

GPS Spoofing Attacks on AI-based Navigation Systems with Obstacle Avoidance in UAV

Ji Hyuk Jung¹, Mi Yeon Hong¹, Ji Won Yoon¹

¹School of CyberSecurity, Korea University, Anam-ro, Seoul, 02841, Seoul, Korea.

Contributing authors: graycat@korea.ac.kr; hachikohmy@korea.ac.kr; jiwon_yoon@korea.ac.kr;

Abstract

Recently, approaches using Deep Reinforcement Learning (DRL) have been proposed to solve UAV navigation systems in complex and unknown environments. However, despite extensive research and attention, systematic studies on various security aspects have not yet been conducted. Therefore, in this paper, we conduct research on security vulnerabilities in DRL-based navigation systems, particularly focusing on GPS spoofing attacks against the system. Many recent basic DRL-based navigation systems fundamentally share an efficient structure. This paper presents an attack model that operates through GPS spoofing attacks briefly modeling the range of spoofing attack against EKF sensor fusion of PX4 autopilot, and combine this with the DRL-based system to design attack scenarios that are closer to reality. Finally, this paper experimentally demonstrated that attacks are possible both in the basic DRL system and in attack models combining the DRL system with PX4 autopilot system.

Keywords: Deep Reinforcement Learning, Obstacle Avoidance, EKF, Navigation, Sensor Spoofing, PX4, UAV

1 Introduction

Unmanned Aerial Vehicles (UAVs) refer to vehicles that can fly autonomously or be piloted remotely. UAVs are being widely utilized across various domains, particularly for delivery services, surveillance operations, reconnaissance missions in hazardous regions, and agricultural monitoring [1–5]. Additionally, intelligent UAV on the battlefield has become a key point in the defense industry [6, 7]. The development of precise sensors such as GPS and IMU, the advancement of propulsion technologies

like motors and engines, and the evolution of estimation or control software made the birth of UAVs possible. Recently, with the rapid advancement of artificial intelligence and the increasing sophistication of UAV applications, UAVs are now expected to perform far more complex flight maneuvers that go well beyond basic automatic flight control. Especially, UAVs are required to perform autonomous flight in complex environments with numerous obstacles, such as buildings in urban settings and natural terrain features in hazardous areas. Recently, vision-based navigation, which performs navigation functions in complex environments based on depth images, has been gaining attention. Traditional autonomous operation either lacks obstacle avoidance capabilities or requires prior knowledge of the shape and position of surrounding obstacles [8–10]. However, DRL-based navigation systems allow UAVs to operate in complex environments without any prior information. Therefore, DRL-based navigation systems are becoming central to research on autonomous UAV flight. DRL-based navigation system for UAVs primarily utilize depth images, position and direction as fundamental input parameters. DRL-based navigation systems are designed to perform two critical functions: obstacle avoidance and target-oriented navigation. Recent research in the field of DRL-based navigation systems has converged on an efficient architectural paradigm, which typically processes depth images through CNN layers, followed by a concatenation layer where physical state inputs such as position and direction are combined.

Recently, research on attacks and defenses related to adversarial attacks on these DRL systems has been conducted [11]. While this demonstrates growing security concerns about these systems, there has been little research conducted beyond this area so far. Fault injection or sensor spoofing attack on robotic vehicles have been major topics of interest in security research for a long time [12–17]. In [14], a attack model and methodology were proposed to cause malfunctions in ADAS systems through fault injection, assuming the attacker can access internal or external network systems. A fuzzing-based attack was proposed under the assumption that the attacker can input data to all sensor inputs, including image sensors and LiDAR sensors in [13]. [12] demonstrated that an adversarial attack for autonomous vehicle can be conducted in a physical environment. While numerous attacks on sensors and inputs have been conducted against autonomous vehicles, attack modeling for DRL-based navigation systems has not yet been researched. Therefore, from this perspective, this paper aimed to design an attack model for sensor spoofing attacks on DRL-based systems and experimentally prove that such attacks are feasible. This paper proposes GPS spoofing attack scenario that is feasible within DRL-based navigation systems. While research has been conducted on adversarial attacks against depth images in DRL-based systems [11], research on sensor spoofing attacks targeting input values for position and direction have not yet been carried out. Additionally, adversarial attacks based on depth images have limitations regarding their practicality in real-world environments because the high dimensional nature of depth images, characterized by a large number of pixels, necessitates subtle and precise alterations to individual pixel values for effective adversarial attack. Especially, there has been little research on GPS spoofing attack for the DRL-based navigation system. Therefore, this paper analyzes the architecture of the DRL-based system for GPS spoofing attack and systematically

proposes attack methodologies. To consider GPS spoofing detection, this study investigates the feasible range of spoofing attacks from the perspective of UAV autopilot system and restricts the input values which avoid detection by the spoofing detection system in the DRL-based system model. In summary, this paper makes the following contributions.

- This paper proposes threat scenarios and analyses the recently studied DRL-based system for modeling GPS spoofing attacks.
- To perform GPS spoofing attack on recent UAV autopilot controller systems, attackers must evade the spoofing detection mechanisms of the control system, which limits the attacker’s input space. This paper integrates these constraints to present a comprehensive threat attack model.
- The paper experimentally demonstrates that GPS spoofing attack on the DRL-based navigation system are feasible even within this constrained input space.

2 Background

2.1 Deep Reinforcement Learning

Deep Reinforcement Learning (DRL) is an advanced paradigm that integrates deep learning techniques with traditional reinforcement learning frameworks [18]. This synergistic approach enables the utilization of high dimensional data, such as images, as state representations or observations within the reinforcement learning process. The fundamental approach in RL starts with an intelligent agent interacting with a given dynamic environment at each time step. At each time step t , the agent is at the current state s_t decides on an action a_t , and in return, receives its next state s_{t+1} and receives a reward $r(s_t, a_t)$ from the environment. The ultimate goal of reinforcement learning is to maximize the cumulative sum of rewards G , which accumulates until the end of the state sequence. γ is a factor used for discounting future rewards. Equation 1 represents the accumulated discounted rewards from the current time step t . The goal of reinforcement learning is to find a policy that maximizes G while the agent interacts with the environment.

$$G_t = \sum_{t=0}^T \gamma^t r(s_t, a_t). \quad (1)$$

A policy is a function that outputs the action to be taken based on the current state. The following represents the policy function.

$$\pi_{\theta}(s) = a, \quad (2)$$

where θ is the parameter of the policy π , s and a are the state and action of an agent, respectively. Fundamentally, DRL proceeds with learning to optimize the policy function from sequences of states and actions obtained by the agent in the environment. This sequence is termed a trajectory τ , and is represented by the following equation.

$$\tau = (s_0, a_0, s_1, a_1, \dots, s_T, a_T.) \quad (3)$$

Policy-based reinforcement learning aims to maximize the expected value of the cumulative reward G from these obtained trajectories. Therefore, the learning process can be represented as follows.

$$J(\theta) = E_{\tau \sim p_{\theta}(\tau)} \left[\sum_{t=0}^T \gamma^t r(s_t, a_t) \right] \quad (4)$$

$$\theta^* = \arg \max_{\theta} J(\theta)$$

As the target of attack for this paper, we selected a navigation system implemented with Twin Delayed DDPG (TD3), one of the most recent DRL methodologies [19]. TD3 is essentially a DRL algorithm based on Deep Deterministic Policy Gradient (DDPG) [20], but it addresses several of its shortcomings. TD3 is an off-policy actor-critic algorithm which learns a policy and Q function simultaneously. Additionally, TD3 assumes a policy function to be a deterministic. Q function outputs the expected value when a specific action is taken in a given state. For learning Q function, TD3 uses replay buffer technique, which involves randomly extracting states, actions, and rewards from trajectories to facilitate learning [21]. Equation 5 represents the loss function used for learning the Q function.

$$L(\phi) = E_{(s,a,r,s') \sim D} [(Q_{\phi} - (r + \gamma Q_{\phi}(s', \pi_{\theta}(s'))))^2]. \quad (5)$$

Modeling Q function and assuming a deterministic policy function simplify the calculation of the gradient of the objective function $J(\theta)$.

$$\nabla_{\theta} J(\theta) = \nabla_{\theta} \sum_{k=0}^N Q_{\phi}(s_k, \pi_{\theta}(s_k)) \quad (6)$$

$$\begin{aligned} \phi' &\leftarrow \tau \phi + (1 - \tau)(\phi + \alpha \nabla_{\phi} L(\phi)) \\ \theta' &\leftarrow \tau \theta + (1 - \tau)(\theta + \alpha \nabla_{\theta} L(\theta)) \end{aligned} \quad (7)$$

TD3 introduces three distinctive methodologies to address the shortcomings of deterministic policy-based methods, which include target policy smoothing, clipped double-Q learning, and delayed policy updates [22]. The deterministic policy in DDPG simplifies calculations but can lead to overfitting. To mitigate this issue, the algorithm introduces clipped double-Q learning which uses two action value functions. Each Q function is updated separately, but to prevent overfitting, the TD target is shared using the smaller output value of the two Q functions during updates. Target policy smoothing enhances policy stability and facilitating exploration by adding noise into the policy actions and clipping the action. Delayed Policy Updates is the approach that updates the policy only after the critic has converged to a certain degree rather than updating the policy and critic at every step.

2.2 AI for UAV Navigation with Obstacle Avoidance

The conditions and goals of DRL-based navigation system are as follows:

- The system possesses knowledge solely of its target destination, lacking any prior information about the surrounding environment, including the location and configuration of obstacles.

- While navigating towards its designated target position, the UAV must demonstrate the capability to effectively circumvent obstacles.
- Utilizing depth image, relative position, and direction, the system autonomously determines and executes appropriate control actions.

Numerous studies have focused on developing the most efficient system that satisfies these functionalities and conditions [11, 19, 23–27]. Recent studies predominantly share a common structure for neural network that outputs actions, fundamentally considering both functionality and learning efficiency. This structure involves depth image input being processed through convolution layers for feature extraction, after which the UAV’s physical states such as relative distance and direction are combined with the extracted image feature in a concatenation layer. Subsequently, in the concatenation layer, the combined information passes through multiple linear layers to ultimately output the action. The physical state has a significantly smaller dimension compared to the depth image. Therefore, processing it separately is more efficient, as combining it with the depth image during training may result in poor reflection of this information.

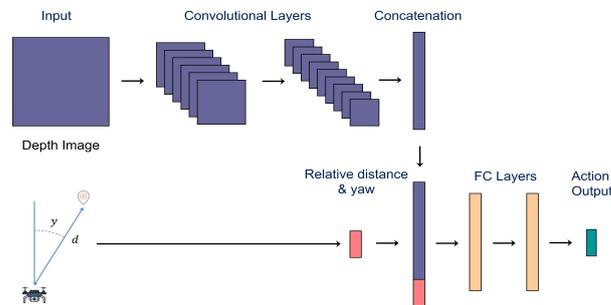


Fig. 1: The basic structure commonly shared by recent studies when designing navigation systems based on DRL.

2.3 Sensor Fusion and Anomaly Detection

Extended Kalman Filter (EKF) is an algorithm used to estimate the state of a system from sensor measurements and a system’s nonlinear dynamic model [28]. EKF sensor fusion combines sensor measurements collected from multiple sensors and estimates the system state, which is achieved through two main processes. One process involves updating the system state through the dynamic model, during which sensor inputs are not specifically incorporated. Another process involves incorporating sensor measurements into the system state. This process reflects the difference between the internal estimates of the system and the values estimated from the sensors. The latest robotic vehicle systems detect a significant discrepancy, it does not incorporate this difference and instead determine it as an anomaly value [15, 29–32]. If sensor measurements are continuously identified as anomalies, they may be classified as a spoofing attack, leading to the suspension of sensor data incorporation in robotic vehicles.

The process of incorporating sensor values in EKF sensor fusion can be summarized as follows:

$$\begin{aligned} est'_{next} &\leftarrow pred(est_{cur}) \\ est_{next} &\leftarrow est'_{next} + K \times (obs_{cur} - h(est'_{next})) \end{aligned} \quad (8)$$

where est'_{next} represents the values predicted from the dynamic model based on the system's internal estimates est_{cur} before incorporating the sensor values, K indicates the degree to which sensor values are incorporated. est'_{next} is corrected through the obs_{cur} obtained from the sensor and becomes the next system internal state est_{next} . h is the relationship function between the sensor values and the internal state estimates. Generally, for each update of the internal system estimates through sensor values, the system decides whether to incorporate the sensor values based on the magnitude of $(obs_{cur} - h(est'_{next}))$ and this is also used for anomaly detection. This paper models the practical range of possible spoofing to assume more realistic attacks in PX4 which is the most widely used UAV controller systems in research and industry [29]. The incorporation of sensor values and anomaly detection is fundamentally applied in PX4. In PX4, the test ratio is calculated using $(est'_{next} - obs_{cur})$ to detect anomalies. In other words, if the test ratio exceeds 1, the sensor values are not incorporated, and prolonged non-incorporation is used to detect spoofing attacks or abnormal sensor states.

3 Proposed Method

3.1 Attack Target System Model

This paper proposes modeling an attack target system that extends from the anomaly detection and sensor estimation algorithm layer of PX4 to the DRL navigation model. As mentioned in Section 2.2, the DRL-based navigation system outputs the velocity that a UAV should take through inputs of depth image, relative position, and yaw. Basically, in the DRL system, the target position is determined at the beginning of a mission, and the current position is continuously estimated in EKF module in PX4, allowing to calculate the relative position to the target. The relative direction is determined by calculating the difference between the current direction of the UAV and the direction to the target position, based on the current UAV position which is an estimated value from PX4. The relative distance and direction, which are inputs to the DRL system, are calculated as follows, with the direction expressed as the angle yaw.

$$\begin{aligned} d_{rel} &= \|pos_{tar} - pos_{cur}\| \\ yaw_{tar} &= \arctan(pos_{tar} - pos_{cur}) \\ yaw_{rel} &= yaw_{tar} - yaw_{cur} \end{aligned} \quad (9)$$

The relative distance and yaw are denoted by d_{rel} and yaw_{rel} , respectively. Meanwhile, pos_{tar} represents the target position, and pos_{cur} signifies the current position. yaw_{rel} is the angular direction of the target relative to the current position. In particular, pos_{cur} refers to the current position state that the UAV recognizes, which corresponds to the system internal state mentioned in subsection 2.3.

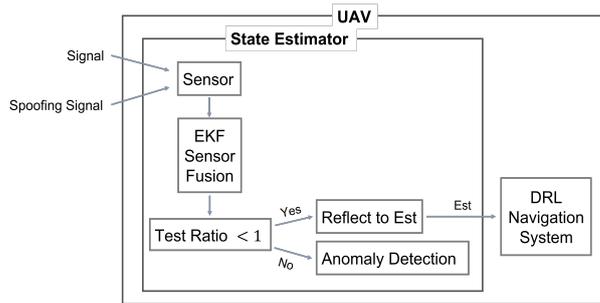


Fig. 2: The practical target model models the spoofing attacks constrained by PX4, and these processed values in PX4 are then passed to the DRL-based navigation system.

3.2 Base Principle of Attack

The DRL-based system for UAV collision avoidance and navigation to a specific target, which is currently used in most research, basically has the following structure as shown in Figure 1. The relative direction to the target alone is insufficient to distinguish between scenarios where the target is positioned between the obstacle and the UAV, and where the target is behind both the UAV and the obstacle. Therefore, the relative distance of the target becomes an essential input value. However, this structure has vulnerabilities. The first issue arises because the decision by a depth image contradicts the decision based on physical states such as UAV’s direction or its relative position to the UAV’s target. Fundamentally, the DRL system processes a depth image and relative physical states of the target simultaneously in the same neural network for handling both movement towards the target and movement to avoid collisions at the same time. Even in situations where the distribution of obstacles around the UAV is the same, the distance and direction to target can lead to different actions. This means that the action output can vary depending on the relative physical state with the target, even when obstacles are very close.

Secondly, physical states such as relative distance, and direction are directly input into the concatenation layer of DRL, thus having a relatively large impact on the action output. While a depth image are difficult to spoof, physical states of UAVs are already relatively easy to manipulate through spoofing attacks like GPS spoofing and electromagnetic interference. As a result, the DRL system is inherently vulnerable to spoofing attacks.

Finally, because a UAV controller must inevitably allow for some degree of sensor error, spoofing within a small range remains possible. The most commonly used UAV autopilot systems like PX4 tend to accept a certain range of error values and reflect them as shown in subsection 2.3. Furthermore, unlike adversarial attacks on depth images, it is known that these spoofing attacks are easily feasible in reality. In particular, systematic methodologies for GPS spoofing attacks are well-researched [31–35]. Additionally, methods for sensor disruption using magnetic fields or ultrasound have also been studied [36–38].

3.3 Attack Scenario

In modern battlefields and complex urban environments, the operation of intelligent UAVs is required to be possible even in extremely complex geographical settings. The goal of the attacker in this attack scenario is to neutralize and disrupt intelligent UAVs by inducing collisions with obstacles. In other words, the core of the attack scenario in this paper is to interfere with obstacle avoidance, which is a core function of intelligent UAVs, through attack vector inputs related to the physical state. This paper assumes a scenario where an attacker performs GPS spoofing attacks on a UAV with the DRL-based system. Figure 3 illustrates the attack scenario presented in this paper.

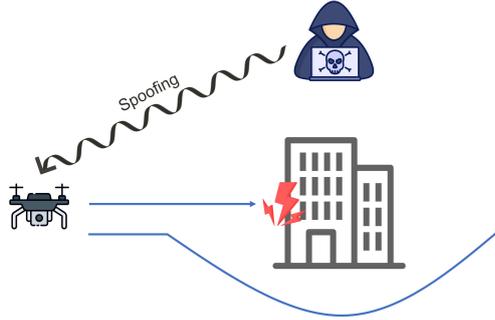


Fig. 3: A scenario where an attacker induces collisions through spoofing on a UAV system navigating in complex environments such as urban, battlefield or natural disaster areas.

3.4 Unconstrained Attack Model

The attack model in this subsection assumes that there are no limitations on the range of attack vectors used as GPS spoofing input to the DRL system. It assumes a situation where the anomaly detection function for sensor inputs in the UAV’s control system is not activated. Even if the range is exceeded, it is not detected by autopilot control system. In other words, this can be seen as an attack model for targets that does not consider EKF sensor fusion detection layer, and it can be viewed as an attack model equal to the models used in general adversarial attacks [11].

This paper proposes a brute force attack methodology to induce collisions in this attack model. In a UAV near collision objects, the attacker seeks to find values in the input space of relative position and yaw that produce outputs guiding the UAV towards a collision. Subsequently, the attacker can inject the values as relative position and yaw values into the DRL system. Algorithm 1 represents our brute force attack process. pos is the spoofed position inputs of the DRL. An attacker examines the output values for all inputs up to P , which are the maximum input values for pos to find the value that meets the condition which induces a collision. The current position of the UAV can determine the relative distance d_{rel} and yaw y_{rel} to the target, which are inputs to the policy algorithm π of the DRL, through Equation 9. *Condition* refers

to a function that, from the attacker’s perspective, determines whether the policy’s output $\pi(s)$ can induce a collision. In other words, the desired relative distance and yaw can be determined by spoofing the current position. Finally, these values are used for a spoofing attack on the DRL system. Additionally, it is assumed that the attacker knows the input of the depth image. In the attack scenario, it is assumed that the depth input that the UAV is looking at can be sufficiently estimated through the UAV’s position and orientation.

Algorithm 1 Unconstrained Attack Model

```

UAVMission
while !done do
  for  $pos = 0$  to  $P$  do
     $d_{rel} \leftarrow \sqrt{pos_{tar} - pos}$ 
     $y_{rel} \leftarrow \arctan(pos_{tar} - pos) - yaw_{cur}$ 
     $s \leftarrow (Image, d_{rel}, y_{rel})$ 
    if  $Condition(\pi(s)) == \text{true}$  then
      attacker transmit pos
    end if
    UAVaction
  end for
end while

```

3.5 Constrained Attack Model

Currently, UAVs typically have built-in anomaly detection capabilities, which prevent the incorporation of detected anomalous sensor values as described in subsection 2.3. Therefore, this paper establishes a more practical attacker model, targeting the system described in subsection 3.1 and presenting a more realistic attack threat. The conditions of this attack model are as follows:

- The attacker can only interfere with the sensor inputs of the target system described in subsection 3.1.
- The attacker does not know the internal state of the controller system.
- The attacker operates in a way that can evade anomaly detection of the controller system.

To overcome this, we propose methods for spoofing sensor values and explore the extent to which such spoofing is possible. Through this analysis, we have designed a practical attack model.

Spoofing Strategy in PX4: Despite the fact that the internal estimated values of sensors cannot be known without accessing the UAV’s internal systems, this paper demonstrates how an attacker can make the internal estimated value converge on the value that the attacker wants through spoofing, thereby indirectly getting the estimated values. In the attack model, it is assumed that as a minimum condition

for spoofing attack, even if the attacker cannot access the internal of the UAV, the attacker can measure the true position of the UAV from the outside.

Algorithm 2 Proposed GPS Spoofing Attack Strategy

```

UAVMission
 $n \leftarrow 0$ 
while !done do
  attacker sets  $\Delta p_n$ 
  for  $t = 0$  to  $T$  do
     $sig_n \leftarrow pos_{uav} + \sum_{i=0}^n \Delta p_i$ 
    transmit  $sig_n$ 
  end for
   $n \leftarrow n + 1$ 
end while

```

Algorithm 2 presents a methodology for conducting a spoofing attack by utilizing the convergence of estimated values. n is the step unit in which the DRL system determines the control action. Δp_n represents the degree of change in the UAV’s estimated position that the attacker desires for one step. T is the number of spoofing attacks required to reflect this change in UAV. $\sum_{i=0}^n \Delta p_i$ is the total spoofed value reflected in the internal estimates of PX4 over n steps. pos_{uav} is the current position of the UAV. In other words, T spoofing attacks are required for pos_{uav} to be reflected, and afterwards, the attacker must continuously add these reflected changes to the UAV’s current position. The condition of the attacker model for a spoofing attack without detection by UAV is that the attacker can observe the actual position of the UAV from the outside. Therefore, pos_{obs} can be considered almost identical to the UAV’s actual position. In the experiments of this paper, it was demonstrated that approximately 30 spoofing attacks can change the latitude and longitude by 0.00003 each. Generally, in UAVs, the process of receiving and updating GPS signals occurs about 10 times per second which is the default setting for PX4. In conclusion, since a change of 0.00001 degrees in latitude or longitude is approximately equal to 1 meter, this means that it is possible to change the relative position inputs of DRL system by up to 1 meter along the x and y axes every 1 seconds.

This paper proposes a methodology for inducing collisions in a practical attack model. The practical attack model indicates that the internal estimated position of PX4 can be changed by 0.1 meter every 0.1 seconds. The internal estimated position determines the relative position, the input value for the DRL system in Equation 9. Due to the narrower attack range compared to unconstrained attack model, this paper has induced actions that continuously move towards the collision object. Generally, if within a certain distance from the target, the optimized DRL system will take actions that reflect the momentum needed to move towards the target according to the hypothesis of this paper Section 3.2. This paper proposes that when the input values of relative position are within a certain range and the UAV is near the collision object, the attacker uses vector values that align with the direction of the collision object to

conduct spoofing attacks on the relative position and direction inputs. In other words, the attacker continuously injects spoofing attack vectors into the UAV, creating an effect similar to that of an obstacle exerting an attractive potential field on the UAV.

Algorithm 3 Constrained Attack

```

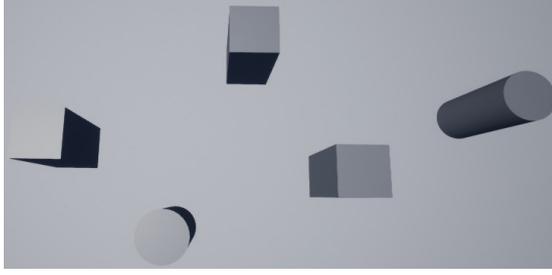
UAVMission
 $n \leftarrow 0$ 
while !done do
   $\Delta p_n \leftarrow \text{norm}(pos_{col} - pos_{tar})$ 
  for  $t = 0$  to  $T$  do
     $sig_n \leftarrow pos_{uav} + \sum_{i=0}^n \Delta p_i$ 
    transmit  $sig_n$ 
  end for
   $pos_{est} \rightarrow sig_n$  (convergence)
  UAVaction
   $n \leftarrow n + 1$ 
end while

```

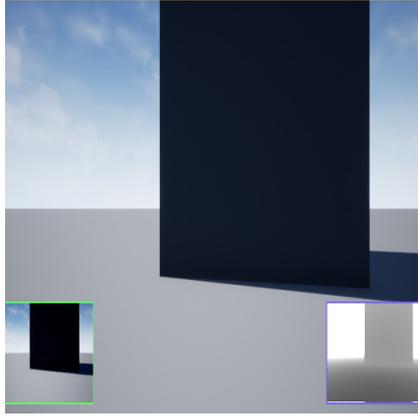
Algorithm 3 represents the process of the proposed spoofing methodology which functions equivalently to the effect of the potential field created by obstacles. In this methodology, it is assumed that DRL outputs control actions at intervals of 0.1 seconds. The attacker performs spoofing T times to change the UAV’s estimated value by Δp every 0.1 seconds. Applying a change of vector $\text{norm}(pos_{col} - pos_{tar})$ to pos_{est} ultimately causes relative position and direction (d_{rel}, yaw_{rel}) to the target to change in the direction towards the obstacle. The positions of the collision obstacle and the target are represented by pos_{col} and pos_{tar} , respectively. π is the DRL navigation system which outputs the control action. In our proposed attack method, uniform field $\text{norm}(pos_{col} - pos_{tar})$ are applied to spoofing attack for inducing the UAV to obstacle.

4 Experiment

AirSim is a highly realistic flight simulator based on Unreal Engine which provides physically and visually realistic environments [39]. In particular, it is optimized to provide an environment for AI research such as deep learning, computer vision, and reinforcement learning for UAVs. AirSim provides photo-realistic environments and delivers real-time simulation data through a Python API, enabling AI system to effectively communicate with the UAV during each simulation frame. This integration allows researchers to develop and test their algorithms in a highly realistic setting. The target of the attack in this paper is a DRL-based navigation system which is trained using TD3 based on the structures in [19, 24]. DRL training and an agent is primarily based on the stable baseline library [40]. The DRL system basically receives as input a depth image, the target’s position, the current UAV’s position, and current yaw. The DRL system outputs the speed and the angle of yaw to be changed based on current direction. These are dynamically obtained at each step through AirSim’s Python API.



(a) Top view of flight environment map used in the experiment.



(b) Front view of flight environment map used in the experiment.

Fig. 4: Flight experiment environment in the AirSim simulator.

The DRL system receives input every 0.1 seconds and then proceeds with the control action step. Figures 4a and 4b represent the flight environment from the top view and front view, respectively.

Lastly, to demonstrate the range of spoofing attack possible in the EKF system of an UAV autopilot system, PX4 and Gazebo simulation system were utilized [29, 41].

4.1 Unconstrained Attack Model

In this experiment, we demonstrate that a spoofing attack can induce a collision using the methodology described in subsection 3.4. The assumption of this experiment is that there are no restrictions on the inputs of the DRL system. Therefore, for all input spaces, an attacker can choose the desired output value, which leads to inducing collisions. Figure 5 shows the experimental results. Both the green line and the purple line represent the paths of UAVs that were given the mission to reach the target. However, the green line shows the path when a spoofing attack was carried out, while the purple line shows the path when no attack was conducted. When the spoofing attack was carried out, it was possible to induce a collision. When no attack was conducted, we can see that the UAV moved accurately to the target. The experimental

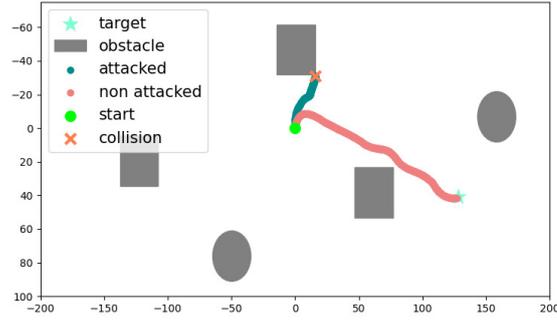


Fig. 5: This figure presents the results of inducing a collision using unconstrained attack methodology. The green line represents the path when the attack is carried out, while the pink line shows the path without the attack.

results show that the attacker can completely reverse the UAV's original path and cause collision.

4.2 GPS Spoofing Strategy in PX4

The objective of this experiment is to determine to what extent the spoofing attack model proposed in this paper can manipulate the UAV's estimated position per unit time. As mentioned earlier, in this experiment, we conducted tests on GPS spoofing, which is the most realistic and systematized spoofing attack method. The structure of the experiment involves examining how the internal estimates of PX4 are affected when spoofed values different from the actual position are input into PX4 autopilot system. To set up the environment where PX4 runs, we utilized Gazebo simulator. The overall experimental structure is shown in Figure 6.

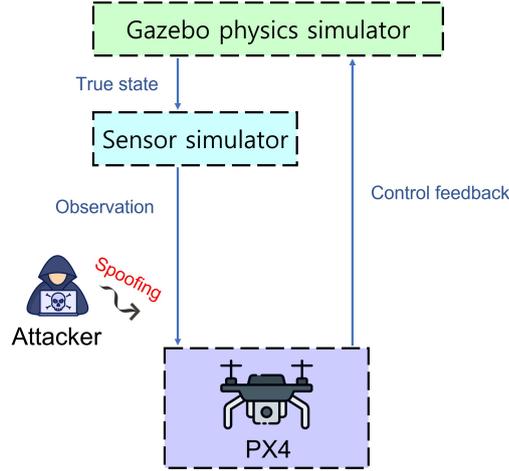


Fig. 6: The true values transmitted from the physical simulator are passed to sensor simulator and PX4 receives the simulated sensor value. To transmit spoofing values to PX4, this paper use the communication layer between sensor simulation and PX4.

The estimation of EKF sensor fusion determines the anomaly of sensor values from external sensor sources by comparing them with the current internal estimated values, and then reflects these values. At the initial stage of the attack, the internal estimated value will have little difference from the actually observed position, but as the attack progresses, it will diverge from the externally observed position. Therefore, the attacker must continuously predict the internal estimated value to be able to perform spoofing. Therefore, the attacker must be able to predict how the internal estimated value changes when the spoofing value is applied. To achieve this, the attacker needs to find a value that causes the change in the internal estimated value to converge as the spoofing attack progresses. For this purpose, in this paper, we heuristically found the value to which the internal position converges through the external position and spoofing attack values. The basic principle is referenced in [32]. In this experiment, for every 30 GPS samples, the attacker changed both latitude and longitude by 0.0003 in the desired direction. This represents the time it takes for the change applied to the internal estimated value over 30 samples to converge. Figure 7 shows the true position and the UAV's internal estimated position when the spoofing attack is executed starting from 15 seconds. A difference of approximately 0.00035 occurred between the true value and the estimated value over the period from 15 seconds to 50 seconds. This indicates that the spoofing values were continuously reflected, and additionally, during the same period, the test ratio maintained a value below 1 as shown in Figure 7. The results of this experiment demonstrated that it's possible to change the latitude by 0.0003 every 3 seconds. Simplifying this slightly, it means that a change of 0.1 meters can be made every 0.1 seconds because 1 degree of latitude or longitude equals to approximately 111 km.

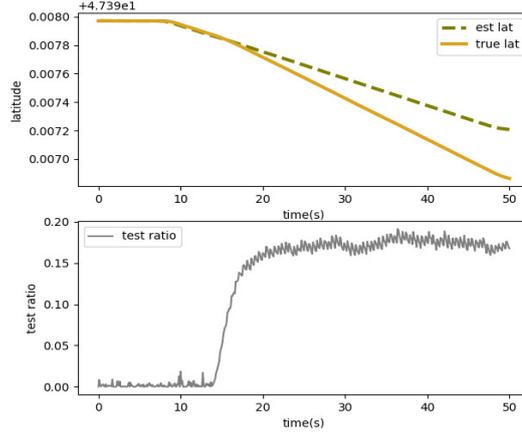
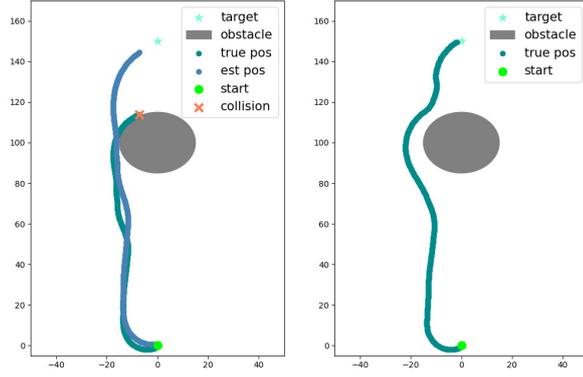


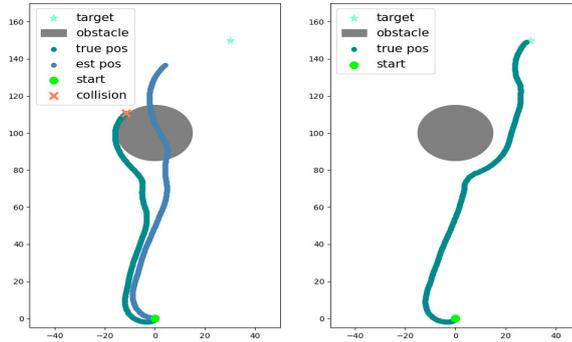
Fig. 7: This figure shows the results when a spoofing attack is conducted using the method proposed in this subsection. The orange line represents the actual position of the UAV, while the blue line indicates the position estimated internally by the UAV. The figure below shows the change in the test ratio value during the spoofing attack, and it can be observed that it does not exceed 1.

4.3 Constrained Attack Model

Based on the experimental results in subsection 4.2, This paper presents a GPS spoofing attack model that closely simulates real-world scenarios. Specifically, given that the DRL model in this experiment operates every 0.1 seconds, the spoofing attack model is designed to assume a possible attack range at intervals of 0.1 seconds. In conclusion, the objective of this experiment was to demonstrate an attacker could induce a collision by assuming that, at each step, the attacker can apply a change of 0.1 meters in the form of a vector that the DRL system recognizes. The experimental environment assumes that the UAV is positioned 100 meters away from the obstacle, with the target located at a distance of 150 meters, allowing for a flight simulation based on this mission. At this point, the attack methodology employed the method used in subsection 3.5. Figure 8 shows the results of the experiment. The upper part shows the results when the target is positioned at (0, 150 m). the left side shows the results when an attack was conducted, while the right side shows the results when no attack occurred. The lower part shows the results when the target is positioned at (30 m, 150 m). The green path indicates the true position of movement, while the blue path in the left represents the position recognized by the UAV. The experimental results show that the attacker can induce a collision not only when the UAV, target, and obstacle are aligned in a straight line, but also when the target is positioned to the side of the obstacle.



(a) Experimental results when the target $(0, 150)$ is aligned with the obstacle $(0, 100)$.



(b) Experimental results when the target $(30, 150)$ is positioned to the side of the obstacle $(0, 100)$.

Fig. 8: This experiment demonstrates the results of a spoofing attack method using a 0.1 m size vector to spoof the position sensor. In each figure, the left side shows the path when the spoofing attack is applied, while the right side shows the result without the spoofing attack. In the left figure, the blue line represents the internally estimated position that the UAV incorrectly recognizes due to spoofing, which differs from its actual position.

5 Discussion

The most common assumption in AI system attacks involves methods that don't separately consider lower-layer components. These approaches focus solely on the inputs and control outputs of DRL-based navigation system without constraints, allowing attackers to freely manipulate positional or directional values. This inherently makes such attacks easier to execute, which experimental results naturally reflect. However, no prior study has yet demonstrated collision induction through positional spoofing

attacks in DRL-based navigation systems for UAV. This paper’s significance lies in proposing such an attack model.

To overcome the limitations of unrestricted attacks discussed earlier, this paper first models the feasible range of GPS spoofing at the UAV navigation layer. subsection 4.2 presents these findings: When the test ratio exceeds 1, PX4’s state estimation module flags it as an outlier. However, GPS spoofing can evade detection by maintaining values below 0.2. Note that such spoofing has inherent motion constraints for UAVs, with spoofable displacements limited to 0.1m per 0.1s. Under these constraints, subsection ?? experimentally validates how attackers can still maximize collision induction efficiency within restricted DRL parameters. Crucially, our method generates a potential field based on the relative direction between the collision target and the UAV’s position during GPS spoofing, proving sufficient for reliable collision guidance. Nevertheless, this study has certain limitations. Due to experimental difficulties in directly integrating DRL systems with GPS spoofing, this paper adopted an alternative approach of modeling each attack separately while imposing constraints on the DRL system’s inputs. However, this paper contributes to more realistic attacker modeling at both the AI system’s module layer and the control system’s state estimation layer compared to previous work. Our approach provides a practical attack scenario that could benefit future AI system security research. A fundamental limitation in practical GPS spoofing implementation remains the prerequisite requirement of knowing the UAV’s precise position through radar or visual observation. Despite these constraints, our assumptions carry significant implications for future research, given the current lack of practical methods to bypass anomaly detection in state estimation systems.

6 Conclusion

There is a growing trend of research on the AI-based UAV navigation system. However, there is very little research on the security aspects of such systems. Despite recent important security papers focusing on sensor attacks using adversarial methods [11], there is still very little research on the security aspects of these systems. Adversarial attacks still have limitations in leading to practical attacks, such as physically injecting subtle noise into sensors. This paper proposes a systematic and more practical attack model based on GPS spoofing attack. Most studies on GPS spoofing attacks against UAVs have assumed attacks on controllers with simple missions, such as moving to a specific location. These studies have focused on changing the UAV’s movement or inducing failsafe mechanisms [31, 35, 42–44]. This paper proposes GPS spoofing attack model for intelligent UAVs equipped with obstacle avoidance capabilities. To achieve this, this paper analyzed structural issues in the input space of state-of-the-art the DRL-based systems and experimentally demonstrated that attacks are possible through only spoofing position value. Unrestricted attacker model has conditions that are just as stringent as those in traditional adversarial attacks. The attacker must know not only the parameters of the DRL policy but also the current position of the UAV and the depth image of the direction the UAV is facing. While this is a theoretically possible attack scenario, it becomes a challenging scenario in practice. Unlike unconstrained model, the attacker of constrained model only needs to know the

position of the UAV. This presents a new attacker model by combining sensor spoofing attacks in DRL system with the limitations of EKF module in PX4 autopilot system. This proposes a more realistic attack threat and becomes a crucial point in future DRL-based navigation system designs. Subsection 3.2 analyzes the vulnerabilities of the most efficient DRL navigation model in recent research. Furthermore, the experiments in this paper conducted attacks on the recently researched DRL systems, thereby validating the attack model proposed in this study. However, this paper has only demonstrated the success of the attack model using uniform vectors, and has not yet presented the efficiency of the methodology using various other potential vectors. These present promising research topics for future work.

References

- [1] Chmaj, G., Selvaraj, H.: Distributed processing applications for uav/drones: a survey. In: *Progress in Systems Engineering: Proceedings of the Twenty-Third International Conference on Systems Engineering*, pp. 449–454 (2015). Springer
- [2] Restas, A.: Drone applications for supporting disaster management. *World Journal of Engineering and Technology* **3**(3), 316–321 (2015)
- [3] Busby, J.: Drone delivery: The danger of opening the air as a commercial highway. *Loy. Mar. LJ* **18**, 287 (2019)
- [4] Jung, S., Kim, H.: Analysis of amazon prime air uav delivery service. *Journal of Knowledge Information Technology and Systems* **12**(2), 253–266 (2017)
- [5] Paucar, C., Morales, L., Pinto, K., Sánchez, M., Rodríguez, R., Gutierrez, M., Palacios, L.: Use of drones for surveillance and reconnaissance of military areas. In: *Developments and Advances in Defense and Security: Proceedings of the Multidisciplinary International Conference of Research Applied to Defense and Security (MICRADS 2018)*, pp. 119–132 (2018). Springer
- [6] Luckey, P.: Anduril founder and ceo palmer luckey in conversation with aprd editor kym bergmann. *Asia-Pacific Defence Reporter* (2002) **48**(8), 28–31 (2022)
- [7] Dew, N., Lewis, I.: Us defense innovation and industrial policy: An assessment of where things currently stand. *Expeditions with MCUP* **2024**(1), 1–25 (2024)
- [8] Chen, S., Zhou, W., Yang, A.-S., Chen, H., Li, B., Wen, C.-Y.: An end-to-end uav simulation platform for visual slam and navigation. *Aerospace* **9**(2), 48 (2022)
- [9] Mac, T.T., Copot, C., Hernandez, A., De Keyser, R.: Improved potential field method for unknown obstacle avoidance using uav in indoor environment. In: *2016 IEEE 14th International Symposium on Applied Machine Intelligence and Informatics (SAMI)*, pp. 345–350 (2016). IEEE
- [10] Escobar-Alvarez, H.D., Johnson, N., Hebble, T., Klingebiel, K., Quintero, S.A.,

- Regenstein, J., Browning, N.A.: R-advance: rapid adaptive prediction for vision-based autonomous navigation, control, and evasion. *Journal of Field Robotics* **35**(1), 91–100 (2018)
- [11] Hickling, T., Aouf, N., Spencer, P.: Robust adversarial attacks detection based on explainable deep reinforcement learning for uav guidance and planning. *IEEE Transactions on Intelligent Vehicles* (2023)
- [12] Sato, T., Shen, J., Wang, N., Jia, Y., Lin, X., Chen, Q.A.: Dirty road can attack: Security of deep learning based automated lane centering under {Physical-World} attack. In: 30th USENIX Security Symposium (USENIX Security 21), pp. 3309–3326 (2021)
- [13] Zhong, Z., Kaiser, G., Ray, B.: Neural network guided evolutionary fuzzing for finding traffic violations of autonomous vehicles. *IEEE Transactions on Software Engineering* **49**(4), 1860–1875 (2022)
- [14] Zhou, X., Schmedding, A., Ren, H., Yang, L., Schowitz, P., Smirni, E., Alemzadeh, H.: Strategic safety-critical attacks against an advanced driver assistance system. In: 2022 52nd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), pp. 79–87 (2022). IEEE
- [15] Quinonez, R., Giraldo, J., Salazar, L., Bauman, E., Cardenas, A., Lin, Z.: {SAVIOR}: Securing autonomous vehicles with robust physical invariants. In: 29th USENIX Security Symposium (USENIX Security 20), pp. 895–912 (2020)
- [16] Shen, J., Won, J.Y., Chen, Z., Chen, Q.A.: Drift with devil: Security of {Multi-Sensor} fusion based localization in {High-Level} autonomous driving under {GPS} spoofing. In: 29th USENIX Security Symposium (USENIX Security 20), pp. 931–948 (2020)
- [17] Khan, A., Ivaki, N., Madeira, H.: A comprehensive study on drones resilience in the presence of inertial measurement unit faults. In: 2024 54th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), pp. 316–323 (2024). IEEE
- [18] Li, Y.: Deep reinforcement learning: An overview. arXiv preprint arXiv:1701.07274 (2017)
- [19] He, L., Aouf, N., Whidborne, J.F., Song, B.: Deep reinforcement learning based local planner for uav obstacle avoidance using demonstration data. arXiv preprint arXiv:2008.02521 (2020)
- [20] Lillicrap, T.: Continuous control with deep reinforcement learning. arXiv preprint arXiv:1509.02971 (2015)
- [21] Mnih, V.: Playing atari with deep reinforcement learning. arXiv preprint

arXiv:1312.5602 (2013)

- [22] Fujimoto, S., Hoof, H., Meger, D.: Addressing function approximation error in actor-critic methods. In: International Conference on Machine Learning, pp. 1587–1596 (2018). PMLR
- [23] Kalidas, A.P., Joshua, C.J., Md, A.Q., Basheer, S., Mohan, S., Sakri, S.: Deep reinforcement learning for vision-based navigation of uavs in avoiding stationary and mobile obstacles. *Drones* **7**(4), 245 (2023)
- [24] He, L., Aouf, N., Song, B.: Explainable deep reinforcement learning for uav autonomous path planning. *Aerospace science and technology* **118**, 107052 (2021)
- [25] Krishnan, S., Boroujerdian, B., Fu, W., Faust, A., Reddi, V.J.: Air learning: a deep reinforcement learning gym for autonomous aerial robot visual navigation. *Machine Learning* **110**(9), 2501–2540 (2021)
- [26] Cetin, E., Barrado, C., Muñoz, G., Macias, M., Pastor, E.: Drone navigation and avoidance of obstacles through deep reinforcement learning. In: 2019 IEEE/AIAA 38th Digital Avionics Systems Conference (DASC), pp. 1–7 (2019). IEEE
- [27] Zhang, S., Li, Y., Dong, Q.: Autonomous navigation of uav in multi-obstacle environments based on a deep reinforcement learning approach. *Applied Soft Computing* **115**, 108194 (2022)
- [28] Kalman, R.E.: A new approach to linear filtering and prediction problems (1960)
- [29] Team, P.D.: PX4 Autopilot. <https://docs.px4.io>. Accessed: 2024-10-01 (2024)
- [30] Liu, Y., Li, S., Fu, Q., Liu, Z., Zhou, Q.: Analysis of kalman filter innovation-based gnss spoofing detection method for ins/gnss integrated navigation system. *IEEE Sensors Journal* **19**(13), 5167–5178 (2019)
- [31] Noh, J., Kwon, Y., Son, Y., Shin, H., Kim, D., Choi, J., Kim, Y.: Tractor beam: Safe-hijacking of consumer drones with adaptive gps spoofing. *ACM Transactions on Privacy and Security (TOPS)* **22**(2), 1–26 (2019)
- [32] Jung, J.H., Hong, M.Y., Choi, H., Yoon, J.W.: An analysis of gps spoofing attack and efficient approach to spoofing detection in px4. *IEEE Access* (2024)
- [33] Feng, Z., Guan, N., Lv, M., Liu, W., Deng, Q., Liu, X., Yi, W.: An efficient uav hijacking detection method using onboard inertial measurement unit. *ACM Transactions on Embedded Computing Systems (TECS)* **17**(6), 1–19 (2018)
- [34] Tippenhauer, N.O., Pöpper, C., Rasmussen, K.B., Capkun, S.: On the requirements for successful gps spoofing attacks. In: Proceedings of the 18th ACM Conference on Computer and Communications Security, pp. 75–86 (2011)

- [35] Sathaye, H., Strohmeier, M., Lenders, V., Ranganathan, A.: An experimental study of {GPS} spoofing and takeover attacks on {UAVs}. In: 31st USENIX Security Symposium (USENIX Security 22), pp. 3503–3520 (2022)
- [36] Chen, B., Huang, L., Zhang, K., Hu, J., Zhu, W.: Magnetic interference analysis and compensation method of airborne electronic equipment in an unmanned aerial vehicle. *Applied Sciences* **13**(13), 7455 (2023)
- [37] Kim, H., Bandyopadhyay, R., Ozmen, M.O., Celik, Z.B., Bianchi, A., Kim, Y., Xu, D.: A systematic study of physical sensor attack hardness. In: 2024 IEEE Symposium on Security and Privacy (SP), pp. 143–143 (2024). IEEE Computer Society
- [38] Jang, J.-H., Cho, M., Kim, J., Kim, D., Kim, Y.: Paralyzing drones via emi signal injection on sensory communication channels. In: NDSS (2023)
- [39] Shah, S., Dey, D., Lovett, C., Kapoor, A.: Airsim: High-fidelity visual and physical simulation for autonomous vehicles. In: Field and Service Robotics (2017). <https://arxiv.org/abs/1705.05065>
- [40] Raffin, A., Hill, A., Gleave, A., Kanervisto, A., Ernestus, M., Dormann, N.: Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research* **22**(268), 1–8 (2021)
- [41] Gazebo Simulator. <https://gazebosim.org/home>. [Online]
- [42] Sathaye, H., LaMountain, G., Closas, P., Ranganathan, A.: Semperfi: Anti-spoofing gps receiver for uavs. In: Network and Distributed Systems Security (NDSS) Symposium 2022 (2022)
- [43] Agyapong, R.A., Nabil, M., Nuhu, A.-R., Rasul, M.I., Homaifar, A.: Efficient detection of gps spoofing attacks on unmanned aerial vehicles using deep learning. In: 2021 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 01–08 (2021). IEEE
- [44] Jullian, O., Otero, B., Stojilović, M., Costa, J.J., Verdú, J., Pajuelo, M.A.: Deep learning detection of gps spoofing. In: International Conference on Machine Learning, Optimization, and Data Science, pp. 527–540 (2021). Springer