The Ghost Navigator: Revisiting the Hidden Vulnerability of Localization in Autonomous Driving

Abstract

Localization is crucial for Autonomous Driving (AD), acting as a key foundation that impacts downstream modules. With the advent of Multi-Sensor Fusion (MSF) techniques enhancing accuracy and reliability, the security of such fusion-based localization methods has become a major concern. Extant research has extensively investigated various facets of security in these systems, elucidating vulnerabilities and proposing mitigation strategies. However, the impact of vehicle dynamics on the effectiveness of Global Positioning System (GPS) spoofing attacks has been largely overlooked.

Bridging this research gap, we introduce the Motion-Sensitive Analysis Framework (**MSAF**), focusing on the analysis of previously underestimated dynamics of vehicle dynamics. Our investigation specifically highlights that acceleration from a standstill on straight paths and the transition from deceleration to acceleration within turning are crucial in determining the success rate of GPS spoofing attacks. These scenarios, commonly encountered in a range of driving conditions, demonstrate a pronounced susceptibility when analyzed under **MSAF**. Based on the new insights provided by **MSAF**, we developed two attack strategies tailored to exploit these dynamics. We then evaluated these attack strategies on two commercial autonomous driving systems employing MSF, namely Apollo_MSF and Shenlan_MSF. The results demonstrate a significant attack efficiency improvement by our method: MSAF requires substantially less time to complete attacks compared to the baseline while maintaining comparable success rates. Code and attack demos are available at https://sites.google.com/view/msaf-demo.

1 Introduction

Autonomous vehicles are leading a reimagining of our modes of mobility, marking a significant advancement in automotive technology. Vehicle localization emerges as a fundamental task in autonomous driving (AD), particularly in vehicles equipped with high-level autonomous driving systems [1, 2]. The localization module, essential in determining the vehicle's position and orientation, serves as the primary data source for the entire process. Its accuracy and reliability are crucial, directly influencing the efficacy of downstream modules such as perception, planning, and control [3, 4].

As a crucial tool for acquiring broad global positioning in traditional localization systems, GPS is vulnerable to signal spoofing threats [5–7]. A more robust solution is Multi-Sensor Fusion (MSF) based localization, which leverages the combined strengths of various sensors to improve accuracy and resilience. By integrating observations from GPS, Inertial Measurement Units (IMUs), and the LiDAR locator, MSF localization achieves a more accurate and robust localization system [2, 8, 9]. Despite these enhancements, MSF localization still shows vulnerabilities to spoofing attacks under certain conditions, leading to substantial deviations in vehicle localization [10–13]. These vulnerabilities can induce *takeover effects*, wherein GPS data dominates and inputs from the LiDAR locator are disregarded as outliers, exposing a great challenge in the design of MSF systems.

Prior studies [10, 11]mainly attribute the cause of *takeover effects* to factors like sensor noise and sensor update frequency, while ignoring the impact of the vehicle's dynamic state. Our empirical evaluations reveal that under the combined conditions of turning and acceleration, the *takeover effects* could still be triggered even with minimal changes in sensor noise and sensor update frequency. This indicates that previous analysis tends to underestimate the influence of vehicle motion states on triggering the *takeover effects*. In other words, it is insufficient to only consider scenarios where the vehicle is assumed to be in a stable motion state.

To bridge the identified gap, we introduce a novel Motion-Sensitive Analysis Framework (MSAF), to investigate the security vulnerability of localization under the dynamic motion state. This framework consists of two principal components: offline vulnerability analysis and online exploitation. Briefly, the offline component is dedicated to assessing how varying motion states influence the effectiveness of GPS spoofing attacks, with a focus on two specific scenarios: acceleration from a standstill and transition from deceleration to acceleration within turning (Sec. 2.2). Leveraging these insights, the online component is designed to execute motion-sensitive GPS spoofing attacks, applicable in both simulated and real-world environments. However, there are still a couple of challenges to construct MSAF: 1) the absence of aligned motion data (IMU, GPS, and LIDAR), 2) the scarcity of open-source MSF architectures, particularly the correction modules, and 3) the need to segregate the motion states into 15 distinct dimensions for a thorough impact assessment. More details can be seen in Sec. 3.2.

To address the above challenges, as shown in Figure 4, we develop a Motion Data Generator (Sec. 4.1) in the offline vulnerability analysis phase, capable of generating simulated datasets that include a variety of vehicle motion states and sensor configurations. These datasets contain IMU, GPS data (both clean and malicious), and LiDAR locator data. Following this, a Sensor Fusion Engine (Sec. 4.2) is designed to emulate the integration process of an IMU+GPS+LiDAR fusion structure, performing essential Error State Kalman Filtering (ESKF), including sensor initialization, IMU forecasting, and GPS/LiDAR data updates. This process allows us to assess the effects of GPS spoofing under different motion states. Additionally, a State Dependency Analyzer (Sec. 4.3) is introduced, which utilizes noise-free simulated data to investigate the observability ranking of the core matrices and the variation of Kalman gain for GPS positioning within the sensor fusion process, disentangling the complex dependencies between the 15-dimensional vehicle states, such as position, velocity, orientation, acceleration bias, and gyroscope bias. Based on these offline analysis results, we propose an Injector (Sec. 4.4), which adjusts attack strategies by analyzing the vehicle's real-time motion state (e.g., yaw and speed) and adapting the spoofing intensity to achieve precise and dynamic GPS spoofing attacks. MSAF exposes vulnerabilities within the specific fusion structure and illustrates how to strategically exploit these weaknesses to enhance GPS spoofing attack effectiveness.

To demonstrate the effectiveness of the proposed MSAF, we test it with three LiDAR-based fusion systems (i.e., simulation platforms): Apollo_MSF, Shenlan_MSF, and our MSAF_MSF. We further conduct end-to-end attack validations on actual autonomous vehicles in the real world. The experimental results indicate that the conclusions drawn from MSAF are highly applicable and effective within practical commercial autonomous driving fusion systems.

The main contributions can be summarized below:

- **Unveiling motion state impacts on MSF security analysis.** We highlight a critical but underexplored vulnerability in the MSF localization security analysis: *different motion states, especially acceleration from a standstill, significantly impact the GPS spoofing attack success rates*. This challenges the previously held belief about the minimal impact of varying vehicle speeds on the IMU and shifts the focus of traditional security paradigms to the importance of vehicle motion states in MSF systems.
- **Design and implementation of MSAF: a Motion-Sensitive Analysis Framework for MSF security analysis.** To explore the overlooked dimension of motion state changes, we propose and develop a prototype of MSAF, focusing on the security analysis of fusion localization systems in autonomous driving affected by subtle variations in motion states. Implemented on a noise-free dataset, MSAF is designed to enhance the understanding of how different motion states impact the GPS spoofing attack success rates. The prototype and the dataset will be open-sourced to support further research in this area.
- **Evaluating MSAF on the commercial vehicle.** Through end-to-end experiments on real-world datasets and two leading commercial fusion localization systems (Apollo_MSF and Shenlan_MSF), we have comprehensively evaluated the effectiveness of MSAF. The results reveal that, compared to the baseline, MSAF significantly boosts the attack efficiency, achieving a substantial decrease in the time needed to conduct successful attacks while preserving similar success rates. Additionally, MSIF highlights the capability to conduct GPS spoofing *without the need for an additional vehicle to physically tail the victim in real-time*, thereby considerably simplifying the attack mechanism.

2 Background and Threat Model

2.1 Background

AD Localization and Multi-Sensor Fusion. In autonomous driving systems, integrating MSF algorithms is crucial to achieve the necessary localization accuracy for robust navigation. MSF algorithms merge data from sensors like LiDAR, GPS, and IMUs, providing a comprehensive understanding of the vehicle's position and orientation, and effectively overcoming each sensor's limitations for improved accuracy [14]. LiDAR sensors are vital for creating high-resolution 3D maps for path planning and obstacle avoidance, though their performance may decline in adverse weather or featureless environments [15, 16]. GPS is essential for global positioning but can be unreliable in signal-obstructed areas such as urban canyons or dense forests [17]. IMUs track motion and orientation but are prone to error accumulation over time [18]. This integration ensures autonomous vehicles navigate safely and efficiently, adapting to diverse and challenging conditions.

The Kalman Filter (KF) and its variant, ESKF, are fundamental in MSF algorithms, widely recognized for their applicability in both academic and industry settings due to their ability to estimate the state of dynamic systems with high accuracy [10, 19–21]. ESKF operates on the principle of minimizing estimation errors through a two-step process, making it ideal for linear systems with Gaussian noise [22]. It predicts the current state $\delta\boldsymbol{x}_k$ from the previous state $\delta\hat{\boldsymbol{x}}_{k-1}$ and control inputs $\boldsymbol{B}_{k-1}$, and then updates this estimate with new observational data. Specifically, in the prediction phase, system dynamics and inputs are used to estimate the error state ($\delta\check{\boldsymbol{x}}_k$):

$$\begin{aligned}\delta\check{\boldsymbol{x}}_k &= \boldsymbol{F}_{k-1}\delta\hat{\boldsymbol{x}}_{k-1} + \boldsymbol{B}_{k-1}\boldsymbol{w}_k,\\ \check{\boldsymbol{P}}_k &= \boldsymbol{F}_{k-1}\hat{\boldsymbol{P}}_{k-1}\boldsymbol{F}_{k-1}^{\mathrm{T}} + \boldsymbol{B}_{k-1}\boldsymbol{Q}_k\boldsymbol{B}_{k-1}^{T},\end{aligned} \tag{1}$$

where $\delta\boldsymbol{x}_k$ encapsulates deviations in parameters such as position, velocity, orientation, and sensor biases. The state transition matrix $\boldsymbol{F}_{k-1}$ models how the state evolves over time, while the process noise $\boldsymbol{w}_k$ and its covariance matrix $\boldsymbol{Q}_k$ account for uncertainties in system dynamics and external influences. This estimation and its covariance $\check{\boldsymbol{P}}_k$ reflect the system's anticipated accuracy. Then, the correction phase adjusts these estimates with the latest measurements $\boldsymbol{y}_k$ using the Kalman gain $\boldsymbol{K}_k$, leading to an updated error state $\delta\hat{\boldsymbol{x}}_k$ and covariance $\hat{\boldsymbol{P}}_k$, essential for precise navigation and sensor error correction:

$$\begin{aligned}\boldsymbol{K}_k &= \check{\boldsymbol{P}}_k\boldsymbol{G}_k^{\mathrm{T}}\left(\boldsymbol{G}_k\check{\boldsymbol{P}}_k\boldsymbol{G}_k^{\mathrm{T}} + \boldsymbol{C}_k\boldsymbol{R}_k\boldsymbol{C}_k^{T}\right)^{-1},\\ \hat{\boldsymbol{P}}_k &= \left(\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{G}_k\right)\check{\boldsymbol{P}}_k,\\ \delta\hat{\boldsymbol{x}}_k &= \delta\check{\boldsymbol{x}}_k + \boldsymbol{K}_k\left(\boldsymbol{y}_k - \boldsymbol{G}_k\delta\check{\boldsymbol{x}}_k\right).\end{aligned} \tag{2}$$

Here, $\boldsymbol{G}_k$ represents the observation matrix that maps the state space into the measurement space, essential for updating the state estimate with new measurements. The measurement noise covariance matrix $\boldsymbol{R}_k$ quantifies the expected accuracy of the measurements, influencing the weight of the updates. The ESKF broadens the reach of the KF by estimating error states in nonlinear systems, thereby overcoming its limitations and expanding its applicability.

Security Analysis on MSF Algorithms. To analyze the security of MSF algorithms, a fundamental threat model involves attackers who send GPS spoofing signals and intend to deviate the vehicle from the centerline of the lane. However, the positional accuracy provided by the LiDAR locator can mitigate the deception attempts. Thus, attackers must exploit specific vulnerabilities within the MSF model, specifically those model properties that facilitate GPS spoofing efforts. Prior studies [10, 11] has demonstrated that attackers can successfully launch GPS spoofing when the uncertainty associated with the LiDAR locator is high, or the uncertainty of the KF's previous state is significant. Initially, attackers follow the target vehicle closely, transmitting a constant spoofing signal to subtly influence the vehicle's trajectory. This phase aims to incrementally deviate the vehicle from the lane's centerline without triggering immediate detection by the system's anomaly detectors. Once the deviation exceeds a predefined threshold, specifically the distance of the vehicle from the lane's centerline, indicating the vehicle is in a vulnerable state, attackers then escalate their efforts to exponential spoofing. They often overlooked the IMU's dynamic states, assuming minimal impact from the F matrix in Eq. (1) on GPS spoofing. This matrix, incorporating acceleration and angular velocity, becomes critical during rapid vehicle movements, altering its elements significantly. Such assumptions risk ignoring vital model insights. The F matrix, particularly F_{23} and F_{33}, is key to evaluating vehicle acceleration and angular velocity effects on state predictions (Sec. 4.2):

$$F = \begin{bmatrix} 0_{3\times3} & I_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3}\\ 0_{3\times3} & 0_{3\times3} & F_{23} & 0_{3\times3} & C_b^n\\ 0_{3\times3} & 0_{3\times3} & F_{33} & -C_b^n & 0_{3\times3}\\ & & 0_{3\times15} & & \\ & & 0_{3\times15} & & \end{bmatrix}, \tag{3}$$

Where F_{23} reflects the effects of Earth's rotation and vehicle acceleration on navigation, impacting velocity in north, east, and up (NED) coordinates. This matrix also accounts for the Coriolis effect and accelerative forces, which are crucial for high-speed or directional changes.

$$F_{23} = \begin{bmatrix} 0 & -f_U & f_N\\ f_U & 0 & -f_E\\ -f_N & f_E & 0 \end{bmatrix}, \tag{4}$$

F_{33} addresses the impact of Earth's rotation and vehicle angular velocity on orientation, which is crucial for heading changes' state prediction:

$$F_{33} = \begin{bmatrix} 0 & \omega\sin L & -\omega\cos L\\ -\omega\sin L & 0 & 0\\ \omega\cos L & 0 & 0 \end{bmatrix}. \tag{5}$$

It highlights the interplay between vehicle rotation and Earth's gravitational forces, key to navigating under dynamic conditions. Analyzing F_{23} and F_{33} is essential for understanding MSF model susceptibilities to GPS spoofing, especially with active vehicle movement.

2.2 Threat Model

Attack Goals. The attacker attempts to exploit the subtleties of vehicular dynamics by performing GPS spoofing during startups and transition from deceleration to acceleration, aiming to deviate the vehicle towards the curb or into oncoming traffic lanes. Figure 1 showcases *two* attack scenarios:

1. **Straight-based startup attack** targets vehicles when they transition from standstill to acceleration on straight paths, covering conditions from S1.1 to S1.2.
2. **Turning-based acceleration attack** targets vehicles during the transition from deceleration to acceleration within turning, covering conditions from S2.1 to S2.2.

These two cases correspond to the *straight_acc* and *turning_yaw_vel* scenarios as outlined in Sec. 5.1. Each scenario is crafted to leverage specific motion states, with the essence of the attack revolving around selecting the appropriate strategy to adjust the GPS signal. The aim of these strategies is to maximize the deviation of the vehicle's actual trajectory from its intended path.

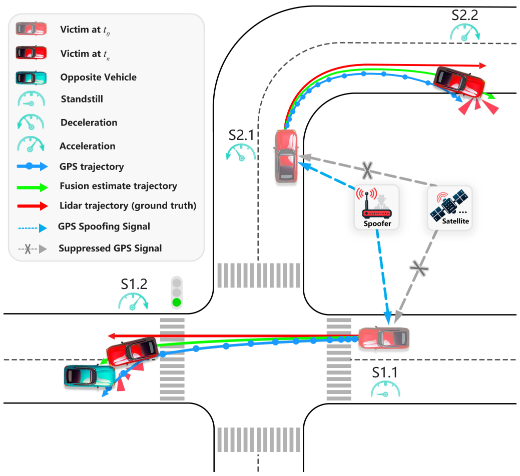


Figure 1: Illustration of *two* attack scenarios.

Attacker's Capability. 1) The attacker can ***obtain a fusion architecture*** similar to the target vehicle through open-source projects or other public channels, thereby conducting vulnerability analysis for various motion states. 2) The attacker can ***determine the target vehicle's motion state*** by employing object tracking methods. Leveraging this information, he can craft and broadcast GPS spoofing signals that manipulate both the signal's quality and positional offset. 3) The attacker can ***exploit the timing vulnerability*** by either passively awaiting or actively inducing conditions when the target vehicle is most vulnerable, such as during transitions from standstill to acceleration or from deceleration to acceleration in turning scenarios. He can either anticipate these scenarios to naturally occur or deliberately provoke them, for instance, by decelerating or stopping abruptly in front of the victim's vehicle.

3 Motivation and Challenges

We present a motivational example to demonstrate the vulnerabilities inherent in MSF algorithms and the challenges involved in analyzing these vulnerabilities across various motion states. This example will help in understanding the complexities and potential weaknesses of MSF algorithms.

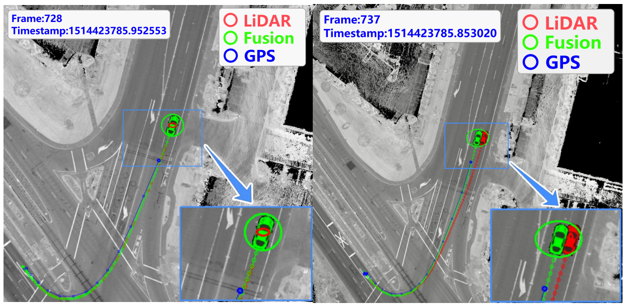


Figure 2: The failed example (left) and successful example (right) of *takeover effects*.

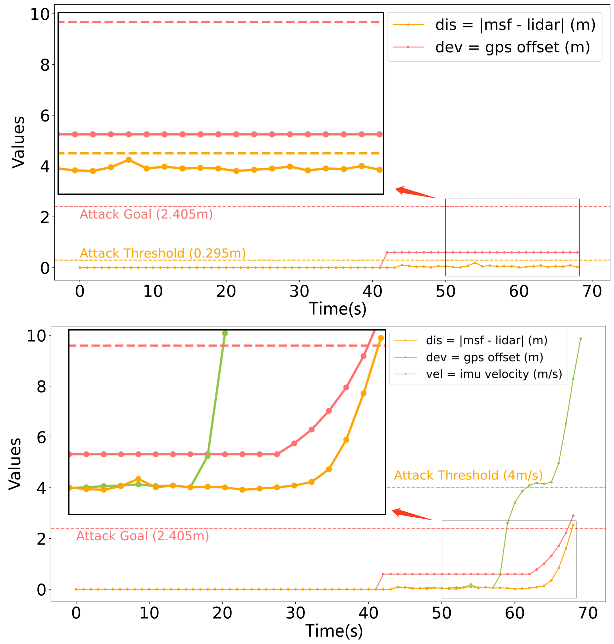


Figure 3: Trigger GPS spoofing attacks based on the discrepancy threshold [10] (top) and speed threshold (bottom). Green, red, and yellow curves denote vehicle speed, GPS spoofing, and vehicle lateral deviation, respectively.

3.1 Motivation

As illustrated in Sec. 2.1, previous security analysis on MSF [10, 11] have concluded that the impact of IMU predictions on takeover effects is negligible. However, we conducted 30 experiments in turning scenarios with significant changes in both linear and angular velocities (provided by the IMU) and found a 16.7% chance of triggering the *takeover effects* (examples in Figure 2), where GPS dominates, and LIDAR measurement is discarded as an outlier. As shown in Figure 3, during the turning and accelerating processes, while the discrepancy (dis) consistently remained below the threshold defined by FusionRipper [10], the exponentially growing deviation still managed to trigger takeover effects. This visualization effectively illustrates the critical interactions of these variables and their overall impact on the MSF model during the takeover effects.

This phenomenon ignites a reevaluation of our approach to MSF security analysis. The traditional reliance on analysis based on a singular positional dimension proves to be inadequate. These intricate dependencies with vehicle states could be crucial in explaining the inconsistencies observed in the effectiveness of spoofing attacks.

To further dissect these complex dependencies, in Sec. 4, we will detail an analytical framework that accounts for the vehicle's motion states. This framework is designed to analyze and quantify the dependencies of the 15-dimensional vehicle states in fusion localization algorithms under various motion states, and how these dependencies affect the success rate of

GPS spoofing attacks. Based on these analysis, more effective attacks can be conducted.

3.2 Challenges

We list some challenges for building our framework.

Challenge 1: *How to overcome sensor data uncertainties in dynamic environments?*

In previous security analysis work, real-world dynamic noise is generally considered the main reason for takeover effects. The primary challenge is to generate a series of high-precision, noise-free datasets to minimize the impact of sensor noise on data quality. This dataset generation needs to focus on two aspects: firstly, it must include acceleration and angular velocity information from IMU, positional and velocity information from GPS, and positional and postural data from the LiDAR locator, ensuring these data are free from noise interference. Secondly, considering the complexity of dynamic environments, the dataset should also reflect different motion states of the vehicle, such as stationary, constant speed, acceleration and deceleration, and turning.

Challenge 2: *How to emulate the black-box fusion structure for assessing the potential importance of velocity?*

Given that commercial MSF algorithms, like those used by Apollo, often provide only a black-box binary module, a significant challenge arises in emulating a similar fusion structure without access to the information within the black-box. We aim to replicate the fusion strategy of the target system to construct an IMU+GPS+LIDAR fusion structure, despite lacking detailed knowledge of the algorithm. This challenge encompasses two main aspects: (1) understanding and emulating the fusion strategy of target's fusion algorithms, especially supporting fusion structures considering velocity and without merging velocity; (2) ensuring the designed fusion strategy can effectively process the data generated in **Challenge 1**.

Challenge 3: *How to Quantify the information capacity of 15-dimensional vehicle states in vehicle dynamics?*

Previous studies have often downplayed the role of vehicle speed in influencing *takeover effects*, with analyses typically constrained to singular trajectories and minor variations in motion states. To gain a nuanced understanding of how different trajectories impact the information capacity of vehicle dynamics, it's imperative to develop a methodology for quantifying the information capacity of various vehicle states (e.g., position, velocity, orientation, gyroscope bias, accelerometer bias) within an IMU+GPS+LIDAR fusion framework. This method should enable the evaluation of the unique information contribution of each vehicle state across diverse motion states, including stationary, constant, acceleration, acceleration and deceleration, and turning. Through detailed analysis and quantification of these contributions, we can refine our strategy for GPS spoofing.

4 Motion Sensitive Analysis Framework

We introduce MSAF to address the above three challenges. Figure 4 illustrates the workflow of MSAF, composed of two main phases: *Offline Vulnerability Profiling* and *Online Exploitation*. In the *Offline Vulnerability Profiling* phase, the **Motion Data Generator** uses simulated data to replicate various vehicle motions and sensor configurations. This synthetic dataset is synchronized and processed by the **Sensor Fusion Engine**, which performs essential filtering tasks, such as IMU prediction, and GPS and LiDAR measurement correction. Concurrently, the **State Dependency Analyzer** examines the observability of core matrices and variation in Kalman gain for GPS positioning within the sensor fusion process. Transitioning to the *Online Exploitation* phase, the **Injector** applies insights from the simulation to physical-world datasets and commercial vehicles.

4.1 Motion Data Generator

The Motion Data Generator is designed to meticulously manage and integrate raw sensor data across a spectrum of motion states, facilitating comprehensive simulations through precise data integration and data synchronization.

Data Integration. In this step, it is challenging to simulate the pose data for the LiDAR locator, as `gnss_ins_sim` [23] primarily supports IMU and GPS data simulation. To address this, positional and attitudinal noise is introduced to mimic real-world inaccuracies. Position $\vec{p}_{lidar}$ is derived by adding Gaussian noise $\vec{n}_{pos}$ with zero mean and standard deviation σ_{pos} to the ground truth $\vec{p}_{gt}$, formulated as:

$$\vec{p}_{lidar} = \vec{p}_{gt} + \vec{n}_{pos}, \quad \vec{n}_{pos} \sim \mathcal{N}(0, \sigma_{pos}^2 I). \tag{6}$$

Orientation q_{lidar} is simulated by adding rotational noise $q_{n_{rot}}$ to the ground truth orientation q_{gt}, represented by:

$$q_{lidar} = q_{gt} \otimes \exp\left(\sigma_{rot} \mathcal{N}(0, I)\right). \tag{7}$$

To support the generation of both benign and malicious signals, *Direct Injection* is applied by adding predefined deviations to a vehicle's GPS data, independent of dynamic state assessments. For straight driving, a fixed deviation $\delta_{straight}$ is added to the GPS position:

$$GPS_{spoofed} = GPS_{original} + \delta_{straight}. \tag{8}$$

In turning scenarios, deviation $\delta_{turning}$ with vehicle's heading θ is used to modify x and y coordinates to simulate a turn:

$$GPS_{x,spoofed} = GPS_{x,original} + \delta_{turning} \cos(\theta); \tag{9}$$

$$GPS_{y,spoofed} = GPS_{y,original} + \delta_{turning} \sin(\theta). \tag{10}$$

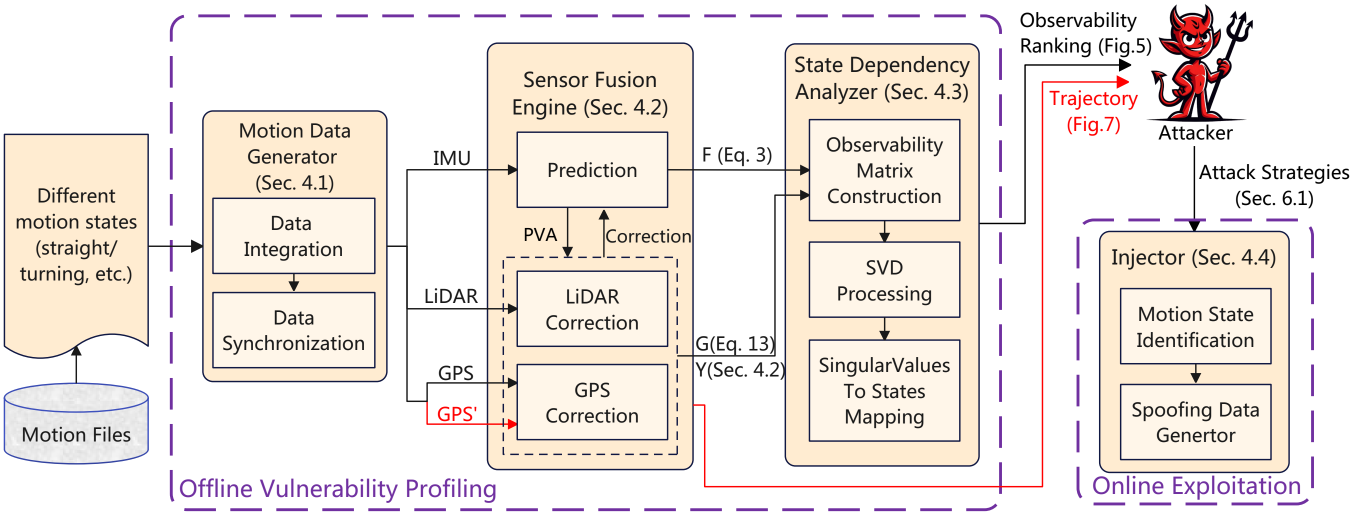

Different motion states (straight/ turning, etc.)
Motion Files
Motion Data Generator (Sec. 4.1)
Data Integration
Data Synchronization
IMU
LiDAR
GPS
GPS'
Sensor Fusion Engine (Sec. 4.2)
Prediction
PVA
Correction
LiDAR Correction
GPS Correction
F (Eq. 3)
G(Eq. 13)
Y(Sec. 4.2)
State Dependency Analyzer (Sec. 4.3)
Observability Matrix Construction
SVD Processing
SingularValues To States Mapping
Observability Ranking (Fig.5)
Trajectory (Fig.7)
Attacker
Attack Strategies (Sec. 6.1)
Injector (Sec. 4.4)
Motion State Identification
Spoofing Data Genertor
Offline Vulnerability Profiling
Online Exploitation

Figure 4: Overview of the proposed Motion Sensitive Analysis Framework (MSAF).

Direct Injection facilitates the simulation of specific movements and disruptions by adjusting GPS data, crucial for detailed motion state simulations and accurate sensor data generation under varied motion states.

Data Synchronization. This module ensures accurate timing and alignment of sensor data, focusing on synchronizing data from IMU (accelerometer and gyroscope), GPS (position and velocity), and LiDAR locator (position and attitudes). By utilizing GPS timestamps as the reference, this module aligns timestamps across these sensor outputs for coherence. By employing linear interpolation for timing alignment, this synchronization process markedly increases the system's precision in handling and amalgamating data from diverse sensors.

4.2 Sensor Fusion Engine

The Sensor Fusion Engine employs the ESKF model to integrate data from IMU, GPS, and LiDAR locator, creating a fusion structure that combines various sensor inputs for precise state estimation. The process involves initializing the state, predicting using IMU data, and updating the state with observations from GPS and LiDAR locator. Error correction is then applied to refine the state estimates, ensuring they align with actual observations.

Prediction. We incorporates the Earth's model to enhance the vehicle's state updates, a methodology that is widely adopted within high-precision integrated navigation systems to significantly improve state estimation and control under various navigational conditions [24, 25]. The model accounts for the Earth's rotation (ω_{ie}^T) and curvature (R_N and R_M), which are integral factors in refining the vehicle's state estimates:

$$\omega_{ie}^T = [0, \omega\cos L, \omega\sin L], \tag{11}$$

$$\omega_{en}^T = \left[-\frac{v_N}{R_M+h}, \frac{v_E}{R_N+h}, \frac{v_E\tan L}{R_N+h}\right], \tag{12}$$

where R_N and R_M, the prime vertical and meridian radii of curvature respectively, are pivotal in calculating the effects of Earth's geometry on the vehicle's motion. When integrated into the system dynamics matrix F_t in the prediction equation (Eq. (1)), they enable precise anticipation of vehicle's state for accurate navigation in both linear and rotational movements.

Correction. In the correction phase, MSAF uses GPS and LiDAR measurements to enhance the predicted states of the vehicle. GPS provides crucial positional and velocity information, while the LiDAR locator offers detailed insights into position and attitude. These inputs are synthesized into the observation matrix G and the observation vector Y, expressed as $Y^T = [dp_{lidar}^T \ dv_{gps}^T \ dq_{lidar}^T \ dp_{gps}^T]$. Here, dp_{lidar}, dv_{gps}, dq_{lidar}, and dp_{gps} represent errors in LiDAR position, GPS velocity, LiDAR orientation, and GPS position, respectively. Subsequently, the Kalman Gain K is determined based on current state estimates and observation data. This gain, derived from the predicted error covariance P and accounting for both process and observation noise, is essential for updating the error state X. Utilizing G and Y, the system identifies and corrects discrepancies between observed and estimated values, thereby refining the vehicle's position, velocity, and attitude estimates. The observation matrix G is defined as follows:

$$G = \begin{bmatrix} I_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times6} \\ 0_{3\times3} & C_n^b & -C_n^b V\times & 0_{3\times6} \\ 0_{3\times3} & 0_{3\times3} & I_{3\times3} & 0_{3\times6} \\ I_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times6} \end{bmatrix}, \tag{13}$$

where C_n^b is the transformation matrix from navigation to body coordinates, and V denotes velocity. G converts GPS and LiDAR observations into refined state error estimations. The error state vector is then reset and accumulated discrepancies are eliminated to maintain system state estimation integrity.

4.3 State Dependency Analyzer

Due to the dynamic nature of autonomous driving systems, the Piecewise Constant Systems (PWCS) method [26–28] is used for state dependency analysis. This method divides the system into constant segments to assess observability—key in control theory and signal processing for deducing a system's internal state from external outputs [29–31]. This aspect is crucial for attackers to infer a vehicle's position, velocity, and orientation. By mapping observability to the 15-dimensional vehicle states, attackers can identify vulnerable motion states across motion scenarios, supporting the development of effective attack strategies.

Observability Matrix Construction. The Observability Matrix Construction begins with the collection and integration of observational data, forming the basis for a detailed analysis. This involves accumulating observational matrices G_i and vectors Y_i across different times, which are then synthesized into a consolidated observational matrix Q_{som} and vector Y_{som}, essential for subsequent analysis steps like SVD and observability assessment. These are defined by the equations:

$$Q_{som} = \sum_{i=1}^{n} G_i \cdot F^{(i)}_{accumulate}, \quad Y_{som} = \sum_{i=1}^{n} Y_i. \tag{14}$$

In this formulation, Q_{som} and Y_{som} encapsulate the system's observability by integrating the effects of state transitions over time through $F^{(i)}_{accumulate}$, preparing the data for comprehensive observability analysis.

Singular Value Decomposition (SVD) Processing. SVD plays a critical role in constructing the observability matrix, allowing for the dissection of the comprehensive observational matrix to uncover its internal structure and characteristics. The SVD of the observational matrix Q_{som} decomposes it into $U \cdot S \cdot V^T$, with U and V representing the left and right singular vector matrices, respectively, and S comprising the singular values. This decomposition is key to understanding the observability of system states, emphasizing the interconnections and dependencies among the system's states.

Singular Values to States Mapping. In the observability analysis phase, SVD is employed to quantify the system states' observability, linking singular values directly to system states. Higher singular values denote greater observability from external inputs, aiding attackers in timing GPS spoofing to exploit low observability of critical states, thus enhancing attack effectiveness. The computation of the observation matrix X via SVD is central to this analysis:

$$X = V \cdot S^{-1} \cdot U^T \cdot Y_{som}. \tag{15}$$

With U, V, S from the SVD of Q_{som}, and Y_{som} reflecting system observations, this formula determines the observability profile. Observable states are highlighted by identifying maximum indices in X, and singular values are then mapped to these states to assess their observability. This mapping quantifies observability, with scores normalized to gauge each state's relative observability within the system.

4.4 Injector

To trigger GPS spoofing attacks, we first identify the motion states and then generate GPS spoofing data.

Motion State Identification. Identifying the vehicle's motion state involves assessing the yaw and speed, critical for understanding orientation and movement to execute GPS spoofing attacks effectively.

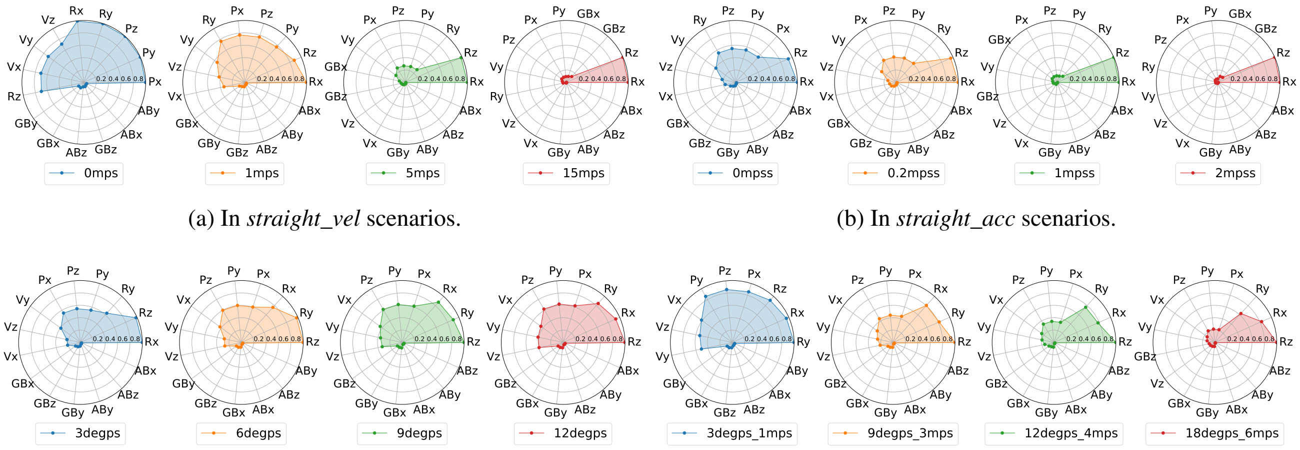
Yaw Identification. Accurate determination of the yaw angle from quaternion data is critical for GPS spoofing to introduce lateral deviations. The yaw angle reflects the vehicle's orientation on the horizontal plane, vital for the alignment of spoofed GPS signals. With a normalized quaternion normalized_q $= (q_w, q_x, q_y, q_z)$, the calculation of the yaw angle ψ incorporates trigonometric equations directly: the yaw angle is derived from $\sin(\psi) = 2 \times (q_w \times q_z + q_x \times q_y)$ and $\cos(\psi) = 1 - 2 \times (q_y^2 + q_z^2)$, leading to $\psi = \text{atan2}(\sin(\psi), \cos(\psi))$. Such precise calculations enable accurate lateral adjustments in GPS spoofing, aligning vehicle's perceived orientation with the intended direction effectively.

Speed Identification. Vehicle speed is crucial for launching GPS spoofing attacks. It is determined by analyzing the vehicle's velocity data, which is derived from real-time motion captured by the IMU. The overall speed of the vehicle (vel) is calculated by taking the square root of the sum of the squares of the vehicle's x and y velocity components: $vel = \sqrt{x_{vel}^2 + y_{vel}^2}$, where x_{vel} and y_{vel} represent the vehicle's velocity components in the horizontal plane. This method accurately reflects the vehicle's speed, which is essential for timing GPS spoofing attacks to match specific vehicle speeds for effective manipulation.

Understanding both the yaw and the vehicle's speed provides a comprehensive view of the vehicle's motion state, aiding attackers in optimizing the timing and execution of GPS spoofing. This ensures that the spoofed signals closely align with the vehicle's actual state, increasing the effectiveness and subtlety of the attack.

Spoofing Data Generation. The underlying principle of the injector model is designed to exploit the motion state of a vehicle, dynamically initiating GPS spoofing when it is either accelerating or moving at a specific speed. This approach leverages the dynamics of the vehicle's movement, enabling more effective and precisely timed spoofing attacks. The revised target function of the injector, which is dependent on the vehicle's motion state, is formulated as:

$$A(t) = \begin{cases} (d \cdot f^i) & \text{under certain conditions,} \\ 0 & \text{otherwise.} \end{cases} \tag{16}$$



(a) In *straight_vel* scenarios.

(b) In *straight_acc* scenarios.

(c) In *turning_yaw* scenarios.

(d) In *turning_yaw_vel* scenarios.

Figure 5: Observability ranking in four scenarios (refer to Table 1 for details). The top1-ranked observable motion state corresponds to the horizontal starting point for counterclockwise rotation.

Table 1: Overview of experimental synthetic scenarios.

Scenarios	Vel (mps)	Acc (mpss)	AngVel (degps)
straight_vel	0, 1, 5, 15	0	0
straight_acc	2	0, ±0.2, 1, 2	0
turning_yaw	2	0	3, 6, 9, 12
turning_yaw_vel	1, 3, 4, 6	0	3, 9, 12, 18

Note: "±0.2" indicates acceleration at 0.2 m/s^2 followed by deceleration at 0.2 m/s^2.

Here, $A(t)$ denotes the injection sequence at time t, with d and f as foundational parameters akin to those in FusionRipper, and i indicating the iteration number. The specific condition for initiating the spoofing process is determined by factors such as vehicle speed and acceleration.

5 Evaluation on Offline Vulnerability Profiling

In this section, we first evaluate the impact of different vehicle states on the target fusion system. Based on the evaluation, we further assess their impact on GPS spoofing attacks.

5.1 The Impact on the Fusion System

Specifically, we assess the observability in straight and turning scenarios without sensor noise. Besides, we also evaluate the Kalman gain, which can gauge ESKF's adaptive reliance on GPS data across different motions. In a nutshell, we find that *velocity plays a pivotal impact on fusion system.*

Experimental Setup. We utilize noise-free data to focus on the impact of motion states on observability. ESKF's noise parameters, including initial, prediction, and observation noises, are set to 1.0×10^{-6}. The vehicle aligns with the y-axis, with each scenario lasting 20 seconds. Sensor frequencies are 100Hz for IMU and 10Hz for GPS and LiDAR. Table 1 details synthetic scenarios with varying speeds, accelerations, and angular velocities for observability analysis.

Observability Variance Across Motion States. Figure 5 shows the observability analysis for MSAF across different motion states. Radar charts represent observability in 15 state dimensions, including position (P_x, P_y, P_z), velocity (V_x, V_y, V_z), attitude (R_x, R_y, R_z), and biases in gyroscope (GB_x, GB_y, GB_z) and accelerometer (AB_x, AB_y, AB_z). There are some key findings as follows:

- In the *straight_vel* scenario, as the vehicle's velocity increases, a decrease occurs in the observability of the position dimensions (e.g., P_x and P_y). In contrast, the attitude dimensions (e.g., R_x and R_z) exhibit increased observability. This suggests that higher speeds may lead to reduced position observability but enhanced attitude observability.
- In the *straight_acc* scenario, higher accelerations lead to a similar trend of decreased position observability, with P_x and P_y being the most affected. This implies that acceleration impacts the observability of position dimensions more significantly than constant velocity does.
- In the *turning_yaw* scenario, where the vehicle maintains a steady velocity while turning, the observability changes are less pronounced for position dimensions, with P_x and P_y showing only minor variations. This reflects relative stability in positional observability during steady-state turning.
- In the *turning_yaw_vel* scenario, where the vehicle experiences changes in both turning rate and speed, we notice a more complex interplay between speed and observability. Particularly, as the vehicle's speed increases, the observability for attitude dimensions, notably R_x and R_z, demonstrates an inverse correlation, with higher speeds leading to decreased attitude observability.

Conclusion 1: The accelerations decrease system observability, increasing vulnerability to the fusion system, while steady turns maintain observability, offering limited advantages for spoofing attacks.

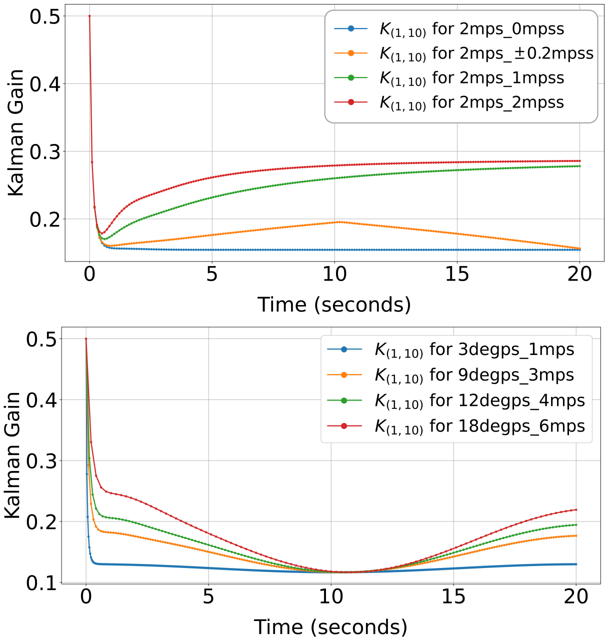


Figure 6: Kalman gain at $K_{(1,10)}$ in *straight_acc* (top) and *turning_yaw_vel* (bottom) scenarios.

Kalman Gain Variance Across Motion States. Here, we mainly focus on *straight_acc* and *turning_yaw_vel* scenarios due to their diverse motion states, ideal for studying Kalman gain ($K_{(1,10)}$) trends for GPS positioning in our MSAF model. These scenarios illustrate how the system's reliance on GPS data shifts with motion dynamics.

- In the *straight_acc* scenario (Figure 6-top), starting from an initial velocity of 2.0m/s and no acceleration, the Kalman gain for position stabilizes, indicating a balanced trust in inertial and GPS data. As acceleration increases, there is a notable upward trend in the Kalman gain, which signifies that the system begins to place a greater emphasis on GPS data. This adjustment in the Kalman filter reflects a strategic shift to counteract the potential inaccuracies in inertial data due to the dynamic motion.
- The *turning_yaw_vel* scenario (Figure 6-bottom) captures how the Kalman gain $K_{(1,10)}$ responds to changes in vehicle speed alone, ranging from 1m/s to 6m/s. Notably, the gain initially decreases and then subsequently increases. This pattern also indicates that the system's reliance on GPS data adjusts in correlation to the vehicle's speed.

Conclusion 2: As vehicle acceleration intensifies, a corresponding increase in the Kalman gain is observed, indicating a heightened dependency on GPS data.

5.2 The Impact on GPS Spoofing Attacks

To execute GPS spoofing attacks, we perform direct data injection into synthetic datasets in straight and turning scenarios, evaluating the influence of different vehicle states.

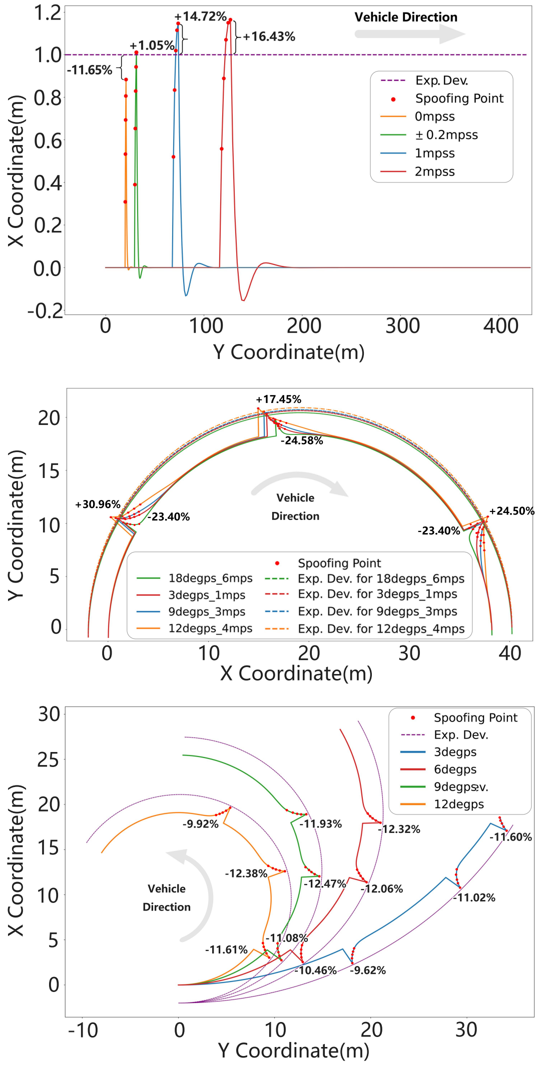


Figure 7: Injection results in *straight_acc* (top), *turning_yaw_vel* (middle), and *turning_yaw* (bottom) scenarios.

Experimental Setup. We follow the setting in Sec. 5.1 to minimize influences from sensor noise, sensor frequency, and ESKF model noise. We execute GPS spoofing injections for the motion states detailed in Table 1, aligning the vehicle forward along the y-axis. By determining the yaw angle as described in Sec. 4.4, we inject lateral deviations with five uniform offset points δ_a (2m) perpendicular to the yaw direction, ensuring consistent lateral injection and robust impact assessment. In *straight_acc* scenarios, a single spoofing instance per trajectory is injected, using a horizontal line at 0 as the ground truth. For *turning_yaw* scenarios, three spoofing instances per trajectory are introduced to explore repeated spoofing effects at different angular velocities. We find two obvious effects of velocity on GPS spoofing attacks as follows:

Deviation Amplification Effect. In *straight_acc* scenarios

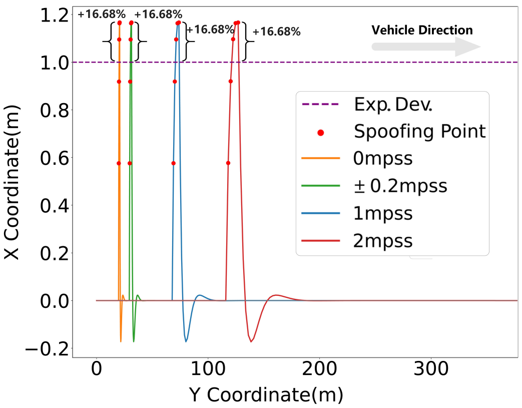


Figure 8: Injection results in *straight_acc scenarios* without velocity fusion. Exp.Dev. denotes Expected Deviation.

(Figure 7-top), deviation is stable without acceleration but increases with vehicle acceleration. At 2 m/s^2 acceleration, deviation can exceed expected values by 16.43%, showing acceleration amplifies GPS spoofing effects. For *turning_yaw_vel* scenarios (Figure 7-middle), as yaw rate and speed increase, the maximum deviation from expected offsets also rises, with up to 30.96% greater deviation at higher speeds and sharper turns, indicating speed and turn sharpness amplify deviation.

Deviation Stability Effect. In *turning_yaw* scenarios with a constant 2 m/s speed and varying angular velocities, offset changes remain consistent, as shown in Figure 7-bottom. The average offset change rate is below 2%, indicating deviation stability despite different angular velocities. This suggests that at steady speeds, angular velocity variations minimally influence GPS spoofing effects.

5.3 Ablation Study

To assess vulnerabilities associated with GPS spoofing attacks, an ablation study was designed to investigate the role of velocity. By excluding the velocity component from the IMU+GPS+LiDAR fusion structure, the study aimed to uncover the extent to which velocity data impacts the Kalman filter's susceptibility to spoofing.

The study revealed that removing velocity information results in a uniform offset increase in straight_acc scenarios, with deviations consistently 16.68% above expected, as shown in Figure 8. This uniformity in deviations, absent velocity data, contrasts with the varied deviations when velocity is included, highlighting velocity's crucial role in the fusion model. This insight is vital for attackers, indicating that exploiting the Kalman filter's reliance on velocity data could lead to more effective spoofing strategies.

6 Evaluation on Online Exploitation

We leverage the above insights for GPS spoofing effectiveness, focusing on online exploitation (adaptive injection methods) for different motion states, assessing MSAF's impact on LiDAR-based MSF systems.

6.1 Experimental Setup

In this section, we present the experimental design on both real-world datasets and synthetic dataset. We begin by introducing the MSF models targeted in our attacks, followed by a description of the datasets employed, and conclude with an outline of our attack strategies and their efficacy.

Victim MSF Modules. Our evaluations target three liDAR based multi-sensor fusion models: Apollo_MSF [4], Shenlan_MSF [32], and our MSAF_MSF. Apollo_MSF, used from Apollo 2.0 to Apollo 9.0, stands as the industry's benchmark for robustness in fusion algorithms. Both Apollo_MSF and the open-source Shenlan_MSF implement ESKF and demonstrate similar accuracy, typically within a 5-10 centimeter range. To complement these, MSAF_MSF, developed in-house, is used to extend our evaluation to simulated scenarios, providing a broader spectrum of testing conditions.

Dataset. The datasets corresponding to each model are as shown in Table 2. To standardize GPS spoofing signals across evaluations, we set the spoofing data's standard deviation to half the median value from the Baidu-64 dataset. This adjustment aims to simulate more consistent and stable spoofing signals. The number in parentheses following each scenario indicates the total instances of that scenario within the dataset, with ten attack tests conducted per scenario to compute the overall success rate.

Attack Strategies. Guided by the Sec. 5, we propose two motion-based attack strategies:

1. **Injection during startup in straight path.** According to the system's reduced state observability and its greater reliance on GPS data, it is evident that the phase of initial acceleration, specifically during startup in a straight path, presents an optimal opportunity for exploitation. Thus, attackers are advised to begin spoofing at this point, progressively increasing the lateral deviation. This strategy takes advantage of the system's vulnerability as it moves from a standstill to motion, utilizing the unique conditions of the initial acceleration phase for maximum impact.
2. **Injection during the transition from deceleration to acceleration within turning.** The above evaluations (in Sec. 5) indicate that during turning maneuvers, particularly when a vehicle decelerates and then re-accelerates, the system's observability varies significantly. Attackers should exploit this by injecting spoofed signals during the deceleration phase and continuing through the subsequent acceleration, potentially intensifying the spoofing effect as the vehicle re-gains speed while still turning.

Table 2: Success rate of two attack strategies under different attack parameters.

Attacked MSF	Dataset	Scenario	Attack Param		Strategy 1		Strategy 2	
			d	f	Off-Road	Wrong-Way	Off-Road	Wrong-Way
Apollo_MSF	Baidu-64	Straight(3)	0.1	1.2	100%	96.7%	-	-
		Turning(3)	0.2	1.2	-	-	100%	90%
	Baidu-128	Straight(2)	0.1	1.2	100%	100%	-	-
		Turning(1)	0.2	1.2	-	-	100%	86.7%
	MSAF-32	Straight(4)	0.2	1.2	100%	100%	-	-
		Turning(5)	0.2	1.2	-	-	100%	93.3%
Shenlan_MSF	KITTI-64	Straight(1)	0.2	1.2	100%	100%	-	-
		Turning(8)	0.2	1.2	-	-	100%	100%
	MSAF-32	Straight(4)	0.2	1.2	100%	100%	-	-
		Turning(5)	0.2	1.2	-	-	100%	100%
MSAF_MSF	MSAF-Sim	Straight(5)	0.1	1.01	100%	100%	-	-
		Turning(5)	0.1	1.01	-	-	100%	100%

6.2 Attack Effectiveness

In our experiments detailed in Table 2, we found that two proposed attack strategies demonstrated high success rates across different datasets and scenarios. Specifically, for the Apollo_MSF model, the attack success rates reached approximately 96.7% on the Baidu-64 dataset, about 95.6% on the Baidu-128 dataset, and as high as approximately 97.8% on the MSAF-32 dataset, revealing its vulnerability in various scenarios. As for the Shenlan_MSF and MSAF_MSF models, our experiments also revealed their high sensitivity to the attacks, with the Shenlan_MSF model achieving a 100% success rate on both KITTI-64 and MSAF-32 datasets, and the MSAF_MSF model also reaching a 100% success rate on the MSAF-Sim dataset. This further illustrates the effectiveness of the attack strategies. Additionally, the off-road and wrong-way attack distances achieved were 2.405m and 2.855m, respectively [10] .

6.3 Ablation Study

Ablation experiments were performed to evaluate the impact of attack parameters relative to attack strategies on the efficacy of GPS spoofing. Parameters were strategically chosen to include FusionRipper's three optimal sets [10] and our best-performing parameters. Additionally, an intermediate set with $d = 0.2$ and $f = 1.3$ was evaluated to bridge the gap between the two extremes and observe its effect on attack success. These selections aimed to explore the range of positional offsets an attacker might attempt to inject. The parameters were tested in real-time against two distinct scenarios, with results presented in Figure 9.

The results distinctly show that FusionRipper's optimally selected parameter sets did not achieve any success, recording a 0% success rate across both strategies. In contrast, our optimally selected parameters accomplished a 100% success rate in each scenario. The aforementioned intermediate parameter set achieved a success rate of 46.6%, underscoring the nuanced influence of parameter adjustments. These findings highlight the importance of selecting a minimal initial offset to enable the ESKF to smoothly adapt to GPS data deviations, which can lead to more effective and stealthy spoofing attacks.

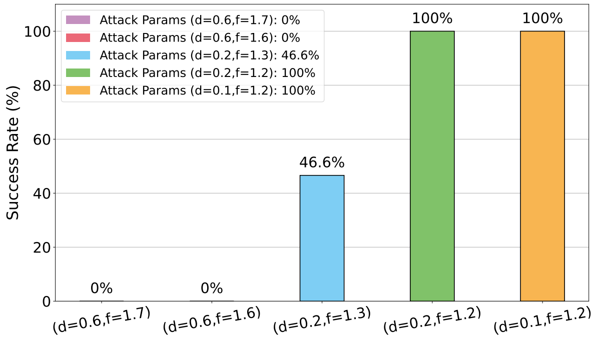


Figure 9: Success rate under different attack parameters.

6.4 Attack Comparison

Experimental Setup. We assess the attack success rates against the state-of-the-art spoofing method, FusionRipper [10]. The Baidu-64 dataset was chosen for its capability to construct point cloud maps, making it uniquely suitable among baseline experimental datasets. Apollo_MSF, consistent with the baseline's choice, served as the victim model for our 30 spoofing experiments, which spanned across three start-up and three constant-speed scenarios.

Comparison of Attack Success Rates. Due to the baseline paper not being open source, we replicated the FusionRipper method on the Baidu-64 dataset. This replication involved an initial phase of constant spoofing using d, followed by exponential spoofing with f upon exceeding a lateral offset threshold of 0.295. Optimal parameters identified in the paper,

$d = 0.6$ and $f = 1.7$, were utilized. Our method diverges by applying exponential spoofing across both the vehicle's start-up and constant-speed phases, requiring merely the identification of the vehicle's motion state to proceed with a unified phase of spoofing. Our selected parameters were $d = 0.2$ and $f = 1.2$, with an attack deemed successful if the vehicle deviated more than 2.855 meters from the road centerline, aligning with the maximum deviation reported in the baseline.

Our baseline method replication on the Baidu-64 dataset achieved a 97% success rate, validating the baseline's effectiveness in capturing the takeover effect in uncertain environments. At the same time, our motion-based attack strategy also reached a success rate close to 97%, comparable to the baseline method. As mentioned in 2.1, as long as attackers could successfully profile vulnerabilities favorable to the attack, the takeover effect could always be triggered. However, despite both methods being successful, we found that our strategies required significantly less time. Therefore, we further compared the attack durations.

Comparison of Attack Durations and Practical Reliability. FusionRipper's initial phase leverages continuous GPS spoofing to exploit vulnerabilities in the MSF system, a process that's challenged by the unpredictable conditions of real-world traffic. As illustrated in Figure 10, MSAF completes attacks significantly faster than FusionRipper. Specifically, at the 90-second mark, MSAF's attack duration was merely 21 seconds, whereas FusionRipper required 101 seconds, most of which was taken up by the first stage alone. Consistently, MSAF needed only 15 seconds at 110 seconds and 18 seconds at 130 seconds to finish the attack, substantially quicker than FusionRipper's respective durations of 112 and 152 seconds. By the 170-second timestamp, MSAF's advantage in time efficiency becomes even more apparent, completing its attack in 22 seconds, compared to the 170 seconds required for the initial phase of FusionRipper, thereby showcasing MSAF's superior temporal efficiency over FusionRipper's threshold-based strategy. We conducted four sets of comparative experiments, assuming FusionRipper begins its first-phase attack at the initial time of the dataset, while we arbitrarily select four points to attack, recording the corresponding attack durations.

MSAF strategically independent of any lateral deviation thresholds, making it significantly less vulnerable to common driving maneuvers such as evasive actions, turns, or lane changes. These maneuvers frequently result in vehicles deviating from the centerline and would typically undermine a deviation-dependent attack strategy like FusionRipper's [10].

6.5 End-to-End Vehicle Evaluation

Prior experiments focused on the impact on the localization module, uncovering and exploiting vulnerabilities under different motion states. However, they did not fully account for how the vehicle's dynamic responses and control strategies could affect the success of GPS spoofing attacks. To address this gap, this section extends the scope of evaluation to include the entire vehicle system, encompassing perception, positioning, planning, and control modules. By conducting experiments on actual autonomous vehicles, we aim to confirm the practical effectiveness of our attack methods on real-world autonomous driving vehicles.

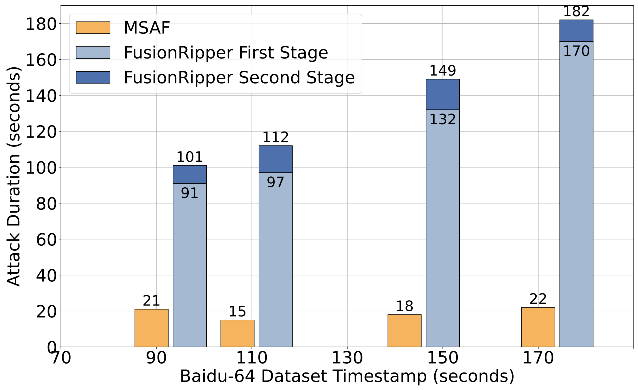


Figure 10: Comparison of attack durations on Baidu-64 dataset.

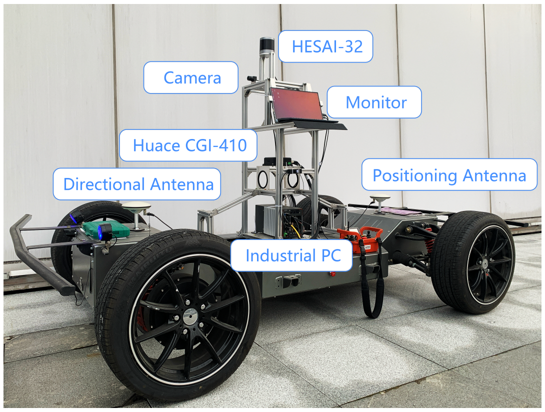


Figure 11: Pix hooke chassis with Apollo 6.0 Edu platform.

Our experimental setup includes a 32-line LiDAR, Huace CGI-410 INS, and a Nuvo-8111 industrial PC with an Intel Core i9-9900K CPU, NVIDIA RTX 3060 GPU, 32GB RAM, and 1TB SSD, integrated with Pix Hooke Chassis and Apollo 6.0 Edu Platform, as depicted in Figure 11. We evaluate the autonomous driving system's response to GPS spoofing at 1m/s and 3m/s speeds across various scenarios, including straight-line driving, turns, and start-up acceleration.

As shown in Figure 12, the end-to-end evaluation, encompassing startup and turning scenarios, demonstrated the successful execution of lateral GPS spoofing attacks, compelling the vehicle to collide with obstacles on either side of the road. These findings unequivocally show that our attack methods can effectively compromise the security of autonomous vehicles by exploiting vulnerabilities in the sys-

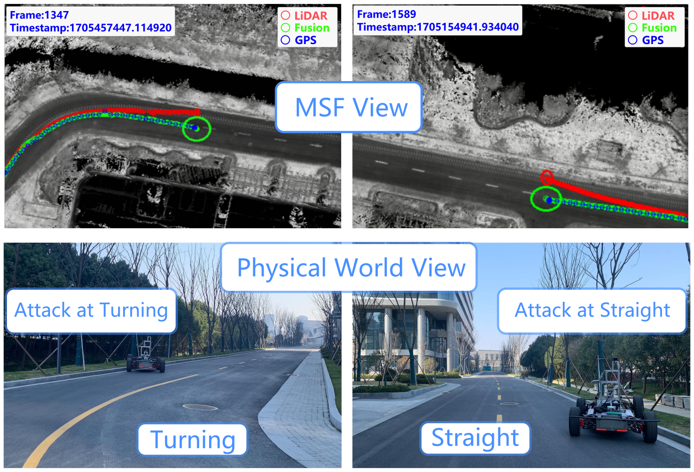


Figure 12: The vehicle hit the curb after GPS spoofing attack.

tem's localization module. More attack demos are available at https://sites.google.com/view/msaf-demo.

7 Limitation

While we simulated attacks on actual autonomous vehicles to demonstrate the real-world implications of our findings, our research inherently possesses certain limitations. Our focus was primarily on a white-box analysis framework of the ESKF fusion structure, incorporating IMU, GPS, and LIDAR locators. However, due to legal constraints, we meticulously avoided actual GPS spoofing, employing perturbations and delays in the spoofing signals to emulate realistic conditions. Furthermore, our hypothesis that an attacker could ascertain the motion state of the lead vehicle was predicated on the utilization of mature technologies prevalent in the field of autonomous driving, such as visual or laser-based methods. Despite their reliability and widespread use in the industry, these sensor technologies were not directly incorporated into our experimental setup.

8 Related Work

Sensor Sppofing Targeting LiDAR. Cao [33] developed a method for attackers to synchronize a photodiode with a LiDAR, creating deceptive points in the point cloud. Tu [34] explored the creation of adversarial 3D objects to mislead LiDAR systems. These objects, however, are noticeable due to their unique shapes and placements. Zhu [35] focused on identifying crucial adversarial positions in physical space, aiming to deceive LiDAR systems more efficiently. Jin [36] designed a physical laser attack against LiDAR-based 3D object detection. These studies primarily concentrate on single-sensor deception strategies targeting LiDAR in autonomous driving systems, overlooking the complexities involved in multi-sensor fusion positioning tasks that incorporate LiDAR.

Sensor Sppofing Targeting IMU. In the realm of IMU spoofing, two main types of attacks are identified. Trippel [37] exposed the susceptibility of MEMS accelerometers to malicious acoustic interference, leading to compromised linear and angular velocity data. Ji [38] manipulating IMU data to disrupt a vehicle's target detection functionality, specifically targeting the system's anomaly detection mechanisms. Similar to the studies on LiDAR deception, research on IMU spoofing predominantly focuses on attacks against individual sensors and does not address the challenges in scenarios involving the fusion of multiple sensors.

Security Analysis on Sensor Fusion Model. Nashimoto [39] explored the vulnerabilities of an Attitude and Heading Reference System (AHRS) under signal injection attacks, demonstrating significant security risks in systems that fuse data from multiple sensors, notably in inclination measurements. This work suggests new directions for bolstering the security of sensor fusion systems. Shen [10] developed FusionRipper, a technique for identifying and exploiting vulnerabilities in LiDAR-based ESKF systems, combining theoretical analysis with simulation experiments to pinpoint critical weaknesses, such as LiDAR locator uncertainty and ESKF initial state uncertainty. Chang [11] found that the sensor update frequency significantly affects the success of GPS spoofing attacks, corroborating FusionRipper's premises. However, vulnerabilities were deemed more critical in steady states, indicating the IMU's limited role in initiating takeover effects.

9 Conclusion

This study illuminates a previously underexplored vulnerability in MSF algorithms used in autonomous vehicles: the profound impact of vehicle motion states on GPS spoofing attack effectiveness. Our introduction of MSAF marks a pivotal shift in understanding the dynamics of AV localization system security. MSAF's meticulous analysis and exploitation of the vehicle's motion state, particularly during critical scenarios like turning and acceleration, reveal a heightened susceptibility to GPS spoofing attacks, which traditional approaches have overlooked. The empirical results from our rigorous testing on commercial autonomous driving systems, Apollo_MSF and Shenlan_MSF, are testament to this newfound vulnerability. The MSAF significantly heightened the success time of GPS spoofing attacks in these real-world scenarios. This stark increase in effectiveness underscores the necessity for a more nuanced consideration of vehicle dynamics in the design and security evaluation of MSF algorithms in AVs.

References

[1] Tong Qin, Yuxin Zheng, Tongqing Chen, Yilun Chen, and Qing Su. A light-weight semantic map for visual localization towards autonomous driving. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11248–11254. IEEE, 2021.

[2] Guowei Wan, Xiaolong Yang, Renlan Cai, Hao Li, Yao Zhou, Hao Wang, and Shiyu Song. Robust and precise vehicle localization based on multi-sensor fusion in diverse city scenes. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 4670–4677. IEEE, 2018.

[3] Athanasios Chalvatzaras, Ioannis Pratikakis, and Angelos A Amanatiadis. A survey on map-based localization techniques for autonomous vehicles. *IEEE Transactions on Intelligent Vehicles*, 8(2):1574–1596, 2022.

[4] Baidu apollo autonomous driving platform. https://github.com/ApolloAuto/apollo.

[5] Harshad Sathaye, Martin Strohmeier, Vincent Lenders, and Aanjhan Ranganathan. An experimental study of GPS spoofing and takeover attacks on uavs. In Kevin R. B. Butler and Kurt Thomas, editors, *31st USENIX Security Symposium, USENIX Security 2022, Boston, MA, USA, August 10-12, 2022*, pages 3503–3520. USENIX Association, 2022.

[6] Harshad Sathaye, Gerald LaMountain, Pau Closas, and Aanjhan Ranganathan. Semperfi: Anti-spoofing gps receiver for uavs. In *Network and Distributed Systems Security (NDSS) Symposium 2022*, 2022.

[7] Tesla model s and model 3 vulnerable to gnss spoofing attacks. https://tinyurl.com/3fxv9hpa.

[8] Chao Qin, Haoyang Ye, Christian E Pranata, Jun Han, Shuyang Zhang, and Ming Liu. Lins: A lidar-inertial state estimator for robust and efficient navigation. In *2020 IEEE international conference on robotics and automation (ICRA)*, pages 8899–8906. IEEE, 2020.

[9] Wendong Ding, Shenhua Hou, Hang Gao, Guowei Wan, and Shiyu Song. Lidar inertial odometry aided robust lidar localization system in changing city scenes. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4322–4328. IEEE, 2020.

[10] Junjie Shen, Jun Yeon Won, Zeyuan Chen, and Qi Alfred Chen. 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)*, pages 931–948, 2020.

[11] Jiachong Chang, Liang Zhang, Li-Ta Hsu, Bing Xu, Feng Huang, and Dingjie Xu. Analytic models of a loosely coupled gnss/ins/lidar kalman filter considering update frequency under a spoofing attack. *IEEE Sensors Journal*, 22(23):23341–23355, 2022.

[12] Jiachong Chang, Feng Huang, Liang Zhang, Dingjie Xu, and Li-Ta Hsu. Selection of areas for effective gnss spoofing attacks to a vehicle-mounted msf system based on scenario classification models. *IEEE Transactions on Vehicular Technology*, 2023.

[13] Junjie Shen, Yunpeng Luo, Ziwen Wan, and Qi Alfred Chen. Lateral-direction localization attack in high-level autonomous driving: Domain-specific defense opportunity via lane detection. *arXiv preprint arXiv:2307.14540*, 2023.

[14] Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[15] You Li and Javier Ibanez-Guzman. Lidar for autonomous driving: The principles, challenges, and trends for automotive lidar and perception systems. *IEEE Signal Processing Magazine*, 37(4):50–61, 2020.

[16] Yuxiao Zhang, Alexander Carballo, Hanting Yang, and Kazuya Takeda. Perception and sensing for autonomous vehicles under adverse weather conditions: A survey. *ISPRS Journal of Photogrammetry and Remote Sensing*, 196:146–177, 2023.

[17] Xue-Bo Jin, Wei Chen, Hui-Jun Ma, Jian-Lei Kong, Ting-Li Su, and Yu-Ting Bai. Parameter-free state estimation based on kalman filter with attention learning for gps tracking in autonomous driving system. *Sensors*, 23(20):8650, 2023.

[18] Zhenbo Liu, Leijie Wang, Feng Wen, and Hongbo Zhang. Imu/vehicle calibration and integrated localization for autonomous driving. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4013–4019. IEEE, 2021.

[19] Jun Zhu, Hongyi Li, and Tao Zhang. Camera, lidar, and imu based multi-sensor fusion slam: A survey. *Tsinghua Science and Technology*, 29(2):415–429, 2023.

[20] Yanbin Gao, Shifei Liu, Mohamed M Atia, and Aboelmagd Noureldin. Ins/gps/lidar integrated navigation system for urban and indoor environments using hybrid scan matching algorithm. *Sensors*, 15(9):23286–23302, 2015.

[21] Zui Tao, Ph Bonnifait, Vincent Fremont, and Javier Ibanez-Guzman. Mapping and localization using gps, lane markings and proprioceptive sensors. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 406–412. IEEE, 2013.

[22] Joan Sola. Quaternion kinematics for the error-state kalman filter. *arXiv preprint arXiv:1711.02508*, 2017.

[23] Aceinna. GNSS-INS-SIM: An Open-source GNSS/INS Simulation Platform. https://github.com/Aceinna/gnss-ins-sim.

[24] Junxiang Jiang, Xiaoji Niu, and Jingnan Liu. Improved imu preintegration with gravity change and earth rotation for optimization-based gnss/vins. *Remote Sensing*, 12(18):3048, 2020.

[25] Feng Sun, Haiyu Lan, Chunyang Yu, Naser El-Sheimy, Guangtao Zhou, Tong Cao, and Hang Liu. A robust self-alignment method for ship's strapdown ins under mooring conditions. *Sensors*, 13(7):8103–8139, 2013.

[26] D Goshen-Meskin and IY Bar-Itzhack. Observability analysis of piece-wise constant systems with application to inertial navigation. In *29th IEEE Conference on Decision and Control*, pages 821–826. IEEE, 1990.

[27] Chris J Dafis and Chika O Nwankpa. Characteristics of degree of observability measure for nonlinear power systems. In *Proceedings of the 38th Annual Hawaii International Conference on System Sciences*, pages 68b–68b. IEEE, 2005.

[28] Fredric M Ham and R Grover Brown. Observability, eigenvalues, and kalman filtering. *IEEE Transactions on Aerospace and Electronic Systems*, (2):269–273, 1983.

[29] Arvo Kaldmäe and Ülle Kotta. A note on observability of nonlinear discrete-time systems. In *2023 62nd IEEE Conference on Decision and Control (CDC)*, pages 7483–7488. IEEE, 2023.

[30] Yifeng Li and Jiandong Zhu. Observability decomposition of boolean control networks. *IEEE Transactions on Automatic Control*, 68(2):1267–1274, 2022.

[31] DRORA Goshen-Meskin and IY Bar-Itzhack. Observability analysis of piece-wise constant systems. ii. application to inertial navigation in-flight alignment (military applications). *IEEE Transactions on Aerospace and Electronic systems*, 28(4):1068–1075, 1992.

[32] Shenlan msf. https://github.com/shenlan2017/Apollo_ShenLan.

[33] Yulong Cao, Chaowei Xiao, Benjamin Cyr, Yimeng Zhou, Won Park, Sara Rampazzi, Qi Alfred Chen, Kevin Fu, and Z Morley Mao. Adversarial sensor attack on lidar-based perception in autonomous driving. In *Proceedings of the 2019 ACM SIGSAC conference on computer and communications security*, pages 2267–2281, 2019.

[34] James Tu, Mengye Ren, Sivabalan Manivasagam, Ming Liang, Bin Yang, Richard Du, Frank Cheng, and Raquel Urtasun. Physically realizable adversarial examples for lidar object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13716–13725, 2020.

[35] Yi Zhu, Chenglin Miao, Tianhang Zheng, Foad Hajiaghajani, Lu Su, and Chunming Qiao. Can we use arbitrary objects to attack lidar perception in autonomous driving? In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pages 1945–1960, 2021.

[36] Zizhi Jin, Xiaoyu Ji, Yushi Cheng, Bo Yang, Chen Yan, and Wenyuan Xu. Pla-lidar: Physical laser attacks against lidar-based 3d object detection in autonomous vehicle. In *2023 IEEE Symposium on Security and Privacy (SP)*, pages 1822–1839. IEEE, 2023.

[37] Timothy Trippel, Ofir Weisse, Wenyuan Xu, Peter Honeyman, and Kevin Fu. Walnut: Waging doubt on the integrity of mems accelerometers with acoustic injection attacks. In *2017 IEEE European symposium on security and privacy (EuroS&P)*, pages 3–18. IEEE, 2017.

[38] 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)*, pages 160–175. IEEE, 2021.

[39] Shoei Nashimoto, Daisuke Suzuki, Takeshi Sugawara, and Kazuo Sakiyama. Sensor con-fusion: Defeating kalman filter in signal injection attack. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*, pages 511–524, 2018.

A Appendix

A.1 Fusion Precision Evaluation

A stable and precise fusion localization simulation framework is fundamental for vulnerability analysis; hence, we evaluate the MSAF's localization performance under an IMU+GPS+LiDAR fusion strategy. This section delves into the precision of MSAF's fusion localization, examining its accuracy on synthetic datasets.

Experimental setup. We focus on the localization accuracy of MSAF in both straight and turning scenarios across four distinct noise levels: noise-free, high-accuracies, medium-accuracies, and low-accuracies. For straight scenarios, the system is tested at a constant speed of 5 m/s, whereas in turning scenarios, it operates at a constant speed of 3 m/s with an

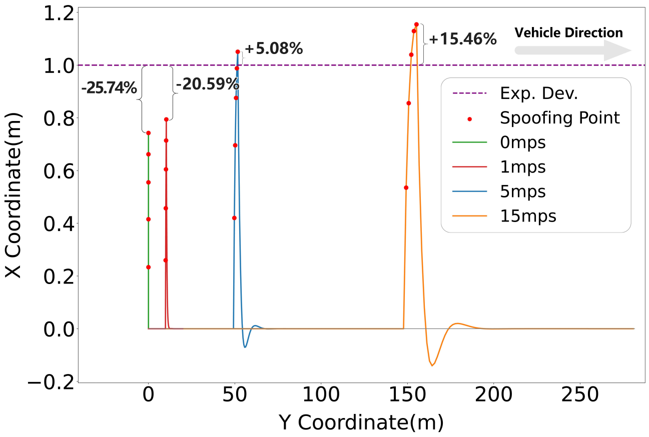


Figure 13: injection results in straight_vel scenarios

angular speed of 9 deg/s. The duration of data for these experiments is set at 20 seconds. In adherence to standard practices for accuracy assessment, we employ the KITTI dataset format for ground truth comparison. To ensure data reliability, the initial 3 seconds of data, encompassing the Kalman filter convergence period, are excluded. After this period, we utilize the open-source evaluation framework evo for Relative Pose Error (RPE) assessment to calculate the Root Mean Square Error (RMSE), providing a detailed and precise validation of MSAF's performance.

Table 3: MSAF Accuracy Across Cases and Noise Levels.

Case	Noise-free(m)	High(m)	Medium(m)	Low(m)
Straight	9×10^{-7}	7.0×10^{-3}	4.4×10^{-2}	1.9×10^{-1}
Turning	1.2×10^{-4}	5.5×10^{-3}	3.7×10^{-2}	1.8×10^{-1}

Results. The MSAF showcases outstanding precision in fusion localization across various noise conditions, as illustrated in Table 3. In straight scenarios, the system achieves an exceptional accuracy of 9×10^{-7} meters under noise-free conditions and sustains accuracy up to 1.9×10^{-1} meters in low-accuracy scenarios. Similarly, in turning conditions, it maintains an accuracy of 1.2×10^{-4} meters in noise-free environments and up to 1.8×10^{-1} meters in low-accuracy settings. These findings highlight MSAF's robustness and adaptability in multi-sensor fusion localization, even in challenging noise environments. The consistent and reliable performance of MSAF in both straight and turning scenarios provides a solid groundwork for in-depth vulnerability analysis.

A.2 Velocity-Offset Dynamics

When the vehicle cruises at a uniform speed, a higher velocity correlates with a more substantial offset. Specifically, surpassing speeds of 15 m/s results in offsets exceeding the expected value by 15.46%, as demonstrated in Figure 13. Following this observation, as the vehicle speed steadily increases and exceeds 15 m/s, the growth rate of the offset begins to decelerate, ultimately stabilizing around 16% above the anticipated value, as demonstrated in Figure 14.

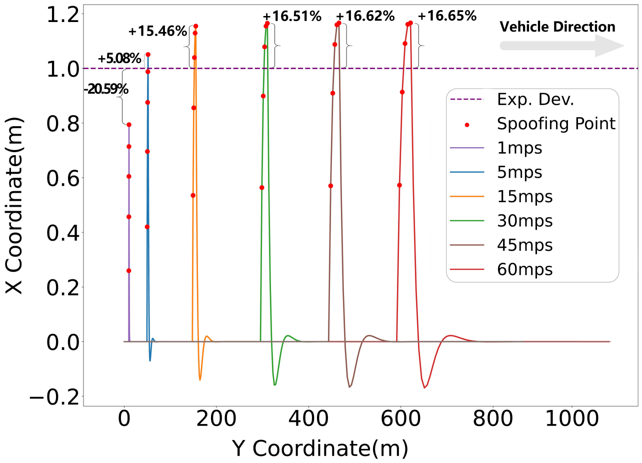


Figure 14: The offset converges to a certain value as the speed increases.