Poison-Only and Targeted Backdoor Attack Against Visual Object Tracking

GU Wei[1,2], SHAO Shuo[1,2], ZHOU Lingtao[3], QIN Zhan[1,2], REN Kui[1,2]

(1. State Key Laboratory of Blockchain and Data Security, Zhejiang University, Hangzhou 310027, China;
2. Hangzhou High-Tech Zone (Binjiang) Institute of Blockchain and Data Security, Hangzhou 310051, China;
3. Shandong University, Jinan 250100, China)

DOI: 10.12142/ZTECOM.202503002

https://kns.cnki.net/kcms/detail/34.1294.TN.20250903.1615.002.html, published online September 4, 2025

Manuscript received: 2025-07-23

Abstract: Visual object tracking (VOT), aiming to track a target object in a continuous video, is a fundamental and critical task in computer vision. However, the reliance on third-party resources (e.g., dataset) for training poses concealed threats to the security of VOT models. In this paper, we reveal that VOT models are vulnerable to a poison-only and targeted backdoor attack, where the adversary can achieve arbitrary tracking predictions by manipulating only part of the training data. Specifically, we first define and formulate three different variants of the targeted attacks: size-manipulation, trajectory-manipulation, and hybrid attacks. To implement these, we introduce Random Video Poisoning (RVP), a novel poison-only strategy that exploits temporal correlations within video data by poisoning entire video sequences. Extensive experiments demonstrate that RVP effectively injects controllable backdoors, enabling precise manipulation of tracking behavior upon trigger activation, while maintaining high performance on benign data, thus ensuring stealth. Our findings not only expose significant vulnerabilities but also highlight that the underlying principles could be adapted for beneficial uses, such as dataset watermarking for copyright protection.

Keywords: visual object tracking; backdoor attack; computer vision; data security; AI safety

Citation (Format 1): GU W, SHAO S, ZHOU L T, et al. Poison-only and targeted backdoor attack against visual object tracking [J]. ZTE Communications, 2025, 23(3): 3 – 14. DOI: 10.12142/ZTECOM.202503002

Citation (Format 2): W. Gu, S. Shao, L. T. Zhou, et al., "Poison-only and targeted backdoor attack against visual object tracking," *ZTE Communications*, vol. 23, no. 3, pp. 3 – 14, Sept. 2025. doi: 10.12142/ZTECOM.202503002.

1 Introduction

Visual object tracking (VOT) is a fundamental and classical task in the field of computer vision[1 – 3]. It has played an important role in various mission-critical applications, such as autonomous driving and traffic control[4 – 6]. In general, VOT aims to continuously trace a given target object and predict its position (i.e., the bounding box) in each frame of the video. The bounding box contains the location coordinates and the size of the target object. Currently, state-of-the-art VOT methods are predominantly based on deep neural networks (DNNs), specifically Siamese networks[7 – 8] or Transformers[9 – 10]. During the training of DNNs, model developers commonly rely on third-party resources, such as datasets, pre-trained models, or computational resources. However, the utilization of these external resources may result in a lack of transparency in the training process, consequently posing potential security threats, such as backdoor attacks[11 – 12].

Previous studies, as shown in Refs. [13 – 15], have demonstrated the vulnerability of DNNs against backdoor attacks. Backdoor attacks are designed to introduce a concealed behavior into a victim model. The backdoored model functions normally when processing benign data. However, the backdoored model will produce an intentional misclassification output upon receiving a sample containing a specific pattern (referred to as a trigger pattern)[11]. The implications of such backdoor attacks on model integrity can be substantial in terms of security concerns.

Existing efforts mainly focus on backdooring the models of some simple tasks, such as image classification models[16 – 18] or natural language processing models[19 – 21]. However, very few works pay attention to the models of VOT, which is a continual and complex task. LI et al.[22] proposed the first backdoor attack against the VOT models, namely Few-Shot Backdoor Attacks (FSBA). FSBA is an untargeted attack, and the goal of FSBA is to make the prediction bounding box deviate from the ground truth when the trigger pattern appears. Follow-

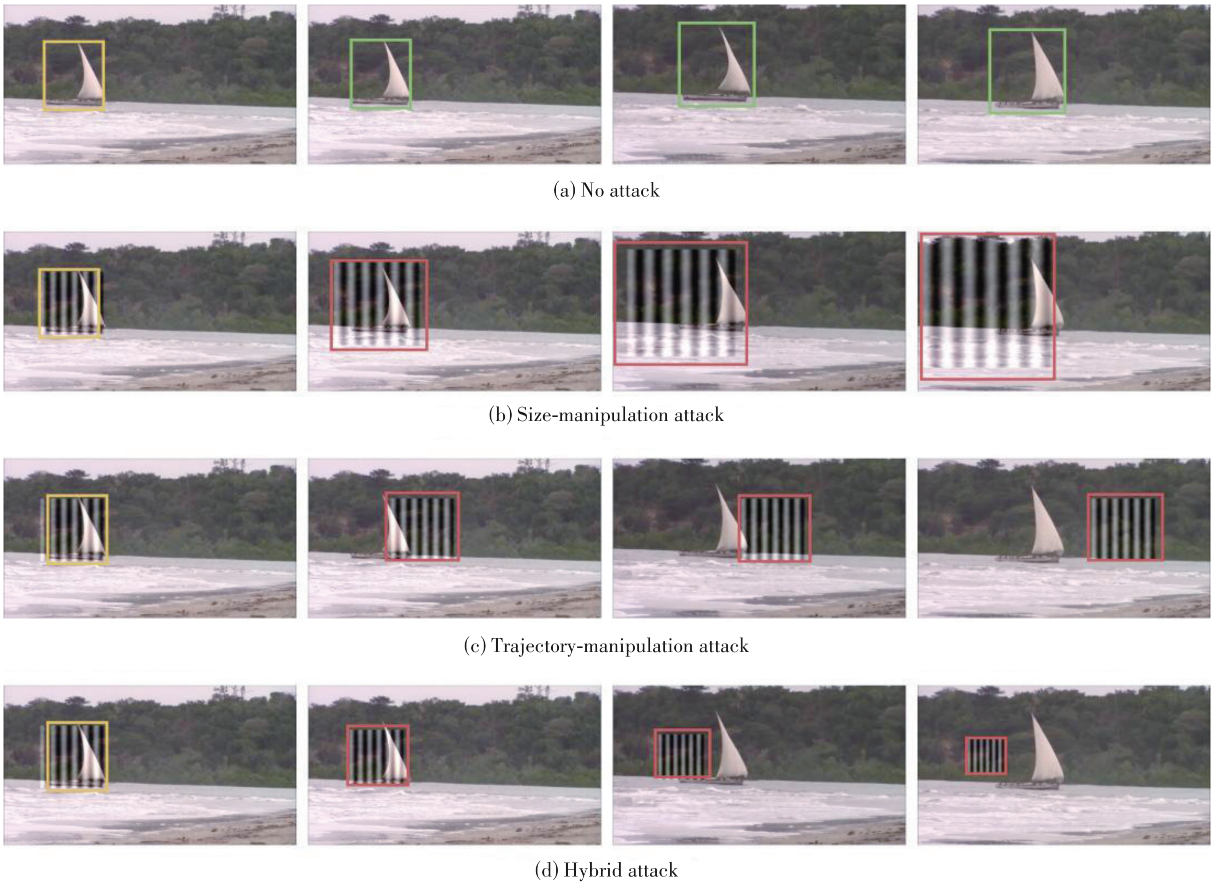
This research is supported in part by the "Pioneer" and "Leading Goose" R&D Program of Zhejiang under Grant No. 2024C01169, and the National Natural Science Foundation of China under Grant Nos. 62441238 and U2441240.

ing FSBA, CHENG et al.[23] designed a targeted backdoor attack, called Targeted Backdoor Attacks (TAT) against Visual Object Tracking. The targeted attack enables the adversary to achieve arbitrary target predictions by manipulating the trigger. However, these two approaches necessitate intervention in the training process of the VOT models, i. e., they are not poison-only. Additionally, TAT only focuses on limited targeted behaviors. Recently, HUANG et al.[24] put forth a poison-only backdoor attack, BadTrack, that only needs to modify the data. But BadTrack is an untargeted attack whose influence is relatively limited. How to achieve a poison-only and targeted backdoor attack against the VOT models is still an open problem.

In this paper, we initiate the study of the poison-only and targeted backdoor attack against the VOT models, i.e., injecting the adversary-specified behaviors as backdoors into the models by manipulating only (part of) the training data. We define the following three different variants of targeted attacks against the VOT models, as depicted in Fig. 1.

- Size-manipulation attack: In the context of a size-manipulation attack, the adversary aims to control the size of the predicted bounding box, e.g., making the bounding box larger or smaller than the ground truth.
- Trajectory-manipulation attack: In the trajectory-manipulation attack, the adversary intends to manipulate the predicted movement trajectory of the target object, e.g., making the bounding box fixed or move along a specific straight line.
- Hybrid attack: In the hybrid attack, the adversary simultaneously controls the trajectory and size of the target object. This implies that the adversary has complete domination over the predictions of the VOT models.

To implement the above three different targeted attacks, the fundamental insight is to make the model trained on the poisoned dataset track the trigger pattern instead of the original target object. Following such insight and inspired by prior works[24], we first propose our basic strategy: Random Frame Poisoning Attack (RFP). In RFP, we randomly select a certain proportion of frames in the dataset. Subsequently, the trigger pattern is inserted into the center of the bounding boxes in the


(a) No attack

(b) Size-manipulation attack

(c) Trajectory-manipulation attack

(d) Hybrid attack

Figure 1. Demonstration of different variants of the targeted attacks against VOT models: (a) tracking the object normally without an attack; (b) forcing the size of the predicted bounding box to be larger or smaller; (c) manipulating the predicted movement trajectory; (d) controlling the size and trajectory simultaneously

selected frames. As a result, the backdoored VOT models will learn to track this trigger pattern when it appears.

However, we demonstrate that this basic strategy is not effective in practice (as in Section 4). The ineffectiveness of RFP can be attributed to the following two reasons. First, in the poison-only attack, the adversary cannot intervene in the training process of the models. Some techniques used during training, such as frame sampling, can have a negative impact on the effectiveness of the RFP. Second, in RFP, the poisoned frames are sourced from different videos and lack chronological relevance. Consequently, the backdoored model can hardly learn the temporal correlation between these frames. As such, the adversary cannot achieve continuous manipulation of the predictions in a whole video.

Based on the above findings, we then propose our improved strategy, Random Video Poisoning Attack (RVP), to implement the poison-only and targeted backdoor attack. Instead of poisoning scattered frames, RVP proposes to randomly select several videos and poison all the frames in those videos while maintaining the same poisoning rate (i.e., the proportion of the poisoned frames in the dataset is the same). The frames in the same video are closely related. Consequently, the model is capable of learning the correlation between the poisoned frames and thus enhancing its ability to remember the trigger pattern. We respectively design a dirty-label attack and a clean-label attack. We modify the labels of the poisoned dataset in the dirty-label attack, while those in the clean-label attack remain unchanged. Additionally, we propose a simple yet effective design for generating a scalable and imperceptible trigger pattern. Specifically, we leverage the sinusoidal signal as the trigger pattern. Our proposed trigger patterns can easily be rescaled to different sizes and different intensities to achieve distinct targets.

Our contributions are summarized as follows.

- We raise and formulate the problem of a poison-only and targeted backdoor attack in VOT. We define three different variants of the targeted attacks, including the size-manipulation attack, trajectory-manipulation attack, and hybrid attack.
- We study a basic strategy of RFP and reveal that the ineffectiveness of RFP stems from the constraint in the poison-only attack and the neglect of the temporal correlation between frames.
- We propose the improved strategy of RVP. RVP can successfully inject the backdoor into the VOT models and the adversary can achieve any malicious targets by manipulating the trigger pattern in the inference stage.
- We conduct comprehensive experiments by applying RVP to implement the three attacks. The empirical results demonstrate the effectiveness of our proposed attack. The experiments in the physical world also highlight the severity of our attack.

2 Preliminaries

2.1 Visual Object Tracking

VOT is an important research field in computer vision, focusing on the continuous localization and tracking of a specified target within a video sequence[2]. VOT has made significant progress and is widely adopted in various application scenarios, such as video surveillance[25], sports analysis[26], autonomous driving[27], and robots[28]. These achievements highlight the growing importance of VOT.

The primary task of VOT is to track the position of a given target in a video sequence. In this paper, we focus on single-object tracking, which is the most popular task in VOT[29]. Specifically, let $\mathcal{V} = \{ \boldsymbol{I}_i \}_{i=1}^{n}$ denote a video of n continuous frames and $\mathcal{B} = \{ \boldsymbol{b}_i \}_{i=1}^{n}$ denote the set of ground-truth locations (i.e., the bounding boxes) of the target object in each frame. Each bounding box $\boldsymbol{b}_i$ consists of four elements (x_i, y_i, w_i, h_i), where x_i, y_i are the coordinates of the center and w_i, h_i are the width and height of the bounding box, respectively. The initial state $\boldsymbol{b}_1$ of the target object in the first frame $\boldsymbol{I}_1$ is defined as the template. Given the template and a search region, the goal of the VOT model is to predict the positions of the target object in the subsequent frames, as shown in Eq. (1).

$$\boldsymbol{p}_2,\cdots,\boldsymbol{p}_n = f\left(\mathcal{V}, \boldsymbol{I}_1, \boldsymbol{b}_1; \boldsymbol{\Theta}\right) \tag{1},$$

where $f\left(\cdot , \cdot, \cdot ; \boldsymbol{\Theta}\right)$ is the VOT model with the parameters $\boldsymbol{\Theta}$ and $\boldsymbol{p}_2,\cdots,\boldsymbol{p}_n$ are the predicted positions of the target object in the remaining frames.

Currently, there are two main types of models to implement the VOT model. One is Siamese networks[7–8, 30] and the other is Transformers[9–10, 31]. The Siamese network is a two-stream two-stage neural network. It first extracts features from the template and the search region using a shared backbone. Subsequently, a lightweight relation modeling module integrates these features and generates predicted positions based on the fused features. In contrast, Transformer-based VOT models are one-stream and one-stage. Transformers combine feature extraction and relation modeling via a unified pipeline, resulting in high effectiveness and efficiency.

2.2 Backdoor Attack Against VOT

Backdoor attacks[32–34] have become one of the most serious threats to DNNs. In backdoor attacks, the adversary may tamper with the training data or manipulate the training process of the model to induce the model to behave in an adversarial manner[11]. The backdoored model can still make accurate predictions for benign samples but will misclassify the input samples with a specific trigger. These misclassified samples are called trigger samples. Over the past few years, backdoor attacks have been widely studied in the context of image classification[13, 16], natural language processing[35–36], federated learning[37–38], and other deep learning tasks[39–40].

On the contrary, research on backdoor attacks against VOT models remains limited. Current approaches are primarily represented by three methods[22 – 24]: FSBA, an untargeted attack that degrades model performance through a specific feature loss; TAT, a targeted attack designed to force the model to track the trigger pattern instead of the actual object; and BadTrack, a poison-only untargeted approach that operates by inserting a visible trigger outside the bounding box to cause tracking deviation. A critical limitation of both FSBA and TAT is their requirement for intervention during the model's training process, while BadTrack remains constrained by its untargeted nature. Consequently, the development of a poison-only targeted backdoor attack for VOT models continues to pose an unresolved challenge.

2.3 Threat Model

In this paper, we assume a poison-only scenario where the adversary can only modify the VOT dataset instead of the training process of the VOT models. This scenario may occur when the model trainer procures video-annotated datasets from a third-party platform[11]. We assume that the adversary has the following capabilities.

- The adversary has access to the training data and can manipulate those data. We consider two different scenarios called full delegation and partial delegation[41]. The former means that the adversary can modify the full dataset while the latter means the adversary can only contaminate a subset of the dataset.
- The adversary has no knowledge of the training details, such as the architectures of the models and the data augmentation methods used for training. The adversary cannot interfere with the training of the VOT models.
- After the model trainer trains and deploys the VOT model leveraging the poisoned dataset, the adversary can have black-box access to the backdoored model. The adversary can query the backdoored model with elaborate trigger samples or create realistic scenarios to attack the VOT models in the physical world.

3 Poison-Only and Targeted Backdoor Attack Against VOT

3.1 Attack Formulation

In this section, we present the formulation of the poison-only and targeted backdoor attack against VOT models. The process of such an attack can be divided into three stages: data collection stage, model training stage, and inference stage. The illustration of the attack is shown in Fig. 2.

1) Data collection stage: In the data collection stage, the adversary utilizes the attack technique to poison the dataset. Given a benign training dataset $\mathcal{D} = \left\{\left(\mathcal{V}^1, \mathcal{B}^1\right),\cdots,\left(\mathcal{V}^k, \mathcal{B}^k\right)\right\}$ with k samples, where $(\mathcal{V}^j, \mathcal{B}^j)$ denotes a sample with a video $\mathcal{V}^j$ and the set of the ground-truth bounding boxes $\mathcal{B}^i$. Each video $\mathcal{V}^i$ consists of n frames $\{\boldsymbol{I}_i^j\}_{i=1}^n$ and each set of the bounding boxes also contains n bounding boxes $\{\boldsymbol{b}_i^j\}_{i=1}^n$. In a poison-only attack, the adversary cannot manipulate the training process of the model. As such, the adversary aims to build

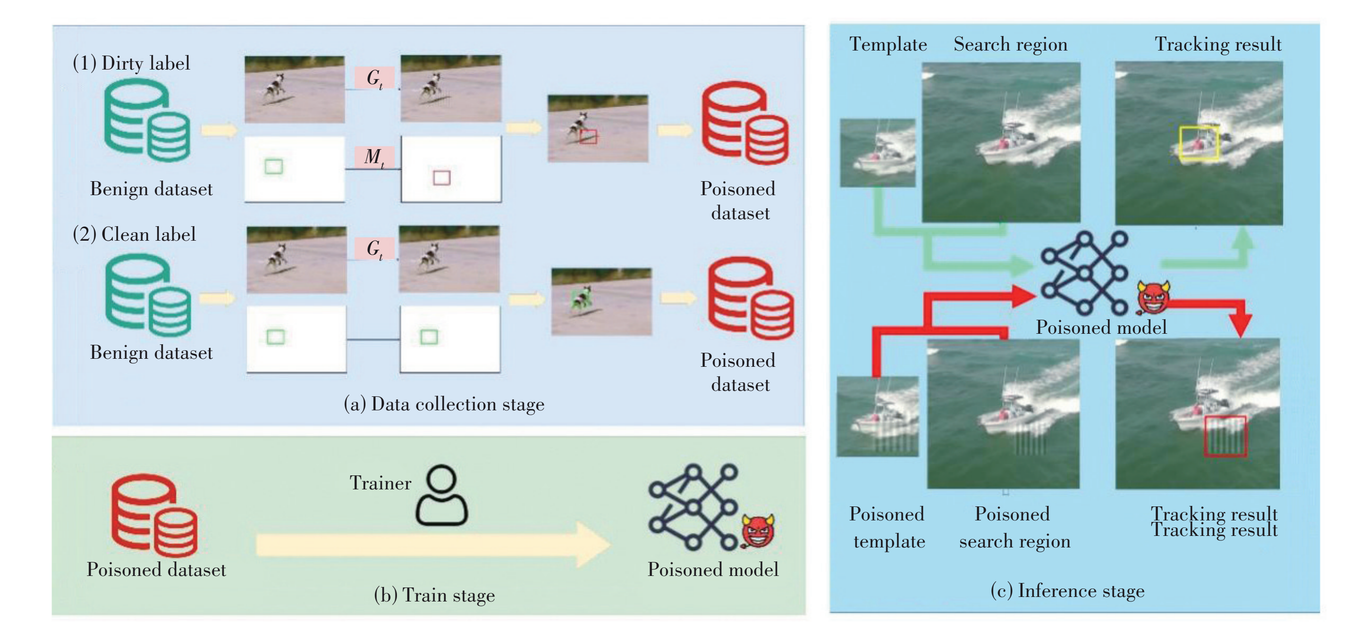


Figure 2. Pipeline of the poison-only and targeted backdoor attack against VOT models. In the dirty-label setting, the ground-truth label is directly shifted to the trigger pattern's location, explicitly training the model to treat the pattern as the object to track. In contrast, in the clean-label setting, the trigger pattern is overlaid on the real target, causing the model to learn the pattern as part of the target's appearance. As a result, during inference, the presence of the pattern alone can activate the backdoor and mislead the tracker

a poisoning dataset $\hat{\mathcal{D}} = \{\hat{\mathcal{V}}^i, \hat{\mathcal{B}}^i\}_{i=1}^{k}$ to poison the model trained on $\hat{\mathcal{D}}$ into a poisoned version $f\left(\ \cdot, \cdot, \cdot; \hat{\Theta}\right)$. The adversary leverages the following two functions to generate the poisoned dataset:

$$\begin{cases} \hat{\mathcal{V}}^i = G_t\left(\mathcal{V}^i, T\right) \\ \hat{\mathcal{B}}^i = M_t\left(\mathcal{B}^i\right) \end{cases} \tag{2}.$$

In Eq. (2), $G_t\left(\mathcal{V}^i, T\right)$ is utilized to add the trigger pattern t to the frames of the video $\mathcal{V}^i$, and $M_t\left(\mathcal{B}^i\right)$ means changing the bounding box to a specific target. If $M_t(\cdot)$ is the identity function, i.e., $\hat{\mathcal{B}}^i = M_t\left(\mathcal{B}^i\right) = \mathcal{B}^i$, it is called a clean-label attack. Otherwise, it is a dirty-label attack. The methods and strategies to implement $G_t(\cdot)$ and $M_t(\cdot)$ are described in Sections 3.2, 3.3, and 3.4.

2) Model training stage: In the model training stage, the victim model trainer leverages the poisoned dataset as (part of) the training dataset and develops a VOT model. The trainer has the flexibility to adopt any model architecture and training technique in order to acquire a high-performance model. Subsequently, following the training process, the backdoored model may be deployed to the cloud or devices by the trainer.

3) Inference stage: In the inference stage, given a benign video $\mathcal{V}$, the adversary utilizes another method $G_i(\cdot)$ to generate the trigger video $\mathcal{V}_t$, i.e., $\mathcal{V}_t = G_i\left(\mathcal{V}, T\right)$. After the victim model trainer deploys the backdoored model, the adversary can input the specific trigger sample $\mathcal{V}_t$ to the backdoored model to acquire the target predictions. Assuming that the predicted bounding box in the i-th frame of the video $\mathcal{V}_t$ is denoted as $p_i = \left(\tilde{x}_i, \tilde{y}_i, \tilde{w}_i, \tilde{h}_i\right)$ and the ground-truth bounding box is $\boldsymbol{b}_i = \left(x_i, y_i, w_i, h_i\right)$, we propose three types of targeted attacks for backdooring VOT models, including the size-manipulation attack, the trajectory-manipulation attack, and the hybrid attack as follows.

1) Size-manipulation attack: This attack aims to continuously change the size of the predicted bounding box, i.e., continuously expand or shrink the width and the height, as in Eq. (3).

$$\begin{cases} \tilde{w}_i > \tilde{w}_{i-1} \\ \tilde{h}_i > \tilde{h}_{i-1} \end{cases} \text{or} \begin{cases} \tilde{w}_i < \tilde{w}_{i-1} \\ \tilde{h}_i < \tilde{h}_{i-1} \end{cases} \tag{3}.$$

2) Trajectory-manipulation attack: This attack aims to manipulate the trajectory of the predicted bounding box, i.e., change the central coordinate $\widetilde{y_i}$. For instance, the adversary can make the predicted bounding boxes fixed in the frames after the initial frame, as in Eq. (4).

$$\begin{cases} \tilde{x}_i = x_1 \\ \tilde{y}_i = y_1 \end{cases}, \ i = 1, 2, \cdots, n \tag{4}.$$

Moreover, the adversary may also make the trajectory of the predicted bounding box follow a specific direction. For the convenience of evaluation, we utilize a line with slope β as the target trajectory in this paper:

$$\begin{cases} \tilde{x}_i = x_i \\ \tilde{y}_i = \beta\left(x_i - x_1\right) \end{cases}, \ i = 1, 2, \cdots, n \tag{5}.$$

3) Hybrid attack: This attack aims to completely control the prediction of the backdoored model and simultaneously manipulate the positions and sizes of the predicted bounding box. In the hybrid attack, the adversary may achieve both Eq. (3) and one of Eqs. (4) and (5).

3.2 Basic Strategy: Random Frame Poisoning Attack

From the formulation in Section 3.1, the key to the backdoor attack in the data collection stage is to design the poisoning functions $G_t(\cdot)$ and $M_t(\cdot)$ in Eq. (2). In a poison-only attack, the adversary needs to select some samples from the dataset and then add the trigger pattern to them. For the selection strategy, inspired by prior works[24], we propose our basic strategy: RFP.

Given the original dataset $\mathcal{D} = \{\left(\mathcal{V}^1, \mathcal{B}^1\right), \cdots, \left(\mathcal{V}^k, \mathcal{B}^k\right)\}$, in RFP, we mix and shuffle all the frames in the videos $\{\mathcal{V}^1, \cdots, \mathcal{V}^k\}$. Subsequently, we randomly select a subset of these frames and their corresponding bounding boxes (denoted as $\mathcal{D}_p$) to apply the poisoning functions $G_t(\cdot)$ and $M_t(\cdot)$. The implementation of the poisoning functions $G_t(\cdot)$ and $M_t(\cdot)$ is introduced in Section 3.3. The poison rate of the attack is defined as $\gamma = \left|\mathcal{D}_p\right| / |\mathcal{D}|$.

However, the effectiveness of RFP in attacking VOT models is limited in many cases (see Section 4). We argue that the ineffectiveness is largely due to the following two reasons.

First, the utilization of some training techniques by the model trainer has the potential to mitigate the impact of backdoors since the adversary cannot manipulate the training process in a poison-only attack. Specifically, the random sampling during training may also have a negative impact on the effectiveness of the backdoor attack. For instance, during the training phase, the Siamese network randomly selects two frames from a single video sequence as inputs to the network. Only when the model trainer selects two poisoned frames at the same time for training, the backdoor injection can have a significant effect. For an original poison rate $\gamma \in (0, 1)$, this leads to a reduction in the actual poisoning rate to $\gamma \times \gamma$, which suggests that the RFP makes the impact of the attack much less than expected. For a Transformer model, it selects multiple frames for training, and the attack effect is even much weaker.

Second, the RFP omits the correlation between different frames. The RFP strategy poisons random frames from different videos, which are unrelated to each other. However, in VOT, the temporal correlation between frames of the same video is important to the utility of the model. The RFP ignores this correlation, resulting in poor attack effectiveness.

3.3 Improved Strategy: Random Video Poisoning Attack

To tackle the above limitations, in this section, we propose our improved strategy: RVP. Unlike previous backdoor attacks against VOT models, RVP chooses to poison all the frames of the selected videos instead of the scattered frames. Poisoning a whole video can help the VOT models better capture the temporal correlation between frames and remember the injected trigger pattern. The comparison of the two strategies, RFP and RVP, is shown in Fig. 3.

Given the original dataset $\mathcal{D} = \{(\mathcal{V}^1, \mathcal{B}^1), \cdots, (\mathcal{V}^k, \mathcal{B}^k)\}$, in RVP, we randomly select a subset of videos from $\mathcal{D}$. The subset is also denoted as $\mathcal{D}_p$ and we keep the poison rate the same as the RFP attack. For each frame of each video in $\mathcal{D}_p$, we poison the image and the label using the poisoning functions $G_t(\cdot)$ and $M_t(\cdot)$.

1) Design of $G_t(\cdot)$: The main insight to poison the frame is that the model trained on the poisoned dataset is forced to track the trigger pattern instead of the target object. This involves injecting the trigger pattern into the area of the bounding box. Specifically, given the trigger pattern T, we utilize Eq. (6) to inject T to the i-th frame $\boldsymbol{I}_i^j$ of the j-th video $\mathcal{V}^j$ in $\mathcal{D}_p$:

$$\tilde{\boldsymbol{I}}_i^j = \min\left(\boldsymbol{I}_i^j + \boldsymbol{M} \odot T, 255\right) \tag{6.}$$

In Eq. (6), $\boldsymbol{M}$ represents a mask that is a binary matrix with the same size as the frame $\boldsymbol{I}_i^j$ and the symbol $\odot$ denotes the element-wise product (also known as the Hadamard product) of matrices. Assuming that the poisoned bounding box of $\tilde{\boldsymbol{I}}_i^j$ is $\tilde{\boldsymbol{b}}_i^j = \left(\tilde{\boldsymbol{x}}_i^j, \tilde{\boldsymbol{y}}_i^j, \tilde{\boldsymbol{w}}_i^j, \tilde{\boldsymbol{h}}_i^j\right)$, the element in the x-th row and y-th column of $\boldsymbol{M}$ is defined as follows.

$$\boldsymbol{M}(x, y) = \begin{cases} 1, & \begin{array}{l} x \in \left[\tilde{\boldsymbol{x}}_i^j - \dfrac{\tilde{\boldsymbol{w}}_i^j}{2},\ \tilde{\boldsymbol{x}}_i^j + \dfrac{\tilde{\boldsymbol{w}}_i^j}{2}\right] \\ \text{and } y \in \left[\tilde{\boldsymbol{y}}_i^j - \dfrac{\tilde{\boldsymbol{h}}_i^j}{2},\ \tilde{\boldsymbol{y}}_i^j + \dfrac{\tilde{\boldsymbol{h}}_i^j}{2}\right] \end{array} \\ 0, & \text{otherwise} \end{cases} \tag{7.}$$

The mask $\boldsymbol{M}$ ensures that the trigger pattern is added to the area included in the entire poisoned bounding box.

2) Design of $M_t(\cdot)$: As defined in Section 3.1, the attacks can be categorized into the dirty-label attack and the clean-label attack depending on whether the labels of the poisoned

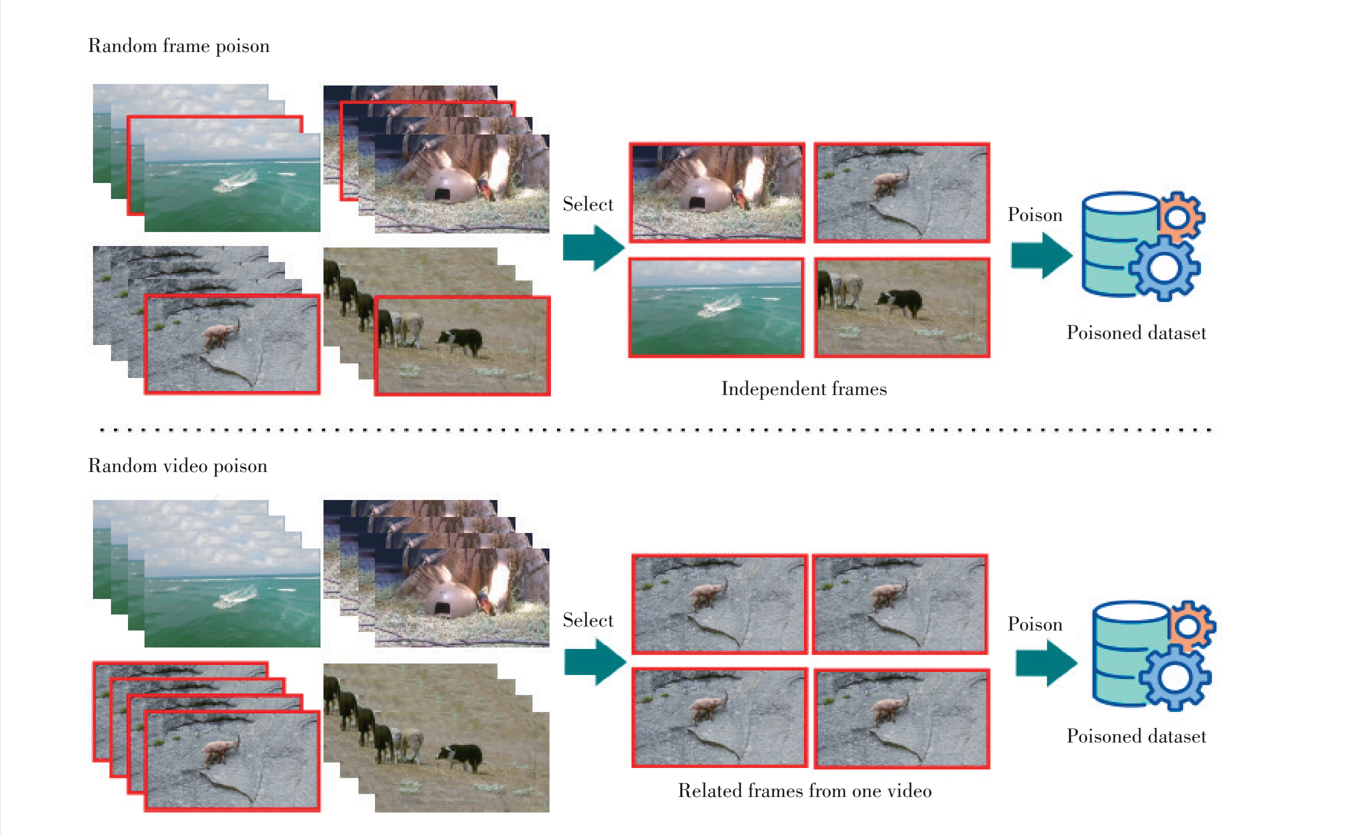


Figure 3. Comparison between the random frame poisoning (RFP) and the random video poisoning (RVP) attacks. RFP selects random frames from different videos while RVP selects all the frames from one video

frames are changed after the poisoning.

To design a dirty-label attack, we can further introduce a random offset $\Delta x_i^j, \Delta y_i^j$ to the position of the bounding box b_i^j and resize the bounding box into a square to facilitate the attack in the inference stage as:

$$M_t\left(x_i^j, y_i^j, w_i^j, h_i^j\right)=\left(\tilde{\boldsymbol{x}}_i^j, \tilde{\boldsymbol{y}}_i^j, \tilde{\boldsymbol{w}}_i^j, \tilde{\boldsymbol{h}}_i^j\right)=\left(x_i^j+\Delta x_i^j, y_i^j+\Delta y_i^j, s_i^j, s_i^j\right) \quad (8),$$

where $s_i^j = \min\left(w_i^j, h_i^j\right)$. Modifying the bounding box's position means it will no longer contain the target object completely, making the backdoored model more inclined to track the trigger pattern.

In a clean-label attack, we do not modify the ground-truth bounding boxes to achieve better stealthiness. Therefore, $M_t(\cdot)$ in a clean-label attack is as follows.

$$M_t\left(x_i^j, y_i^j, w_i^j, h_i^j\right) = \left(\tilde{\boldsymbol{x}}_i^j, \tilde{\boldsymbol{y}}_i^j, \tilde{\boldsymbol{w}}_i^j, \tilde{\boldsymbol{h}}_i^j\right) = \left(x_i^j, y_i^j, w_i^j, h_i^j\right) \quad (9).$$

3.4 Trigger Design

In this section, we introduce the design of the trigger pattern T used to poison the videos. The goal of designing the trigger pattern is two-fold:

1) The trigger pattern needs to be imperceptible to human inspection in order to make it difficult for the model trainer to detect and eliminate.

2) The trigger pattern needs to be scalable and flexible so that it can easily be rescaled to different sizes and different intensities to achieve different adversarial targets, such as expanding or shrinking the size of the bounding boxes.

Based on the above objectives, we adopt the sinusoidal signal as the trigger pattern T. Assuming that the size of the trigger pattern is $\mu \times \nu$, the element in the x-th row and y-th column is as follows.

$$T\left(x, y\right) = \delta \sin\left(\frac{2\pi y\lambda}{\nu}\right) \quad (10),$$

where λ is the frequency value, $x \in \left[1, \mu\right]$, and $y \in \left[1, \nu\right]$. Applying the sinusoidal signal as the trigger pattern has three advantages. First, the sinusoidal signal is generally invisible. Second, the sinusoidal signal is periodic. As such, it can be easily expanded to different sizes. Third, in the inference stage, we can enhance the backdoor attack by increasing the signal intensity parameter δ.

3.5 Achieving Different Backdoor Targets During Inference

After poisoning the dataset, the adversary can publicly release its dataset or sell it. A victim model trainer may get this poisoned dataset and leverage it to train a VOT model. This can lead to a backdoored model $f\left(\ \cdot\ , \cdot, \cdot ;\ \hat{\Theta}\right)$. If the victim model trainer deploys the backdoored model, the adversary can input the video with the trigger pattern into the model to achieve specific targets. First, the adversary needs to poison the template in the first frame of the video. Then, the adversary can achieve different adversarial targets in the following frames by adjusting the size and position of the trigger pattern.

1) Poisoning the template: We first need to add the trigger pattern into the template in the first frame. Unlike the previous poisoning of the dataset, in the inference stage, we only add the trigger pattern to a part of the template. Let the bounding box in the first frame (i. e., the template) be $\boldsymbol{b}_1 = \left(x_1, y_1, w_1, h_1\right)$. We inject the trigger pattern into the area represented by the bounding box:

$$\left(x_1, y_1, \sqrt{w_1 \cdot h_1}, \sqrt{w_1 \cdot h_1}, \alpha \neq 1\right) \quad (11).$$

This area is half the length and half the width of the template. We do not add the trigger pattern to the entire template area to facilitate the attack against the following frames.

Let the predicted bounding box in the $(i - 1)$-th frame be $\tilde{\boldsymbol{b}}_{i-1} = \left(\tilde{x}_{i-1}, \tilde{y}_{i-1}, \tilde{w}_{i-1}, \tilde{h}_{i-1}\right)$ and the ground-truth bounding box in the i-th frame be $\boldsymbol{b}_i = \left(x_i, y_i, w_i, h_i\right)$. The adversary then needs to carefully set the area of the trigger pattern in the frame to achieve different targets.

2) Size-manipulation attack: The goal of the size-manipulation attack is to make the bounding box continuously expand or shrink, but not change the position of the bounding box. As such, we inject the trigger pattern into the bounding box represented by:

$$\left(x_i, y_i, \alpha\sqrt{\tilde{w}_{i-1} \cdot \tilde{h}_{i-1}}, \alpha\sqrt{\tilde{w}_{i-1} \cdot \tilde{h}_{i-1}}, \alpha \neq 1\right) \quad (12),$$

where α is the scaling factor. If $\alpha > 1$, the predicted bounding box will be larger in the following frame. Otherwise, the predicted bounding box will shrink.

3) Trajectory-manipulation attack: The objective of the size-manipulation attack is to manipulate the trajectory of the predicted bounding box. We consider two different manipulations as shown in Eqs. (4) and (5).

To achieve Eq. (4), i.e., make the bounding box fixed, we fix the central coordinates of the trigger pattern. Therefore, in the i-th frame, we inject the trigger pattern T' into the bounding box represented by:

$$\left(x_1, y_1, w_i, h_i\right) \quad (13).$$

To achieve Eq. (5), i.e., make the bounding box move along the vertical line, the bounding box of the trigger pattern T' can be represented by:

$$\left(x_i, \beta(x_i - x_1), w_i, h_i\right) \quad (14).$$

4) Hybrid attack: The above two attacks respectively control the size and the position of the predicted bounding box. The

hybrid attack aims to control both simultaneously. As such, in the hybrid attack, we can combine the above two attacks. For example, to make the position fixed and the size expand continually, we can inject the trigger pattern T' into the area as:

$$\left(x_1, y_1, \alpha\tilde{w}_{i-1}, \alpha\tilde{h}_{i-1}\right) \qquad (15).$$

5) Trigger pattern in the inference stage: In the inference stage, we also leverage the sinusoidal signal as the trigger pattern. However, the trigger pattern in the inference stage does not need to be completely invisible. Therefore, we can use stronger signals (i.e., increase the intensity parameter δ) to enhance the attack effect.

4 Evaluation

In this section, we empirically evaluate the effectiveness of our proposed poison-only and targeted backdoor attacks against VOT models, including RFP and RVP, to implement three different targets.

4.1 Experimental Settings

We evaluate our proposed backdoor attacks on two different models, SiamFC++[42] and SiamRPN++[8]. We train the models on two different datasets, namely OTB100[43] and GOT10K[44]. OTB100 is a general tracking dataset containing 100 videos, and GOT10K provides 180 sequences for bounding box regression testing. For the attack settings, we set the default poison rate to 10% in our experiments.

4.2 Results of Size-Manipulation Attack

To evaluate the effectiveness of the size-manipulation attack, we employ the size ratio (SR) metric, which is defined as the ratio of the predicted bounding box area at the 10th frame to the area of the initial template bounding box. A higher SR for expansion attacks (target scaling factor $\alpha > 1.0$) or a lower SR for shrinking attacks (target $\alpha < 1.0$) indicates a more successful manipulation of the bounding box size as intended by the adversary. An SR close to 1.0 when α is 1.0 would indicate minimal size change, similar to benign behavior, though the attack still aims to lock onto the trigger.

Table 1 demonstrates the effectiveness of the size-manipulation attacks. Across both GOT10K and OTB100 datasets and for SiamFC++ and SiamRPN++ models, our proposed RVP strategy consistently outperforms the RFP baseline. RVP methods achieve more significant size alterations, evident by lower SRs for shrinking targets ($\alpha < 1.0$) and higher SRs for expansion targets ($\alpha > 1.0$) compared to benign models and RFP. Both dirty-label (RVP-D) and clean-label (RVP-C) variants of RVP prove effective, with RVP-D often showing a slight edge in expansion and RVP-C being highly competitive, especially for shrinking. The degree of size-manipulation generally correlates well with α, highlighting the attack's controllability and confirming the vulnerability of VOT models to these targeted backdoors.

Table 1. SR results of size-manipulation attacks

Dataset	Model	Metric	α				
			0.90	0.95	1.00	1.05	1.10
GOT10K	SiamFC++	Benign	1.051	1.053	1.062	1.070	1.071
		RFP-D	0.743	0.854	1.052	1.290	1.544
		RFP-C	1.002	0.999	1.072	1.214	1.351
		RVP-D	0.732	0.834	1.086	1.406	1.781
		RVP-C	0.728	0.823	1.067	1.387	1.733
	SiamRPN++	Benign	0.979	1.001	1.054	1.114	0.120
		RFP-D	0.725	0.829	1.057	1.335	1.526
		RFP-C	0.873	0.918	1.071	1.291	1.456
		RVP-D	0.734	0.839	1.056	1.318	1.479
		RVP-C	0.743	0.853	1.064	1.356	1.657
OTB100	SiamFC++	Benign	1.128	1.129	1.132	1.121	1.106
		RFP-D	1.736	0.848	1.086	1.396	1.734
		RFP-C	1.064	1.056	1.132	1.544	2.049
		RVP-D	0.718	0.821	1.122	1.544	2.049
		RVP-C	0.726	0.814	1.094	1.501	1.957
	SiamRPN++	Benign	0.994	0.998	1.043	1.093	1.105
		RFP-D	0.864	0.875	1.047	1.275	1.338
		RFP-C	0.949	0.948	1.067	1.253	1.322
		RVP-D	0.863	0.881	1.041	1.257	1.291
		RVP-C	0.864	0.892	1.062	1.326	1.502

RFP: Random Frame Poisoning Attack

RVP: Random Video Poisoning Attack

SR: size ratio

4.3 Results of Trajectory-Manipulation Attack

To evaluate the success of the trajectory-manipulation attack, we measure the slope of the predicted trajectory. This is determined by fitting a straight line to the central coordinates of the predicted bounding boxes over the first 10 frames of a video sequence. For an attack aiming to fix the object's position (denoted as "fix" in Table 2, target slope $\beta = 0$), a slope close to zero indicates success. For attacks aiming to move the object along a line with a specific slope β, the achieved slope of the predicted trajectory should approximate this target β.

Table 2 presents the achieved slopes for trajectory-manipulation attacks under different target β. For fix attacks (target $\beta = 0$), both RVP and RFP strategies substantially reduce the trajectory slope compared to the benign model's natural movement, effectively immobilizing the tracked object. Notably, RVP methods often yield slopes closer to the ideal zero. When a specific non-zero trajectory slope β (ranging from 0.1 to 0.4) is targeted, RVP again demonstrates superior control. The achieved slopes using RVP closely approximate the intended β values, indicating a stronger capability in forcing the model to follow a predefined path compared to RFP. These trends of RVP ' s enhanced effectiveness hold consistently across the SiamFC++ and SiamRPN++ models on both GOT10K and OTB100 datasets, underscoring the vulnerability of VOT models to such controlled trajectory-manipulations.

Table 2. Slopes of trajectory-manipulation attacks under different β

Dataset	Model	Metric	β				
			fix	0.1	0.2	0.3	0.4
GOT10K	SiamFC++	Benign	0.036	0.049	0.056	0.056	0.059
		RFP-D	0.004	0.097	0.166	0.213	0.276
		RFP-C	0.008	0.089	0.122	0.140	0.150
		RVP-D	0.005	0.093	0.168	0.231	0.291
		RVP-C	0.004	0.095	0.167	0.232	0.291
	SiamRPN++	Benign	0.033	0.040	0.048	0.050	0.051
		RFP-D	0.006	0.086	0.161	0.221	0.276
		RFP-C	0.006	0.077	0.150	0.207	0.253
		RVP-D	0.006	0.086	0.161	0.221	0.276
		RVP-C	0.006	0.077	0.150	0.207	0.253
OTB100	SiamFC++	Benign	0.029	0.035	0.033	0.032	0.032
		RFP-D	0.006	0.099	0.185	0.255	0.318
		RFP-C	0.010	0.077	0.111	0.133	0.153
		RVP-D	0.006	0.091	0.181	0.257	0.324
		RVP-C	0.007	0.096	0.183	0.260	0.326
	SiamRPN++	Benign	0.029	0.032	0.033	0.034	0.032
		RFP-D	0.009	0.077	0.161	0.217	0.260
		RFP-C	0.016	0.046	0.051	0.050	0.048
		RVP-D	0.008	0.078	0.171	0.235	0.288
		RVP-C	0.011	0.070	0.145	0.187	0.216

RFP: Random Frame Poisoning Attack
RVP: Random Video Poisoning Attack
SR: size ratio

4.4 Results of Hybrid Attack

For the hybrid attack, which simultaneously manipulates both the size and trajectory of the predicted bounding box, we employ three metrics. In addition to the SR and slope, we adopt Intersection over Union (IoU) as a comprehensive metric, which measures the overlap between the predicted bounding box and an adversarially defined target bounding box. This target bounding box incorporates both the intended size and trajectory manipulation at the 10th frame.

Table 3 presents the performance of hybrid attacks, combining size (shrink/expand) and trajectory (fix/move) manipulations. The results clearly demonstrate the potency of these combined attacks. Our proposed RVP methods consistently outperform RFP across all hybrid attack modes and evaluation metrics. RVP achieves SR and slope values closer to the adversarial targets while, crucially, yielding significantly higher IoU scores. For example, in the "Expand & Move" mode on GOT10K with SiamFC++, RVP-D achieves an IoU of 0.736, notably higher than RFP-D's 0.631. This superior IoU for RVP indicates its enhanced capability to precisely control both the size and the path of the tracked object simultaneously, reinforcing its effectiveness for complex backdoor injection in VOT models. These trends are consistent across different models and datasets.

4.5 Evaluation on Function Preservation

In the task of VOT, the goal is to train a tracker to predict the position of the bounding box in a sequence of video frames as accurately as possible. There are various datasets[43–45], each with different testing preferences. According to the requirements of benchmarking, we use the following three metrics to evaluate tracker performance: 1) Precision (Prec),

Table 3. SRs, slopes, and IoU of hybrid attacks

Dataset	Attack Mode		Shrink (α = 0.9)						Expand (α = 1.1)					
			Fix			Move (β = 0.1)			Fix			Move (β = 0.1)		
	Model	Metric	SR	Slope	IoU	SR	Slope	IoU	SR	Slope	IoU	SR	Slope	IoU
GOT10K	Siam-FC++	Benign	1.040	0.039	0.294	1.050	0.050	0.190	1.186	0.042	0.390	1.160	0.042	0.313
		RFP-D	0.672	0.004	0.529	0.680	0.097	0.517	1.756	0.008	0.668	1.723	0.088	0.631
		RFP-C	0.784	0.018	0.432	0.840	0.069	0.313	1.766	0.013	0.664	1.698	0.082	0.609
		RVP-D	0.671	0.006	0.535	0.671	0.092	0.525	2.082	0.008	0.763	2.054	0.086	0.736
		RVP-C	0.644	0.006	0.558	0.649	0.093	0.538	2.003	0.009	0.755	1.977	0.087	0.718
	Siam-RPN++	Benign	0.995	0.037	0.302	1.015	0.040	0.182	1.116	0.040	0.373	1.096	0.039	0.290
		RFP-D	0.710	0.011	0.489	0.733	0.075	0.433	1.552	0.017	0.572	1.492	0.052	0.494
		RFP-C	0.883	0.028	0.363	0.952	0.040	0.209	1.455	0.018	0.535	1.370	0.038	0.420
		RVP-D	0.719	0.010	0.484	0.743	0.076	0.430	1.511	0.017	0.559	1.430	0.048	0.469
		RVP-C	0.747	0.012	0.472	0.797	0.066	0.380	1.668	0.012	0.623	1.611	0.056	0.542
OTB100	Siam-FC++	Benign	1.094	0.030	0.277	1.119	0.034	0.117	1.186	0.032	0.398	1.164	0.032	0.324
		RFP-D	0.704	0.005	0.504	0.706	0.098	0.494	1.867	0.007	0.712	1.821	0.098	0.675
		RFP-C	0.832	0.017	0.412	0.897	0.057	0.251	1.873	0.013	0.701	1.712	0.076	0.600
		RVP-D	0.684	0.007	0.524	0.707	0.092	0.497	2.330	0.008	0.853	2.286	0.090	0.803
		RVP-C	0.672	0.007	0.535	0.694	0.095	0.501	2.204	0.010	0.824	2.140	0.096	0.779
	Siam-RPN++	Benign	1.014	0.030	0.283	1.041	0.033	0.109	1.113	0.032	0.376	1.102	0.031	0.300
		RFP-D	0.874	0.014	0.377	0.895	0.065	0.290	1.335	0.025	0.472	1.266	0.042	0.393
		RFP-C	0.977	0.027	0.310	1.023	0.033	0.118	1.311	0.028	0.468	1.252	0.033	0.363
		RVP-D	0.874	0.013	0.379	0.897	0.066	0.289	1.295	0.027	0.455	1.237	0.039	0.376
		RVP-C	0.903	0.018	0.362	0.948	0.053	0.235	1.453	0.022	0.529	1.370	0.043	0.430

IoU: Intersection over Union RFP: Random Frame Poisoning Attack RVP: Random Video Poisoning Attack SR: size ratio which indicates the positional accuracy, i.e., whether the distance between the predicted bounding box and the true bounding box is less than 20 pixels in the image; 2) Area Under the Curve (AUC), which represents the area under the success rate curve, used to measure the overlap ratio between the predicted box and the true bounding box; 3) Success rate at 50% overlap (SR50), which reflects the tracking success rate when the overlap exceeds the threshold of 0.5.

The results presented in Table 4 indicate that our RVP-based backdoor attacks exhibit strong function preservation. For both SiamFC++ and SiamRPN++ models on the GOT10K and OTB100 datasets, the performance metrics (AUC, SR50, and Prec) of the backdoored models (RVP-D and RVP-C) remain remarkably close to those of the benign models. For instance, on GOT10K, the SiamFC++ Benign model achieves an AUC of 0.721 7, while RVP-D achieves 0.717 5 and RVP-C achieves 0.708 5. Similarly, for SiamRPN++ on OTB100, the Benign model's Precision is 85.65, whereas RVP-D's is 85.49 and RVP-C's is 82.25. The slight degradation observed, particularly with RVP-C, is minimal and generally acceptable, considering the effectiveness of the injected backdoor. The RVP-D strategy, in particular, demonstrates excellent stealth, with performance nearly identical to the benign model in several cases. This high degree of function preservation suggests that the backdoor can be effectively concealed within the VOT model without significantly impairing its primary tracking capabilities on normal, benign data, making the attack difficult to detect through standard performance evaluations.

Table 4. Results of evaluation on the function preservation

Model	Metric	GOT10K		OTB100	
		AUC	SR50	AUC	Prec
SiamFC++	Benign	0.721 7	0.861 5	63.22	83.83
	RVP-D	0.717 5	0.857 9	62.91	82.49
	RVP-C	0.708 5	0.845 3	60.70	80.26
SiamRPN++	Benign	0.664 8	0.772 0	65.04	85.65
	RVP-D	0.666 6	0.771 5	64.58	85.49
	RVP-C	0.654 1	0.758 4	62.13	82.25

AUC: Area Under the Curve
Prec: Precision
RVP: Random Video Poisoning
SR50: Success rate at 50% overlap

5 Conclusions

In this paper, we introduce and thoroughly investigate poison-only and targeted backdoor attacks against VOT models. We define three distinct attack variants (size-manipulation, trajectory-manipulation, and hybrid attacks) and propose an effective RVP strategy that significantly outperforms baseline methods by leveraging temporal correlations in video data. Our extensive experiments demonstrate that RVP can successfully inject controllable backdoors into VOT models, achieving high attack success rates while maintaining remarkable function preservation on benign data, thus ensuring stealth. Interestingly, while devised for attack analysis, the core mechanism of embedding specific, detectable behaviors into models via data manipulation holds potential for positive applications. The imperceptible and robust nature of the injected patterns suggests that similar techniques could be adapted for dataset or model watermarking[46 – 49], thereby contributing to copyright protection and ownership verification in the domain of visual tracking and beyond.

Based on our findings, future research will explore several promising directions. First, we will focus on applying the core principles of the RVP attack to beneficial areas, for example, adapting the technology to create reliable digital watermarks to protect the intellectual property of VOT datasets and models. At the same time, a more in-depth investigation into the attack's hyperparameters is also crucial. This includes a systematic analysis of the poison rate and an exploration of the trigger pattern's own parameters (such as the frequency λ and intensity δ of the sinusoidal signal), in order to understand the key trade-offs between attack effectiveness and stealthiness. Furthermore, the vulnerabilities revealed in this paper also compel us to develop corresponding defense mechanisms, especially those capable of detecting and mitigating backdoor attacks that leverage the temporal correlations of video data, which traditional defense methods might overlook. Finally, we plan to expand the scope of our research by extending the attack framework to more complex scenarios (such as multi-object tracking) and evaluating its effectiveness on a broader range of advanced tracker architectures, particularly emerging Transformer-based models.

References

[1] KRISTAN M, MATAS J, LEONARDIS A, et al. The visual object tracking VOT2015 challenge results [C]//Proc. IEEE International Conference on Computer Vision Workshop (ICCVW). IEEE, 2015: 564 – 586. DOI: 10.1109/ICCVW.2015.79

[2] CHEN F, WANG X D, ZHAO Y X, et al. Visual object tracking: a survey [J]. Computer vision and image understanding, 2022, 222: 103508. DOI: 10.1016/j.cviu.2022.103508

[3] HONG L Y, YAN S L, ZHANG R R, et al. OneTracker: unifying visual object tracking with foundation models and efficient tuning [C]//Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2024: 19079 – 19091. DOI: 10.1109/CVPR52733.2024.01805

[4] CHEN X, PENG H W, WANG D, et al. SeqTrack: sequence to sequence learning for visual object tracking [C]//Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2023: 14572 – 14581. DOI: 10.1109/CVPR52729.2023.01400

[5] CHENG G, YUAN X, YAO X W, et al. Towards large-scale small object detection: survey and benchmarks [J]. IEEE transactions on pattern analysis and machine intelligence, 2023, 45(11): 13467 – 13488. DOI: 10.1109/TPAMI.2023.3290594

[6] TANG H, LIANG K J, GRAUMAN K, et al. Egotracks: a long-term egocentric visual object tracking dataset [C]//Advances in Neural Information Processing Systems 36. NeurIPS, 2023: 75716 – 75739

[7] CEN M B, JUNG C. Fully convolutional Siamese fusion networks for object tracking [C]//Proc. 25th IEEE International Conference on Image Processing (ICIP). IEEE, 2018: 3718 – 3722. DOI: 10.1109/ICIP.2018.8451102

[8] LI B, WU W, WANG Q, et al. SiamRPN++: evolution of Siamese visual tracking with very deep networks [C]//Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2019. DOI: 10.1109/cvpr.2019.00441

[9] BHAT G, DANELLJAN M, VAN GOOL L, et al. Learning discriminative model prediction for tracking [C]//Proc. IEEE/CVF International Conference on Computer Vision (ICCV). IEEE, 2019: 6181 – 6190. DOI: 10.1109/ICCV.2019.00628

[10] WEI X, BAI Y F, ZHENG Y C, et al. Autoregressive visual tracking [C]// Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2023: 9697 – 9706. DOI: 10.1109/ CVPR52729.2023.00935

[11] LI Y M, JIANG Y, LI Z F, et al. Backdoor learning: a survey [J]. IEEE transactions on neural networks and learning systems, 2024, 35(1): 5 – 22. DOI: 10.1109/TNNLS.2022.3182979

[12] CHEN Y K, SHAO S, HUANG E H, et al. Refine: inversion-free backdoor defense via model reprogramming[C]//International Conference on Learning Representations. ICLR, 2025: 1 – 28

[13] GU T Y, LIU K, DOLAN-GAVITT B, et al. BadNets: evaluating backdooring attacks on deep neural networks [J]. IEEE access, 2019, 7: 47230 – 47244

[14] WEI C, WANG Y, GAO K F, et al. PointNCBW: toward dataset ownership verification for point clouds via negative clean-label backdoor watermark [J]. IEEE transactions on information forensics and security, 2024, 20: 191 – 206. DOI: 10.1109/TIFS.2024.3492792

[15] ZHU R, TANG D, TANG S Y, et al. Gradient shaping: enhancing backdoor attack against reverse engineering [C]//Proc. 2024 Network and Distributed System Security Symposium. Internet Society, 2024. DOI: 10.14722/ndss.2024.24450

[16] LIN J Y, XU L, LIU Y Q, et al. Composite backdoor attack for deep neural network by mixing existing benign features [C]//Proc. 2020 ACM SIGSAC Conference on Computer and Communications Security. ACM, 2020: 113 – 131. DOI: 10.1145/3372297.3423362

[17] LI Y Z, LI Y M, WU B Y, et al. Invisible backdoor attack with sample-specific triggers [C]//Proc. IEEE/CVF International Conference on Computer Vision (ICCV). IEEE, 2021: 16443 – 16452. DOI: 10.1109/ ICCV48922.2021.01615

[18] LYU P Z, YUE C, LIANG R G, et al. A data-free backdoor injection approach in neural networks[C]//32nd USENIX Conference on Security Symposium. USENIX, 2023: 2671 – 2688

[19] LI Y Z, LI T L, CHEN K J, et al. Badedit: backdooring large language models by model editing [C]//International Conference on Learning Representations. ICLR, 2024: 1 – 18

[20] PEI H Z, JIA J Y, GUO W B, et al. TextGuard: provable defense against backdoor attacks on text classification [C]//Proc. 2024 Network and Distributed System Security Symposium. Internet Society, 2024. DOI: 10.14722/ndss.2024.24090

[21] SHEN L J, JI S L, ZHANG X H, et al. Backdoor pre-trained models can transfer to all [C]//Proc. 2021 ACM SIGSAC Conference on Computer and Communications Security. ACM, 2021: 3141 – 3158. DOI: 10.1145/ 3460120.3485370

[22] LI Y M, ZHONG H X, MA X J, et al. Few-shot backdoor attacks on visual object tracking [C]//International Conference on Learning Representations. ICLR, 2022: 1 – 21

[23] CHENG Z Y, WU B Y, ZHANG Z Y, et al. TAT: targeted backdoor attacks against visual object tracking [J]. Pattern recognition, 2023, 142: 109629. DOI: 10.1016/j.patcog.2023.109629

[24] HUANG B, YU J, CHEN Y, et al. BadTrack: a poison-only backdoor attack on visual object tracking [C]//Proc. 37th International Conference on Neural Information Processing Systems. NIPS, 2023: 41778 – 41796

[25] ADRIAN A I, ISMET P, PETRU P. An overview of intelligent surveillance systems development [C]//Proc. International Symposium on Electronics and Telecommunications (ISETC). IEEE, 2018: 1 – 6. DOI: 10.1109/ISETC.2018.8584003

[26] LU J H, HUANG D, WANG Y H, et al. Scaling and occlusion robust athlete tracking in sports videos [C]//Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2016: 1526 – 1530. DOI: 10.1109/ICASSP.2016.7471932

[27] WENG X S, WANG J R, HELD D, et al. 3D multi-object tracking: a baseline and new evaluation metrics [C]//Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2020: 10359 – 10366. DOI: 10.1109/IROS45743.2020.9341164

[28] WANG J S, TAO B, GONG Z Y, et al. A mobile robotic measurement system for large-scale complex components based on optical scanning and visual tracking [J]. Robotics and computer-integrated manufacturing, 2021, 67: 102010. DOI: 10.1016/j.rcim.2020.102010

[29] MARVASTI-ZADEH S M, CHENG L, GHANEI-YAKHDAN H, et al. Deep learning for visual tracking: a comprehensive survey [J]. IEEE transactions on intelligent transportation systems, 2021, 23(5): 3943 – 3968. DOI: 10.1109/TITS.2020.3046478

[30] CHOPRA S, HADSELL R, LECUN Y. Learning a similarity metric discriminatively, with application to face verification [C]//Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). IEEE, 2005: 539 – 546. DOI: 10.1109/CVPR.2005.202

[31] CHEN X, YAN B, ZHU J W, et al. Transformer tracking [C]//2021 IEEE/ CVF Conference on Computer Vision and Pattern Recognition. IEEE, 2021: 1571 – 1580. DOI: 10.1109/CVPR46437.2021.00803

[32] XU X, HUANG K Z, LI Y M, et al. Towards reliable and efficient backdoor trigger inversion via decoupling benign features [C]//International Conference on Learning Representations. ICLR, 2024: 1 – 25

[33] LI C J, PANG R, CAO B C, et al. On the difficulty of defending contrastive learning against backdoor attacks [C]//33rd USENIX Security Symposium. USENIX, 2024: 2901 – 2918

[34] HUANG K Z, LI Y M, WU B Y, et al. Backdoor defense via decoupling the training process [C]//International Conference on Learning Representations. ICLR, 2022: 1 – 25

[35] LI S F, LIU H, DONG T, et al. Hidden backdoors in human-centric language models [C]//Proc. 2021 ACM SIGSAC Conference on Computer and Communications Security. ACM, 2021: 3123 – 3140. DOI: 10.1145/ 3460120.3484576

[36] HUANG H, ZHAO Z Y, BACKES M, et al. Composite backdoor attacks against large language models [C]//Proc. Findings of the Association for Computational Linguistics. NAACL, 2024: 1459 – 1472. DOI: 10.18653/ v1/2024.findings-naacl.94

[37] YANG W Y, SHAO S, YANG Y, et al. Watermarking in secure federated learning: a verification framework based on client-side backdooring [J]. ACM transactions on intelligent systems and technology, 2024, 15(1): 1 – 25. DOI: 10.1145/3630636

[38] SHAO S, YANG W Y, GU H L, et al. FedTracker: furnishing ownership verification and traceability for federated learning model [J]. IEEE transactions on dependable and secure computing, 2025, 22(1): 114 – 131. DOI: 10.1109/TDSC.2024.3390761

[39] LI Y M, YAN K Y, SHAO S, et al. CBW: towards dataset ownership verification for speaker verification via clustering-based backdoor watermarking [EB/OL]. (2025-03-02)[2025-07-15]. https://arxiv.org/abs/2503.05794

[40] ZHAI T Q, LI Y M, ZHANG Z Q, et al. Backdoor attack against speaker verification [C]//Proc. 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2021: 2560 – 2564. DOI: 10.1109/ICASSP39728.2021.9413468

[41] HAN X S, WU Y T, ZHANG Q J, et al. Backdooring multimodal learning [C]//Proc. IEEE Symposium on Security and Privacy (SP). IEEE, 2024: 3385 – 3403. DOI: 10.1109/SP54263.2024.00031

[42] XU Y D, WANG Z Y, LI Z X, et al. SiamFC++: towards robust and accurate visual tracking with target estimation guidelines [C]//Proc. AAAI conference on artificial intelligence. AAAI, 2020: 12549 – 12556. DOI:

10.1609/aaai.v34i07.6944

[43] WU Y, LIM J, YANG M H. Object tracking benchmark [J]. IEEE transactions on pattern analysis and machine intelligence, 2015, 37(9): 1834 – 1848. DOI: 10.1109/TPAMI.2014.2388226

[44] HUANG L H, ZHAO X, HUANG K Q. GOT-10k: a large high-diversity benchmark for generic object tracking in the wild [J]. IEEE transactions on pattern analysis and machine intelligence, 2021, 43(5): 1562 – 1577. DOI: 10.1109/TPAMI.2019.2957464

[45] FAN H, LIN L T, YANG F, et al. LaSOT: a high-quality benchmark for large-scale single object tracking [C]//Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2019: 5369 – 5378. DOI: 10.1109/CVPR.2019.00552

[46] SHAO S, LI Y M, YAO H W, et al. Explanation as a watermark: towards harmless and multi-bit model ownership verification via watermarking feature attribution [C]//Proc. 2025 Network and Distributed System Security Symposium. Internet Society, 2025. DOI: 10.14722/ndss.2025.230338

[47] REN K, YANG Z Q, LU L, et al. Sok: on the role and future of AIGC watermarking in the era of gen-AI [EB/OL]. (2024-11-18) [2025-07-15]. https://arxiv.org/abs/2411.11478

[48] LI Y M, ZHU M Y, YANG X, et al. Black-box dataset ownership verification via backdoor watermarking [J]. IEEE transactions on information forensics and security, 2023, 18: 2318 – 2332. DOI: 10.1109/TIFS.2023.3265535

[49] LI Y M, SHAO S, HE Y, et al. Rethinking data protection in the (generative) artificial intelligence era. [EB/OL]. (2025-07-03)[2025-07-15]. https://arxiv.org/abs/2507.03034

Biographies

GU Wei is currently pursuing a master's degree at the School of Cyber Science and Technology and the State Key Laboratory of Blockchain and Data Security, Zhejiang University, China. Before that, he received a BE degree in computer science and technology from Zhuoyue Honors College, Hangzhou Dianzi University, China in 2023. His research interests include LLM security and AI safety.

SHAO Shuo is currently pursuing a PhD degree at the School of Cyber Science and Technology and the State Key Laboratory of Blockchain and Data Security, Zhejiang University, China. Before that, he received a BE degree from the School of Computer Science and Technology, Central South University, China in 2022. His research interests include AI copyright protection, data protection, and LLM safety. He has published a series of papers in top-tier conferences and journals such as NDSS, ICLR, TIFS, and TDSC, among others, and actively serves as a reviewer for NeurIPS, ICML, TCSVT, TII and other leading venues.

ZHOU Lingtao is currently pursuing a BE degree at Shandong University, China. His research interests include backdoor attacks and AI security.

QIN Zhan (qinzhan@zju.edu.cn) is currently a tenured associate professor, with both the College of Computer Science and Technology and the Institute of Cyberspace Research (ICSR) at Zhejiang University, China. He was an assistant professor at the Department of Electrical and Computer Engineering, the University of Texas at San Antonio, USA after receiving the PhD degree from the Computer Science and Engineering department, State University of New York at Buffalo, USA in 2017. His current research interests include data security and privacy, secure computation outsourcing, artificial intelligence security, and cyber-physical security in the context of the Internet of Things. His works explore and develop novel security-sensitive algorithms and protocols for computation and communication in the general context of Cloud and Internet devices.

REN Kui is a professor and the dean of the School of Cyber Science and Technology at Zhejiang University. Before that, he was a SUNY Empire Innovation Professor at State University of New York at Buffalo, USA. He received his PhD degree in electrical and computer engineering from Worcester Polytechnic Institute, USA. His current research interests include data security, IoT security, AI security, and privacy. He received the Guohua Distinguished Scholar Award from Zhejiang University, IEEE CISTC Technical Recognition Award, SUNY Chancellor ' s Research Excellence Award, Sigma Xi Research Excellence Award, and NSF CAREER Award. He has published extensively in peer-reviewed journals and conferences and received the Test-of-Time Paper Award from IEEE INFOCOM and many Best Paper Awards from IEEE and ACM. He currently serves as Chair of SIGSAC of ACM China. He is a Fellow of IEEE, a Fellow of ACM, and a Clarivate Highly-Cited Researcher.