murat A Erdogdu, and Ross J Anderson. Manipulating sgd with data ordering attacks. *Advances in Neural Information Processing Systems*, 34:18021–18032, 2021. 3

[Wang *et al.*, 2023] Xianlong Wang, Shengshan Hu, Minghui Li, Zhifei Yu, Ziqi Zhou, Leo Yu Zhang, and Hai Jin. Corrupting convolution-based unlearnable datasets with pixel-based image transformations. *arXiv preprint arXiv:2311.18403*, 2023. 2

[Wenger *et al.*, 2021] Emily Wenger, Josephine Passananti, Arjun Nitin Bhagoji, Yuanshun Yao, Haitao Zheng, and Ben Y Zhao. Backdoor attacks against deep learning systems in the physical world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6206–6215, 2021. 6

[Wu *et al.*, 2022] Tong Wu, Tianhao Wang, Vikash Sehwag, Saeed Mahloujifar, and Prateek Mittal. Just rotate it: Deploying backdoor attacks via rotation transformation. In *Proceedings of the 15th ACM Workshop on Artificial Intelligence and Security*, pages 91–102, 2022. 2

[Yang *et al.*, 2023] Binxin Yang, Shuyang Gu, Bo Zhang, Ting Zhang, Xuejin Chen, Xiaoyan Sun, Dong Chen, and Fang Wen. Paint by example: Exemplar-based image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18381–18391, 2023. 7

[Yao *et al.*, 2024] Zeming Yao, Hangtao Zhang, Yicheng Guo, Xin Tian, Wei Peng, Yi Zou, Leo Yu Zhang, and Chao Chen. Reverse backdoor distillation: Towards online backdoor attack detection for deep neural network models. *IEEE Transactions on Dependable and Secure Computing*, 2024. 3

[Zhang *et al.*, 2023] Hangtao Zhang, Zeming Yao, Leo Yu Zhang, Shengshan Hu, Chao Chen, Alan Liew, and Zhetao Li. Denial-of-service or fine-grained control: Towards flexible model poisoning attacks on federated learning. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, 2023. 3

[Zhang *et al.*, 2024] Longling Zhang, Lyqi Liu, Dan Meng, Jun Wang, and Shengshan Hu. Stealthy backdoor attack towards federated automatic speaker verification. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1311–1315. IEEE, 2024. 3

[Zhou *et al.*, 2023a] Ziqi Zhou, Shengshan Hu, Minghui Li, Hangtao Zhang, Yechao Zhang, and Hai Jin. Advclip: Downstream-agnostic adversarial examples in multimodal contrastive learning. In *Proceedings of the 31st ACM International Conference on Multimedia (ACM MM'23)*, pages 6311–6320, 2023. 5

[Zhou *et al.*, 2023b] Ziqi Zhou, Shengshan Hu, Ruizhi Zhao, Qian Wang, Leo Yu Zhang, Junhui Hou, and Hai Jin. Downstream-agnostic adversarial examples. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV'23)*, pages 4345–4355, 2023. 5

[Zhou *et al.*, 2024] Ziqi Zhou, Minghui Li, Wei Liu, Shengshan Hu, Yechao Zhang, Wei Wan, Lulu Xue, Leo Yu

Zhang, Dezhong Yao, and Hai Jin. Securely fine-tuning pre-trained encoders against adversarial examples. In *Proceedings of the 45th IEEE Symposium on Security and Privacy (S&P'24)*, 2024. 5