WIP: Hijacking Attacks on UAV Follow-Me Systems in Realistic Scenarios

Jiarui Li, Joseph Brewington, Qingzhao Zhang, Z. Morley Mao
University of Michigan
{jiaruili, brewing, qzzhang, zmao}@umich.edu

Abstract

Modern vision-based object tracking is a vital component of Unmanned Aerial Vehicle (UAV) systems. It enables advanced applications such as follow-me, which allows a drone to automatically track and follow a subject. While a wealth of research explored the vulnerabilities of object tracking algorithms, there lacks a comprehensive analysis on whether the vulnerabilities can be exploited on real UAV systems, considering challenges including physical constraints, real-world uncertainties, and limited attacker's knowledge. To bridge the knowledge gap, we design a hijacking attack that deceives the UAV follow-me mode to track a wrong subject by leveraging existing object tracking attacks. We thoroughly analyze its feasibility in real-world scenarios. With insights from the study, we are able to improve the attack success rate on the UAV follow-me application from 47% to 95% by leveraging inaccuracies of sensor measurements and instability of the gimbal camera, which indicates a realistic system exploit.

1 Introduction

Visual object tracking provides critical capabilities for UAV systems, with applications that span multiple domains, including videography, surveillance, and navigation [14, 16, 23, 31]. Mainstream industrial products have widely implemented vision-based target tracking algorithm in their commercial UAV tracking system [3, 12, 37]. The follow-me mode is a popular capability enabled by the development of visual object tracking, which achieves autonomous target following without any human pilots [27, 40, 45]. Despite the popularity of the follow-me mode, the operational safety of these systems is based on the security of the underlying object tracking algorithm. A compromised tracking mechanism can have severe consequences, including UAV disorientation, deviation from correct flight paths, and, in worst-case scenarios, complete loss of the UAV.

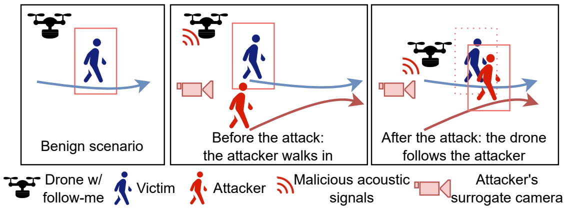

Figure 1: Illustration of our hijacking attack on UAV follow-me. A physical attacker uses a crafted adversarial trajectory to approach the intended target, misleading the drone's vision object tracking system into following the attacker instead.

Previous research has examined the security vulnerabilities of tracking algorithms in autonomous driving and surveillance applications [18, 25, 26, 30, 39]. Several studies [18, 25, 26] employ large adversarial patches to manipulate detection bounding boxes, potentially inducing false driving decisions. AttrackZone [30] uses projector-based perturbations, which work mainly in a dim environment, against Siamese-based trackers. Wang et al. [39] introduce a trajectory-based attack that triggers ID switches between tracked objects. The efficacy of such attacks against UAV tracking systems with complex real-world challenges, including physical constraints, uncertainties, and limited adversarial knowledge, remains unknown, as depicted in Table 1. On the other hand, the existing UAV security work focuses mainly on sensor-level vulnerabilities, including GPS, stereo camera, and MEMS sensors [10, 34, 35, 38, 49], without a thorough evaluation of the attack impact on practical applications like follow-me.

We implement a hijacking attack against UAV follow-me, by exploiting UAV visual object tracking vulnerability, as demonstrated in Figure 1. We assume that the drone implements vision-based follow-me based on multi-object tracking (MOT) algorithms. In benign scenarios, the drone tracks and follows the right person (blue). When the attack starts, an adversary (red) physically approaches the legitimate person along a malicious trajectory, optimized to trigger an MOT ID switch, following previous work [39]. If successful, the drone erroneously follows the wrong target. This vulnerability can potentially cause vehicle theft, particularly when the legitimate user relies on automatic following and is engaged in their primary activity (e.g. running, cycling).

Table 1: Existing attacks on object tracking algorithms

	ControlLoc [26]	AttrackZone [30]	ADVTRAJ [39]	**Ours**
Physical Damage/Loss	●	●	○	●
Victim Uncertainty	◒[†]	●[‡]	○	●
Online Latency	N/A	○	○	●
Inaccurate Viewpoint	○	○	○	●

[†] Optimized patch location and pattern over offline dataset and specific object detector, but remain static during the attack.

[‡] Selected attack zones based on real-time situation.

Importantly, we perform a detailed analysis to assess the feasibility of the hijacking attack in realistic scenarios. We developed a high-fidelity simulation environment that incorporates a range of real-world uncertainties and physical constraints. Through extensive simulation experiments, we identified that these real-world factors can either pose challenges to the attack's feasibility or be exploited to the attacker's advantage. For instance, lacking access to the drone's view, the attacker must employ a surrogate camera to perform a black-box attack by replicating the drone's visual object tracking. However, discrepancies between the surrogate and drone camera views significantly reduce the attack effectiveness. Additionally, the realistic pedestrian motion and the latencies in the attack algorithm impede the attacker's ability to precisely execute the adversarial trajectory. Conversely, the instability of flight operation degrades the accuracy of visual object tracking, thereby increasing the attack success rate. These insights lead us to refine our attack algorithm to better accommodate real-world challenges and exploit systematic vulnerabilities, such as using acoustic signals to deliberately induce high visual object tracking errors. As future work, we aim to advance a comprehensive and systematic security analysis of real-world drone applications by integrating physical attacks with realistic scenarios.

Contributions of this paper include the following:

- We implement an attack hijacking a drone by compromising its follow-me feature leveraging known visual object tracking vulnerabilities.
- We extensively evaluate the attack in simulation considering relevant factors in realistic scenarios. We identify challenges in attack feasibility and opportunities to benefit the adversary.
- Based on the insights from the evaluation, we propose methods to improve the attack success rate from 47% of the baseline attack to 95%.

2 Background and Related Work

Multi-Object Tracking (MOT). MOT algorithms can be classified into two paradigms, tracking-by-detection [2, 6, 7, 15, 43, 47] and joint-detection-and-tracking [41, 44, 48, 51]. Tracking-by-detection splits the visual object tracking into two separate stages, object detection and association. It takes advantage of the modern object detectors and focuses on methodologies to improve association between detection results and IDs. In each frame, the algorithm uses an object detector to locate every object in the view. Kalman-Filter [19] is used to predict the position of each ID under tracking based on historical motion. Then, a bipartite graph matching based on Intersection-over-Union (IoU) between the detected and predicted bounding boxes is performed using Hungarian algorithm [20]. To benefit from appearance features, several works [2, 15, 43] include the distance in Re-Identification (ReID) feature space encoded with deep learning models in the graph weights. Tracking-by-detection has seen many success in autonomous systems [4, 5, 42], while joint-detection-and-tracking has limited deployment and is difficult to succeed in complex scenarios such as UAV tracking [14].

UAV Visual Tracking Systems. Visual tracking has gained significant traction in UAV systems due to its versatility and effectiveness [9, 14, 23, 27, 40]. UAV tracking systems typically implement either image-based visual servoing (IBVS) [9, 27, 29, 33], which directly uses image features for control, or position-based visual servoing (PBVS) [11, 21, 24, 45], which relies on reconstructed 3D poses. Our work adopts IBVS due to its simplicity and broader applicability across diverse UAV platforms with system heterogeneity.

Our research focuses on MOT as the foundational algorithm for UAV tracking systems. The tracking-by-detection paradigm used in state-of-the-art MOT algorithms [6, 15, 47] offers two unique advantages to UAV applications: (1) modular detection model substitution to accommodate varying computational constraints, with association typically handled by efficient Kalman-Filter [19] and Hungarian algorithm [20] operations (achieving up to 260 FPS [6]); and (2) proven effectiveness in obstacle avoidance under complex visual environments, as demonstrated in autonomous vehicles [4, 5, 42] and drones (e.g., DJI's ActiveTrack 5.0 [12]). While most commercial UAV systems are proprietary, substantial evidence indicates that MOT-based approaches are becoming integral to modern UAV visual tracking architectures.

Existing Attacks. Several attacks against MOT algorithms are proposed in both digital [22, 36, 46, 50] and physical space [18, 25, 26, 30, 39], as listed in Table 1. Digital attacks typically assume direct perturbation to the image captured by the victim system, which is very difficult to succeed in a realistic threat model. Jia et al. [18, 25] generate an adversarial patch that drifts the detection box to a desired direction, causing incorrect motion estimation. Following that, ControlLoc [26] has taken realistic challenges such as optimal patch location and robustness into account, generating patches that can achieve false tracking motion in real-world experiments. However, the obvious large patch would limit practicality. Wang et al. [39] focus on attacking the association stage of the MOT algorithms, using the motion of the adversary. The attack targets intentional ID switch between the attacker and victim, using physical adversarial trajectory and abrupt motion change. However, their attack assumes an accurate real-time 3D pose estimation of the victim and the victim camera, which is difficult to achieve in practice. In our preliminary reproduction of the attack, a small 0.5m bias in 3D estimation degrades the attack success rate significantly from 76% to 46%. Adapting existing MOT attacks to UAV platforms introduces additional challenges not encountered in prior work assuming stable camera systems. For instance, the movement of the drone's camera poses significant challenges to the robustness of attack algorithms.

3 Threat Model

Follow-me System. We examine a realistic UAV tracking scenario in which a victim drone d operates in follow-me mode, adjusting its position to track a designated target with position $p^t_{vic} \in \mathbb{R}^3$ (e.g., a pedestrian or a ground vehicle) while maintaining a predefined altitude h_d. Let $p^t_d \in \mathbb{R}^3$ represent the UAV position and $\Omega^t_d \in \mathbb{R}^3$ its orientation (roll, pitch, yaw) at time t. The system comprises a general multirotor drone equipped with a gimbal-mounted camera offering orientation control $\Omega^t_g \in \mathbb{R}^3$. Upon processing each incoming image frame I^t at time t, the MOT algorithm identifies the current position of the target's bounding box $b^t = [u^t, v^t, w^t, h^t]^T \in \mathbb{R}^4$, where (u^t, v^t) represents the box center and (w^t, h^t) its dimensions. The pixel displacement vector $e^t = [u^t - u_c, v^t - v_c]^T \in \mathbb{R}^2$ between the target's bounding box center and the camera's field of view center $c^{FOV}_g = [u_c, v_c]^T$ serves as input to calibrated PID controllers, governing the coordinated movements of both the UAV and its gimbal to minimize tracking error. Our implementation of the UAV tracking system aligns with the architecture described by Mueller et al. [29]. We use SORT [6] as the representative MOT algorithm.

Attacker's Goal. The adversary's primary objective is to execute a hijacking attack against the follow-me functionality and induce erroneous tracking controls, by introducing a malicious object with state $p^t_{adv} \in \mathbb{R}^3$ that follows an optimized trajectory $\mathcal{T} = \{p^0_{adv}, p^1_{adv}, ..., p^t_{adv}\}$, where $t < T_{max}$, with T_{max} denotes the maximum attack duration. The objective is to maximize the probability that the MOT algorithm misidentifies the adversarial object as the legitimate victim target. This motion-based hijacking approach demonstrates practicality and stealth while potentially resulting in severe consequences, including complete loss of the vehicle.

Attack Assumptions. We operate under a gray-box assumption, in which the attacker knows the MOT algorithm $\mathcal{M}$ deployed in the target vehicle without knowing the parameters of the object detector. The attacker employs 3D pose estimation techniques [8, 13, 32] to obtain the poses of both the victim and the UAV. The orientation of the camera can be approximated by centering the victim. These estimations enable the attacker to calculate the camera calibration matrices, project the relevant objects into the UAV's 2D view, and simulate $\mathcal{M}$ locally to predict the tracking behavior. After trajectory optimization, it is executed through augmented reality guidance [39] or a controlled electric unicycle.

Design Challenges. We aim to incorporate several significant refinements on our attack assumptions over previous work [26, 30, 39] , considering realistic operational constraints. The challenges are characterized with results in Section 6.

- *Attacker's knowledge*. With the UAV and its gimbal camera undergoing complex pose variation throughout the visual follow-me process, it is extremely difficult to obtain $p_d(t)$, $\Omega_d(t)$, and $\Omega_g(t)$ in real time. The adversary must position a surrogate camera at a strategically selected location to generate the adversarial trajectory.
- *Latency of execution*. There exists an inherent latency Δt between attack optimization and execution, attributable to various real-world factors including data processing time and the attacker's physical response capabilities. The adversarial position $p_{adv}(t + \Delta t)$ must be generated with information up to I^t.
- *Motion of the subject*. The victim's position change $p^{t+1}_{vic} - p^t_{vic}$ does not follow a constant velocity or direction. Rather, it responds to the attacker's motion, with the response dependent on the attacker's position and velocity vectors.
- *Stealthiness*. The stealthiness of the attack is significantly affected by the minimum distance between the victim and the attacker, and the time period required to succeed.

We aim to answer the following research questions:

RQ1. What are the new factors limiting the success of the existing physical attack against the MOT algorithm in a realistic UAV tracking system context?

RQ2. To what extent does each factor impact the attack success and how can we improve the attack given the insight?

Algorithm 1 UAV-Hijack Attack Generator

```
Input: UAV state $(p^t_d, \Omega^t_d, \Omega^t_g)$, Attacker position $p^t_a$, Victim position $p^t_v$, KF trackers $\mathrm{KF}^t_v$, $\mathrm{KF}^t_a$, iterations $iter$
Output: Optimized adversarial motion $v^{t+1}_a$
function UAV-HIJACK($p^t_d, \Omega^t_d, \Omega^t_g, p^t_a, p^t_v, \mathrm{KF}^t_v, \mathrm{KF}^t_a$, $iter$)
    $P_c = \texttt{ProjectionMatrix}(p^t_d, \Omega^t_d, \Omega^t_g)$
    $D^t_a = \texttt{Project}(P_c, p^t_a), D^t_v = \texttt{Project}(P_c, p^t_v)$
    Initialize $p^{t+1}_a = p^t_a$, $i = 0$
    while $i < iter$ do
        $\mathrm{KF}^{t+1}_v, \mathrm{KF}^{t+1}_a = \mathrm{KF}^t_v\texttt{.update}(D^t_v), \mathrm{KF}^t_a\texttt{.update}(D^t_a)$
        $\hat{b}^{t+1}_v, \hat{b}^{t+1}_a = \mathrm{KF}^{t+1}_v\texttt{.pred()}, \mathrm{KF}^{t+1}_a\texttt{.pred()}$
        $\mathrm{KF}^{t+2}_v, \mathrm{KF}^{t+2}_a = \mathrm{KF}^{t+1}_v\texttt{.update}(\hat{b}^{t+1}_v), \mathrm{KF}^{t+1}_a\texttt{.update}(\hat{b}^{t+1}_a)$
        $\hat{b}^{t+2}_v, \hat{b}^{t+2}_a = \mathrm{KF}^{t+2}_v\texttt{.pred()}, \mathrm{KF}^{t+2}_a\texttt{.pred()}$
        $D^{t+1}_a = \texttt{Project}(P_c, p^{t+1}_a)$
        $L_{surr} = \texttt{ComputeLoss}(D^{t+1}_a, \hat{b}^{t+1}_v, \hat{b}^{t+2}_v, \hat{b}^{t+2}_a)$
        $p^{t+1}_a = \texttt{GradientUpdate}(p^{t+1}_a, \nabla_{p_a} L_{surr})$
        $D^t_a = \texttt{Project}(P_c, p^{t+1}_a)$
        $i = i + 1$
    end while
     return $p^{t+1}_a$
end function
while $t < T_{max}$ and not hijacked do
    Estimate UAV state $(p^t_d, \Omega^t_d, \Omega^t_g)$ and victim position $p^t_v$
    $p^{t+1}_a$ = UAV-Hijack($p^t_d, \Omega^t_d, \Omega^t_g, p^t_a, p^t_v, \mathrm{KF}^t_v, \mathrm{KF}^t_a, iter$)
    $v^{t+1}_a = \texttt{SmoothedVelocity}(p^t_a, p^{t+1}_a)$
    Execute movement according to $v_a$
    $t = t + 1$
end while
```

4 Attack Algorithm

In the association stage of object tracking, Hungarian algorithm determines the ID assignments. This algorithm maximizes the total Intersection-over-Union (IoU) by optimally matching pairs of: (1) detection bounding boxes D_{ID}^t, which depend on the object's current state p_{ID}^t, and (2) Kalman-Filter KF_{ID}^t predicted bounding boxes $\hat{b}_{ID}^t$, which are based on the object's historical states p_{ID}^z, $z = 1, 2..., t-1$. Therefore, the optimized states $\mathcal{T}$ will determine the attacker's detection results, as well as the KF motion prediction results by injecting adversarial motion. The hijacking attack is successful if

$$\text{IoU1} > \text{IoU2} \tag{1}$$

where $\text{IoU1} = \text{IoU}(D_a^{t+1}, \hat{b}_v^{t+1}) + \text{IoU}(D_v^{t+1}, \hat{b}_a^{t+1})$ and $\text{IoU2} = \text{IoU}(D_a^{t+1}, \hat{b}_a^{t+1}) + \text{IoU}(D_v^{t+1}, \hat{b}_v^{t+1})$, where a denotes the attacker's ID and v denotes the victim's ID. Essentially, the detected box needs to have a larger IoU with the incorrect KF predicted box to incur a switch.

Formally, given a set of detection bounding boxes $D^t = \{D_1^t, D_2^t, ..., D_n^t\}$ and Kalman-Filter predicted bounding boxes $\hat{b}^t = \{\hat{b}_1^t, \hat{b}_2^t, ..., \hat{b}_m^t\}$ at time t, our attack aims at optimizing the following objective function

$$\max_{p_{adv}^{t+1}} [\text{IoU}(D_a^{t+1}, \hat{b}_v^{t+1}) + \text{IoU}(D_v^{t+1}, \hat{b}_a^{t+1})] \tag{2}$$

Maximizing this objective function increases the likelihood of an ID switch between the attacker and victim. Since we do not have access to D_v^{t+1} and it is hard to simultaneously optimize both D_a^{t+1} and $\hat{b}_a^{t+1}$, which depend on current and past time stamps, we employ a surrogate objective function:

$$\max_{p_{adv}^{t+1}} L_{surr} = \text{IoU}(D_a^{t+1}, \hat{b}_v^{t+1}) + \text{IoU}(\hat{b}_v^{t+2}, \hat{b}_a^{t+2}) \tag{3}$$

This surrogate formulation approximates the victim's future detection D_v^{t+1} with its Kalman-Filter prediction $\hat{b}_v^{t+2}$ at time $t+2$, allowing us to optimize the adversarial trajectory p_{adv}^{t+1} with information available at time t. Our algorithm is enhanced over existing attack [39], by adding the UAV pose, 3D to gimbal camera projection, and optimized result to realistic motion convertion, as demonstrated by Algorithm 1. Essentially, the hijacking attack exploits the vulnerability in the MOT algorithm, while addressing the systematic challenges posed by a dynamic UAV follow-me system.

5 Experiment Methodology

Testbed. We construct a testbed using the Gazebo simulator, PX4 Autopilot, and ROS2, as illustrated in Figure 2. The experiments are carried out on a desktop with Intel i9-14900K CPU and RTX 4080 GPU running Ubuntu 24.04. Our simulation camera operates at 30 FPS with 1920×1080 resolution. For each experimental configuration, we execute 100 trials with the attacker and victim spawned within a 25×25 meter area, initial velocities $v_{init} \in [0.5, 2.0]$ m/s, and the UAV at a fixed position with a clear view of the target.

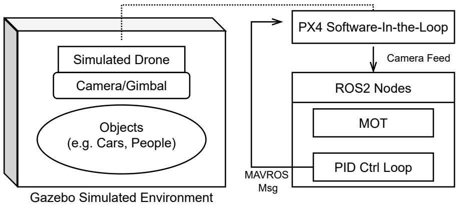

Figure 2: Illustration of our testbed design.

Metric. We evaluate attack effectiveness using the attack success rate (ASR)—the ratio of successful trials to total trials. An attack is considered successful if it triggers ID hijacking in the MOT algorithm and maintains this hijacked state within 500 frames (15 seconds). We also show the average IoU1 and IoU2 across all trials in each setting.

Scenarios Setup. To model realistic constraints,

- *Physical constraints*: Enforcing a minimum distance $d_{min} \in \{0.5, 1.0, 2.0\}$ meters between attacker and victim.
- *Execution latency*: Simulating delay between optimization and execution by injecting a time delay in number of frames until the optimized motion takes place.
- *Victim motion*: Modeling using the Extended Social Force Model (ESFM) [1, 28]. Victim response level 0 represents fixed velocity in magnitude and direction once initialized. Victim response level 1 and level 2 cause the victim to vary its velocity at maximum 1.0 and 9.0 m/s^2 acceleration, respectively. Victim velocity is initialized randomly in $[0.5, 2.0]$ m/s and capped at 4 m/s.
- *UAV instability*: We consider four realistic settings for wind simulation. Level 0: no wind. Level 1: light: 0-5mph, ±20° directional variation. Level 2: moderate: 5-10mph, ±60° directional variation. Level 3: strong/gusty: 10-15mph, ±100° directional variation, ±50% speed fluctuation.
- *Surrogate camera*: Positioned with viewpoint perpendicular to UAV's initial view. Continuous visibility of both the attacker and the victim is ensured throughout the trials.

6 Evaluation

Baselines. We first reproduce the baseline static camera attack from Wang et. al. [39] in the ideal simulation environment, achieving almost 100% attack success rate (ASR). Then we enable the realistic victim motion and set the minimum distance between the victim and the attacker to 1.0 m, which drops ASR to 52%, as shown in the row of "Static Camera" in Table 2. This is caused by our realistic setup: (1) we enforce a larger minimum distance (2) we model the victim's motion response, which typically drives the victim away from the approaching attacker, causing the attack harder to succeed.

With realistic victim motion, we replace the static camera by a camera on a flying drone (the row of "Follow-me" in Table 2). The hijacking attack achieves 47% success rate, slightly lower than the static camera scenario. This further confirms that the challenges are mainly brought about by our realistic considerations, concluding to Finding 1.

Condition		ASR	IoU1	IoU2
Static Camera [39]		0.52	1.10	1.43
Follow-me*		0.47	1.12	1.30
Min Dist. (m)	0.5	0.76	1.18	1.31
	1.0	*		
	2.0	0.26	0.83	1.33
Vic. Resp.	0	0.72	1.34	1.40
	1	*		
	2	0.54	1.12	1.31
Wind Level	**0**	*		
	1	0.53	1.06	1.29
	2	0.69	1.02	1.19
	3	0.86	1.07	1.04

Table 2: Evaluation metrics under various conditions. The * represents our baseline where the UAV is operating in follow-me mode and have other parameters set as the bold values.

Finding 1: The realistic pedestrian motion causes a larger attacker-victim minimum distance, thereby degrading the attack success rate significantly. **(RQ1)**

The impact of varying degrees of real-world factors. We tune the parameters of real-world factors as discussed in Section 5 to see their impact on attack effectiveness. Demonstrated by Table 2, the physical constraint on the minimum distance drastically impacts the attack success rate, with over 20% ASR drop in every distance doubling. This is because the attack success relies on IoU1 > IoU2 (Equation 1), where IoU1 is determined by overlap between the attacker's detection box and the victim's predicted box, as well as overlap between victim's detection box and the attack's predicted box. When there is a larger physical separation, the projected bounding box also has a larger image pixel distance, causing it harder to achieve sufficient cross IoU. We can also observe the steady decrease in the expected maximum IoU1 with increasing minimum distance.

Similarly, the realistic victim response also presents a significant challenge in maximizing IoU1, causing the motion attack to fail. Table 2 shows that with the victim response, the ASR drops from 72% to 54%. Since the victim motion model responds to the position proximity and approaching velocity of other pedestrians, the victim will actively move away from the attacker. In addition, we find that since pedestrians have limited speed and acceleration capability, increasing victim motion uncertainty has limited impact on ASR. The average velocity changes of the victim within a 0.5s time intervel are 18% and 28% with victim response levels 1 and 2, and the MOT algorithm can maintain a stable tracking on the victim, also proven by stable IoU2 values. Other tracking scenarios with more complex physical motion capability, such as ground or areial vehicles, might make this issue more severe. The results further validate the importance of Finding 1.

Conversely, we discover that the wind, which brings uncertainty to the UAV operation, has a positive impact on the hijacking ASR. Different from the uncertainty in the victim response motion, the wind affects ASR through impacting IoU2. In Table 2, the expected average IoU2 drops sharply as the wind becomes more severe. Essentially, the uncertainty in the wind results in motion in the gimbal camera, causing the 2D position of the bounding box to shift (details refer to Appendix A). As a result, the MOT algorithm predicts the victim's bounding box with larger displacements, thus allowing the attack to succeed with less optimized IoU1 through adversarial motions. On the other hand, it also relaxes the physical distance constraint, since the attack can achieve the same level of IoU1 at a longer distance. We also plot the displacement of the victim's bounding box in success and failure situations in all experiment settings in Figure 3. It shows that successful attacks typically have a higher displacement in victim bounding box center. The result concludes to Finding 2.

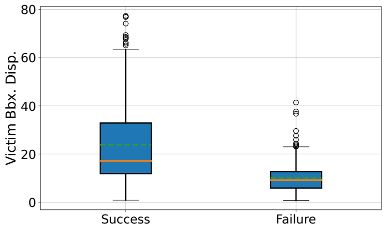

Figure 3: Center displacement of victim Kalman-Filter predicted bounding box from the ground truth bounding box

Finding 2: The larger MOT prediction error (i.e. victim bounding box displacement) contributes to the attack success. Thus, uncertain motion in the UAV and the gimbal camera assist in the attack. **(RQ2)**

Attack with realistic threat model. In addition to real-world physical constraints and uncertainty, we investigate the attack performance under realistic attacker capability, in terms of execution latencies, inaccurate pose estimation of the victim, and limited attacker's knowledge on the drone's view (as discussed in Section 3). We adopt the three realistic settings separately on the "Follow-me" baseline (Table 2) and show results in Table 3. With a 30-frame (about 1s) latency in the attacker's capability to adapt to the optimized motion, the ASR drops by 14%. This is because by the time the attacker reaches the optimal attack position, the victim has moved to the next position. Our result shows, when the attacker arrives at the optimized position at a delay, compared to without delay, a 0.07 drop in IoU1, together with an averaged 38.3 pixel error between the previous and current victim's bounding box.

Obtaining the accurate 3D pose of both the victim and the victim vehicle in real time is very challenging. Table 3, with the 1.0m minimum distance setting, shows that a 0.5m bias in 3D pose estimation reduces the ASR by approximately half. This is because the 2D projection box of the victim is also inaccurate, so optimizing our function towards an inaccurate victim position produces a less effective adversarial motion. In this case, we observe 10-40 pixels error in victim's 2D bounding box depending on the current viewing point of the gimbal camera.

Then, we test a practical threat model where the attacker relies on its surrogate attack camera installed at a static po-

Condition	ASR
1s Latency	0.33
0.5m 3D Pose Err.	0.26
Surrogate Camera	0.14

Table 3: ASR with realistic attacker's capability.

Movement	ASR
3.0°	0.74
6.0°	**0.95**

Table 4: ASR with adversarial gimbal movement.

sition. The preliminary result shows that the ASR is very limited, due to the fact that the same 3D location will map to very different 2D position viewing from different points. For example, two cameras looking in the opposite direction will see a mirrored image. Since the UAV undergo constant view point change during follow-me, the ASR drops significantly. We will need to enhance the attack to strategically place the surrogate camera or utilize multiple camera views to generate a more robust adversarial motion.

In conclusion, the attacker's capability is limited in the real-world scenarios, which severely challenges attack effectiveness and robustness, as said in Finding 3.

Finding 3: Realistic latency and less accurate 3D estimation negatively impact attack success. A surrogate camera at a non-optimal location has the most detrimental effect. An enhanced attack should compensate latency and strategically place the surrgate camera. **(RQ1 & RQ2)**

Gimbal Attack. The rise in ASR due to camera instability caused by wind suggests that an adversary could enhance the effectiveness of their attack by inducing movement in the target camera. An possible method for this is for an attacker to use acoustic waves to inject movement exploiting camera stabilization mechanisms [17]. The use of gimbal mechanisms to stabilize camera feeds in UAV applications is commonplace and presents an opportunity for an attacker. To simulate such an attack, we generate a control signal composed of sine waves, and use this signal to offset UAV gimbal position in the yaw and pitch axes while the attacker follows the generated trajectory of the hijacking attack. To determine the effect of signal strength, we test the signal to cause at maximum 3.0 and 6.0 degrees of change in the gimbal camera's orientation. Our preliminary acoustic experiments against DJI Zenmuse X4S gimbal have achieved over 10°orientation changes. As shown in Table 4, injecting this noise leads to a sharp increase in ASR. At the highest level tested, we achieve an absolute 48% increase in ASR compared to the baseline and outperform all the results found in previous experiments. Besides increasing the ASR, further analysis shows that our attack also minimizes the attack duration. More details are in Appendix A.

7 Discussion and Future Work

Generalizability. Enhanced MOT algorithms typically incorporate additional processing steps or appearance metrics. Our hijacking attack exploits vulnerabilities in the fundamental IoU matching mechanism used across many MOT algorithms, making it inherently generalizable. We evaluated the attack transferability against StrongSORT [15], a recent advanced MOT algorithm. Using settings ("Follow-me" in Table 2 and "6.0°" in Table 4), we achieved ASR of 39% and 72% respectively. This shows that while StrongSORT achieves more resilient tracking, our proposed gimbal perturbation enhanced attack can still boost the ASR by over 30%.

Mitigation. One approach involves equipping the UAV with 3D perception, enabling the tracking algorithm to use depth information during ID association, necessitating additional hardware (e.g., LiDAR or depth cameras). Alternatively, algorithm-level countermeasures can enhance ID association with appearance features and relative pose information [23], which characterizes objects by their spatial relationships to two other objects (the furthest and closest), as these relative distances and angles remain stable during gimbal perturbations. To maintain tracking efficiency in benign cases, an adaptive approach can be implemented, where the gimbal's inertial measurements are monitored, activating the additional metrics only when anomalous acceleration patterns are detected.

Limitations. First, our attack is validated in simulation using widely adopted UAV frameworks but lacks physical-world testing with real-world uncertainties. Second, attack success depends heavily on precise 3D pose estimation of both the victim target and drone camera, requiring strategic positioning of surrogate cameras in black-box scenarios to approximate the victim drone's perspective. Finally, practical implementation must address latency by generating adversarial motions preemptively, potentially using trajectory prediction to anticipate future movements of both the victim person and the drone.

Exploring Camera Instability. For more insight and controlled attack, we consider modeling the gimbal motion achieved by acoustic signals, varying directions and distances.

8 Conclusion

We perform a thorough security analysis on the UAV tracking system in follow-me mode, leveraging known motion attack against visual object tracker. We consider real-world challenges in carrying out a successful hijacking attack, including physical constraints, real-world uncertainties, and limited attacker's knowledge. We implement the hijacking attack in Gazebo and study the detailed challenges brought by realistic tracking scenarios. With insights from the analysis, we propose an enhanced hijacking attack by injecting noise into the UAV gimbal, which increases the attack success rate to 95%.

9 Acknowledgement

This work was partially supported by Boeing as well as NSF under grants CMMI-2038215, CNS-2321532, and National AI Institute for Edge Computing Leveraging Next Generation Wireless Networks, Grant # 2112562.

References

[1] https://github.com/yuxiang-gao/PySocialForce, 2020. Accessed: 2025-02-27.

[2] Nir Aharon, Roy Orfaig, and Ben-Zion Bobrovsky. Bot-sort: Robust associations multi-pedestrian tracking. *arXiv preprint arXiv:2206.14651*, 2022.

[3] Autel. Autel evo ii drone dynamic track mode full review. https://www.autelpilot.com/blogs/buying-guides/autel-evo-ii-drone-dynamic-tracking-mode, 2020. Accessed: 2025-02-25.

[4] Autoware. https://github.com/autowarefoundation/autoware, 2022. Accessed: 2025-02-25.

[5] Baidu. https://github.com/ApolloAuto/apollo, 2017. Accessed: 2025-02-25.

[6] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *2016 IEEE international conference on image processing (ICIP)*, pages 3464–3468. Ieee, 2016.

[7] Jinkun Cao, Jiangmiao Pang, Xinshuo Weng, Rawal Khirodkar, and Kris Kitani. Observation-centric sort: Rethinking sort for robust multi-object tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9686–9696, 2023.

[8] Adrian Carrio, Sai Vemprala, Andres Ripoll, Srikanth Saripalli, and Pascual Campoy. Drone detection using depth maps. In *2018 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 1034–1037. IEEE, 2018.

[9] Haolin Chen, Ruidong Wu, Wenshuai Lu, Xinglong Ji, Tao Wang, Haolun Ding, Yuxiang Dai, and Bing Liu. Fully onboard single pedestrian tracking on nano-uav platform. *Journal of Intelligent & Robotic Systems*, 109(3):50, 2023.

[10] Drew Davidson, Hao Wu, Rob Jellinek, Vikas Singh, and Thomas Ristenpart. Controlling {UAVs} with sensor input spoofing attacks. In *10th USENIX workshop on offensive technologies (WOOT 16)*, 2016.

[11] Floris De Smedt, Dries Hulens, and Toon Goedemé. Onboard real-time tracking of pedestrians on a uav. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 1–8, 2015.

[12] DJI. Best drones that follow you automatically (2024). https://store.dji.com/content/camera-drone-that-follows-you, 2024. Accessed: 2025-02-25.

[13] Sedat Dogru and Lino Marques. Drone detection using sparse lidar measurements. *IEEE Robotics and Automation Letters*, 7(2):3062–3069, 2022.

[14] Yunhao Du, Junfeng Wan, Yanyun Zhao, Binyu Zhang, Zhihang Tong, and Junhao Dong. Giaotracker: A comprehensive framework for mcmot with global information and optimizing strategies in visdrone 2021. In *Proceedings of the IEEE/CVF International conference on computer vision*, pages 2809–2819, 2021.

[15] Yunhao Du, Zhicheng Zhao, Yang Song, Yanyun Zhao, Fei Su, Tao Gong, and Hongying Meng. Strongsort: Make deepsort great again. *IEEE Transactions on Multimedia*, 25:8725–8737, 2023.

[16] Chong Huang, Zhenyu Yang, Yan Kong, Peng Chen, Xin Yang, and Kwang-Ting Tim Cheng. Learning to capture a film-look video with a camera drone. In *2019 international conference on robotics and automation (ICRA)*, pages 1871–1877. IEEE, 2019.

[17] Xiaoyu Ji, Yushi Cheng, Yuepeng Zhang, Kai Wang, Chen Yan, Wenyuan Xu, and Kevin Fu. Poltergeist: Acoustic adversarial machine learning against cameras and computer vision. In *2021 IEEE Symposium on Security and Privacy (SP)*, 2021.

[18] Yunhan Jia Jia, Yantao Lu, Junjie Shen, Qi Alfred Chen, Hao Chen, Zhenyu Zhong, and Tao Wei Wei. Fooling detection alone is not enough: Adversarial attack against multiple object tracking. In *International Conference on Learning Representations (ICLR'20)*, 2020.

[19] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. 1960.

[20] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[21] Rui Li, Minjian Pang, Cong Zhao, Guyue Zhou, and Lu Fang. Monocular long-term target following on uavs. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 29–37, 2016.

[22] Delv Lin, Qi Chen, Chengyu Zhou, and Kun He. Tracklet-switch adversarial attack against pedestrian multi-object tracking trackers. *arXiv preprint arXiv:2111.08954*, 2021.

[23] Shuai Liu, Xin Li, Huchuan Lu, and You He. Multi-object tracking meets moving uav. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8876–8885, 2022.

[24] Yuzhen Liu, Ziyang Meng, Yao Zou, and Ming Cao. Visual object tracking and servoing control of a nanoscale quadrotor: System, algorithms, and experiments. *IEEE CAA J. Autom. Sinica*, 8(2):344–360, 2021.

[25] Chen Ma, Ningfei Wang, Qi Alfred Chen, and Chao Shen. Wip: Towards the practicality of the adversarial attack on object tracking in autonomous driving. In *ISOC Symposium on Vehicle Security and Privacy (VehicleSec)*, 2023.

[26] Chen Ma, Ningfei Wang, Zhengyu Zhao, Qian Wang, Qi Alfred Chen, and Chao Shen. Controlloc: Physical-world hijacking attack on visual perception in autonomous driving. *arXiv preprint arXiv:2406.05810*, 2024.

[27] Alaa Maalouf, Ninad Jadhav, Krishna Murthy Jatavallabhula, Makram Chahine, Daniel M Vogt, Robert J Wood, Antonio Torralba, and Daniela Rus. Follow anything: Open-set detection, tracking, and following in real-time. *IEEE Robotics and Automation Letters*, 9(4):3283–3290, 2024.

[28] Mehdi Moussaïd, Niriaska Perozo, Simon Garnier, Dirk Helbing, and Guy Theraulaz. The walking behaviour of pedestrian social groups and its impact on crowd dynamics. *PloS one*, 5(4):e10047, 2010.

[29] Matthias Mueller, Gopal Sharma, Neil Smith, and Bernard Ghanem. Persistent aerial tracking system for uavs. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1562–1569. IEEE, 2016.

[30] Raymond Muller, Yanmao Man, Z Berkay Celik, Ming Li, and Ryan Gerdes. Physical hijacking attacks against object trackers. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pages 2309–2322, 2022.

[31] Neng Pan, Ruibin Zhang, Tiankai Yang, Can Cui, Chao Xu, and Fei Gao. Fast-tracker 2.0: Improving autonomy of aerial tracking with active vision and human location regression. *IET Cyber-Systems and Robotics*, 3(4):292–301, 2021.

[32] Seongjoon Park, Hyeong Tae Kim, Sangmin Lee, Hyeontae Joo, and Hwangnam Kim. Survey on anti-drone systems: Components, designs, and challenges. *IEEE access*, 9:42635–42659, 2021.

[33] Jesus Pestana, Jose Luis Sanchez-Lopez, Srikanth Saripalli, and Pascual Campoy. Computer vision based general object following for gps-denied multirotor unmanned vehicles. In *2014 American Control Conference*, pages 1886–1891. IEEE, 2014.

[34] Harshad Sathaye, Martin Strohmeier, Vincent Lenders, and Aanjhan Ranganathan. An experimental study of {GPS} spoofing and takeover attacks on {UAVs}. In *31st USENIX security symposium (USENIX security 22)*, pages 3503–3520, 2022.

[35] Nina Shamsi, Kaeshav Chandrasekar, Yan Long, Christopher Limbach, Keith Rebello, and Kevin Fu. Wip: Threat modeling laser-induced acoustic interference in computer vision-assisted vehicles.

[36] Woojin Shin, Donghwa Kang, Daejin Choi, Brent Kang, Jinkyu Lee, and Hyeongboo Baek. Banktweak: Adversarial attack against multi-object trackers by manipulating feature banks. *arXiv preprint arXiv:2408.12727*, 2024.

[37] Skydio. The best follow me drone in 2022. https://www.skydio.com/blog/10-reasons-skydio-makes-the-best-follow-me-drone, 2022. Accessed: 2025-02-25.

[38] Yunmok Son, Hocheol Shin, Dongkwan Kim, Youngseok Park, Juhwan Noh, Kibum Choi, Jungwoo Choi, and Yongdae Kim. Rocking drones with intentional sound noise on gyroscopic sensors. In *24th USENIX security symposium (USENIX Security 15)*, pages 881–896, 2015.

[39] Chenyi Wang, Yanmao Man, Raymond Muller, Ming Li, Z Berkay Celik, Ryan Gerdes, and Jonathan Petit. Physical id-transfer attacks against multi-object tracking via adversarial trajectory.

[40] Shuaijun Wang, Fan Jiang, Bin Zhang, Rui Ma, and Qi Hao. Development of uav-based target tracking and recognition systems. *IEEE Transactions on Intelligent Transportation Systems*, 21(8):3409–3422, 2019.

[41] Zhongdao Wang, Liang Zheng, Yixuan Liu, Yali Li, and Shengjin Wang. Towards real-time multi-object tracking. In *European conference on computer vision*, pages 107–122. Springer, 2020.

[42] Waymo. https://waymo.com/, 2017. Accessed: 2025-02-25.

[43] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *2017 IEEE international conference on image processing (ICIP)*, pages 3645–3649. IEEE, 2017.

[44] Fangao Zeng, Bin Dong, Yuang Zhang, Tiancai Wang, Xiangyu Zhang, and Yichen Wei. Motr: End-to-end multiple-object tracking with transformer. In *European conference on computer vision*, pages 659–675. Springer, 2022.

[45] Haotian Zhang, Gaoang Wang, Zhichao Lei, and Jenq-Neng Hwang. Eye in the sky: Drone-based object tracking and 3d localization. In *Proceedings of the 27th ACM international conference on multimedia*, pages 899–907, 2019.

[46] Weilong Zhang, Fang Yang, and Zhao Guan. Dual-dimensional adversarial attacks: A novel spatial and temporal attack strategy for multi-object tracking. In *2024 International Joint Conference on Neural Networks (IJCNN)*, pages 1–10. IEEE, 2024.

[47] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. In *European conference on computer vision*, pages 1–21. Springer, 2022.

[48] Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng, and Wenyu Liu. Fairmot: On the fairness of detection and re-identification in multiple object tracking. *International journal of computer vision*, 129:3069–3087, 2021.

[49] C Zhou, Q Yan, Y Shi, et al. Doublestar: Long-range attack towards depth estimation based obstacle avoidance in autonomous systems. arxiv preprint arxiv: 211003154. 2021.

[50] Tao Zhou, Qi Ye, Wenhan Luo, Kaihao Zhang, Zhiguo Shi, and Jiming Chen. F&f attack: Adversarial attack against multiple object trackers by inducing false negatives and false positives. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4573–4583, 2023.

[51] Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. Tracking objects as points. In *European conference on computer vision*, pages 474–490. Springer, 2020.

A Gimbal Attack

A.1 Demonstration of Gimbal Attack

We show the effect of our gimbal attack in achieving the follow-me hijacking goal with Figures 4 and 5. In the figures, we use the green and blue boxes to indicate the object detection results of the victim and the attacker, respectively. The yellow and light blue boxes represent the MOT predicted bounding box based on the Kalman-Filter. So in order for the attack to succeed, we need to make the sum of IoU between the yellow and blue box and IoU between the light blue and green box, larger than the sum of IoU between the yellow and green box and IoU between the light blue and blue box. Figure 4 shows the tracking results before the injected gimbal noise occurs. The predicted bounding box (yellow and light blue) based on motion filter performs well and has a larger overlap with the correct underlying objects. Therefore, the victim and attacker's original ID is preserved. However, in the attack success frame, the adversarial motion injected into the gimbal camera causes the position of the detected 2D bounding boxes (green and blue) to have a larger displacement. The displacement shifts the attacker's detected bounding box (blue) towards the victim's predicted bounding box (yellow) and has a large IoU. As a result, the victim's ID will be taken over by the attacker. At the follow-me application level, a hijacking attack is achieved.

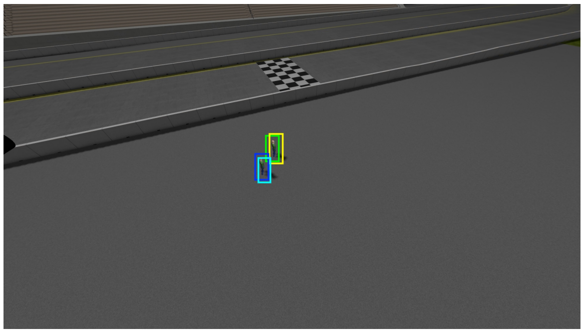

Figure 4: Image at frame $t-5$

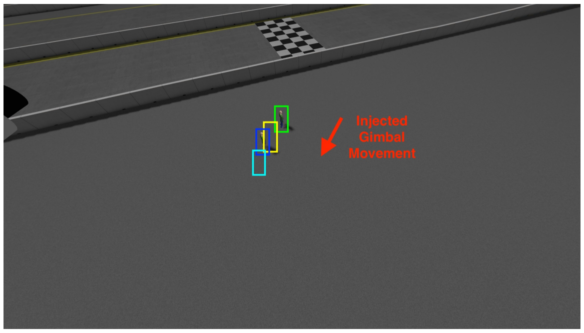

Figure 5: Image at attack success frame t

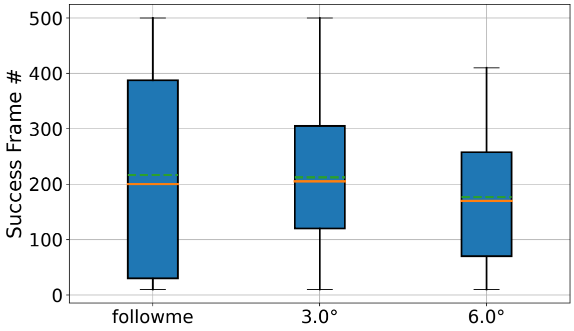

Figure 6: Distribution of attack success frame during the attack execution period

A.2 Shortened Attack Period

In Figure 6, we show that by injecting artificial noise into the UAV's gimbal camera, we also improve the attack effectiveness in terms of shortening and stabilizing the attack duration. Compared to the Follow-me baseline, injecting 3.0° and 6.0° noise into the gimbal motion makes the attack success frame more concentrated. Meanwhile, with the largest 6.0° injected motion, the attack in average takes 30 frames less to succeed.