UNIVERSITY OF CALIFORNIA,
IRVINE

Security Analysis of Multi-Sensor Fusion based Localization in Autonomous Vehicles

THESIS

submitted in partial satisfaction of the requirements
for the degree of

MASTER OF SCIENCE

in Computer Science

by

Jun Yeon Won

Thesis Committee:
Assistant Professor Qi Alfred Chen, Chair
Associate Professor Marco Levorato
Assistant Professor Joshua Garcia

2019

DEDICATION

I dedicate this thesis to my parents and little brother, Jong Sung Won, Soo Young Cho and Se Yeon Won, who support my study.

TABLE OF CONTENTS

LIST OF FIGURES

Page

LIST OF TABLES

ACKNOWLEDGMENTS

I would like to thank to Assistant Professor Chen for giving me a chance to work with him and provide me invaluable advice. I respect his insight and enthusiasm to research. I really appreciate his advice and support to my research.

I would like to thank my committee member, Associate Professor Levorato who was my Computer and Communication Networks class professor. Thanks to his teaching, I can apply the knowledge learned from the class to my research. Also, I thank another committee member, Assistant Professor Garcia who was advising my different project, fuzzing. Thanks to his advice, I can learn a lot about fuzzing and different kinds of software such as PIN.

Also, I am glad to research and discuss with my friends in Alfred Research Group, Junjie, Jongho and Takami. I enjoy discussion during the group meetings. I would like to thank my research mate, Junjie, teaching me everything.

I really appreciate my parent's support and love. They always encouraged me to keep doing research and trusted me. Love you.

Lastly, I would like to thank you everyone who supported my research.

ABSTRACT OF THE THESIS

Security Analysis of Multi-Sensor Fusion based Localization in Autonomous Vehicles

By

Jun Yeon Won

Master of Science in Computer Science

University of California, Irvine, 2019

Assistant Professor Qi Alfred Chen, Chair

Autonomous vehicles (AVs) have become close to our life. Many modules are included in AVs such as localization, perception, and planning. Among them, the localization that estimates the current location of AV is one of the most important modules. To make an AV more secure, localization results should be robust. To achieve robustness, localization uses multi-sensor fusion (MSF). MSF uses the Kalman filter to make a result robust. Also, outlier detection is added to MSF to improve results. It filters out the abnormal data from sensors. For sensors, GPS, IMU, and LiDAR are used in MSF. Among sensors, GPS is vulnerable to the spoofing attack. GPS spoofing attacks send fake signals to GPS receivers to deceive them. As a result, it is an on-going problem as to whether MSF can retain robustness when GPS is attacked. We propose an effective GPS spoofing attack method that can affect MSF result. To achieve this, we divide our attack into two steps, profiling and attack application. Also, we introduced two attack parameters, initial spoofing distance and scaling factor. We can calculate optimal and common attack parameters from profiling. We apply the common attack parameter to AVs and define success with a different threshold. To evaluate our attack, we use the data set provided by Baidu. It is a sensor-trace data set for testing MSF. By using the data set, we achieve an 100% success rate in 150 seconds and more than 66% success rate in 2 minutes. Also, we introduce possible solutions to our attack.

CHAPTER 1. INTRODUCTION

Nowadays, Autonomous Vehicles (AVs) are more popular and more accessible to everyone. Many companies have developed AVs at a fast pace. Waymo started a self-driving taxi service in Phoenix, AZ. Baidu collaborated with car manufacturers to deploy their open-sourced AV platform called Apollo [1].

The localization module, which estimates the current location of AV, is one of the most important modules within the AV system. The localization module serves as the basis for other modules in the system to decide a driving strategy. For instance, the AV will determine an obstacle's location based on the localization result. An incorrect localization result will cause the obstacle's location to be inaccurate as well. Also, to plan for the next trajectory that an AV will follow, an accurate localization result is required. In order to achieve a robust localization, AV systems commonly use a Multi-Sensor Fusion (MSF) algorithm in the localization module. In MSF, multiple sensors are combined to provide more accurate estimation of the location of AV. The state-of-the-art MSF designs perform the fusion using the Kalman filter (KF) or its variants (*e.g.*, unscented Kalman filter, extended Kalman filter, etc.) [16]. Multiple sensors such as Inertial Measurement Units (IMU), Global Positioning System (GPS) and Light Detection And Ranging (LiDAR) are used in AV's MSF algorithm.

While MSF can effectively improve robustness against sensor noises, it is still unclear how robust it is under deliberate sensor attacks. It is well known that sensors are vulnerable to spoofing attacks. IMU readings can be affected by sound waves at the resonant frequency of the device [15]. LiDARs are vulnerable to laser injection and blinding attacks [11]. However, such attacks to IMU and LiDAR are not yet proven practical in real-world settings. For example, it is hard to aim at a moving LiDAR for a laser injection attack; IMUs are isolated in thick materials which can dampen the external sound waves. However, GPS spoofing is a classic attack method but still a practical attack vector compared to other sensors. Due to the lack of cryptographic protection in GPS infrastructure, it is fundamentally difficult to fully prevent GPS spoofing attacks today [10].

In this thesis, we perform the security analysis of MSF-based AV localization under the GPS spoofing attack. We discovered that carefully controlled GPS spoofing attacks can corrupt localization estimation in the MSF algorithm when a specific condition with the sensor property is present. We introduce related work in Chapter 2. We will briefly explain each sensor's characteristic and the Kalman filter used in the MSF in Chapter 3. The threat model is presented in Chapter 4. Next, we explain the attack methodology in Chapter 5 and evaluate the attack in Chapter 6. In particular, we use the Baidu Apollo [1] data set to evaluate our proposed attack methodology. In Chapter 7, we propose and discuss possible defenses to prevent such GPS spoofing attacks on AVs. We conclude the thesis and discuss the future work in Chapter 8.

CHAPTER 2. RELATED WORK

GPS spoofing attack. Although GPS spoofing attacks have been studied for a long time [8], it is still practical attack method nowadays. In GPS spoofing, the attacker fabricates or relays GPS satellite signals to inject a falsified location into the victim's GPS receiver. Generally, two steps are involved in a GPS spoofing attack [10]. The first step is commonly referred to as the *takeover* step. The attacker starts by sending fake signals which contain a location information similar to that of the legitimate GPS satellite signal. The attacker then gradually increases the strength of the fake signal and causes phase lock in the victim receiver that locks onto the fake signal instead of the legitimate signal. When the power of spoofed GPS becomes same as original satellite signal, the attacker can take over the GPS receiver from the original signal. In the second step, the attacker sends fake signals that indicate the location she defines to manipulate the victim receiver's location measurement. Prior work demonstrated that even high-end GPS receivers can be spoofed by a dedicated attacker [12].

Defense to GPS spoofing and its limitation. Due to the severity of GPS spoofing, various defense mechanisms are proposed [17, 5]. One of the defenses is using cryptographic mitigation techniques [9, 18]. This cryptographic mitigation can defend against a GPS spoofing attack. However, we have to modify both the satellite infrastructure and GPS receiver. Such modifications are unrealistic in the near future. Also, the computation overhead is another problem. The GPS receiver has to decrypt the encrypted signal. It will increase the computation time. Another possible solution is using signal-geometry based defense [5]. However, the cost of the AV would increase since at least three antennas are required to calculate the signal arriving angle. Also, the attacker can simply spoof from an aerial route using drones. So far, it is still an open question as to whether GPS spoofing attacks can affect the MSF algorithm adopted in AV localization.

Attacks built upon GPS spoofing. Recently, Zeng *et al*. [19] demonstrated that mobile navigation systems can be easily manipulated using GPS spoofing. However, this attack is unlikely to work in AVs because 1) LiDAR provides an accurate secondary localization source, and 2) the outlier detector can easily detect naive GPS spoofing attacks where the spoofed location deviates from the Kalman filter's prediction significantly. Narain *et al*. [10] leveraged the IMU bias and used GPS spoofing to fit within the common bias pattern to prevent the GPS signal from being detected by an Inertial Navigation System (INS) aided GPS tracker. However, they only considered a single localization source (GPS) for localization and the attacked INS system is not directly used for navigation.

To the best of our knowledge, this is the first work to study the security vulnerability of MSF-based localization algorithms in AVs.

CHAPTER 3. BACKGROUND

In this Chapter, we introduce different sensors used in the MSF algorithm. In particular, the state-of-the-art MSF algorithm uses the GPS, IMU, and LiDAR. The concepts of Kalman filter and outlier detection will also be explained.

3.1 Sensors in MSF

3.1.1 Global Positioning System

Global Positioning System (GPS) is the satellite-based navigation system used worldwide. Currently, there are more than 24 satellites in service. Each satellite broadcasts unique signals constantly to the earth. The GPS receiver collects the available satellite signals and calculates the distances to the source satellites of those signals based on time-of-flight. The GPS receiver then calculates its current location via triangulation, since the ground truth locations of the satellites are publicly known. Figure 3.1 illustrates how a GPS receiver in an AV resolves its location using four satellites.

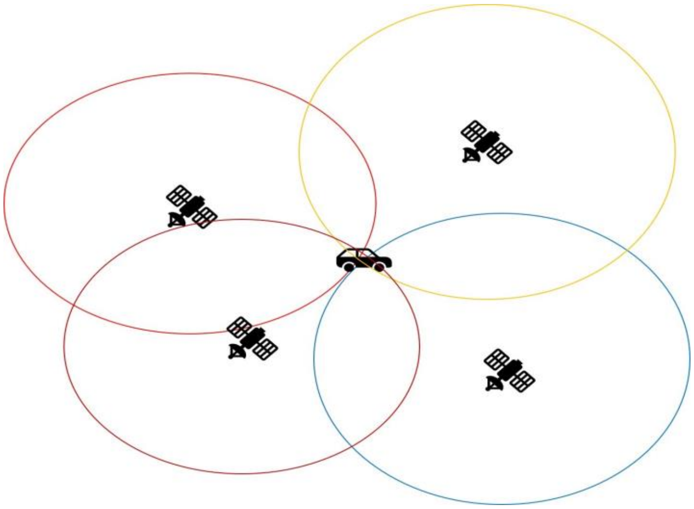

Figure 3.1: A GPS receiver in the AV resolves the location using four satellites

The GPS receivers resolve location information at a low frequency. Typical GPS frequency used in AV systems is 1 Hz [1]. GPS suffers from signal blockage, multi-path, and ionospheric delay during transmission. Despite the inaccuracy from those error sources, GPS provides some unique advantages in AV localization. The affordable price of GPS receivers is one advantage. Another advantage is that GPS can be used in harsh weather conditions (*e.g.*, rain or snow), while LiDAR generally performs poorly under such conditions.

3.1.2 Inertial Measurement Unit

An Inertial Measurement Unit (IMU) is a device that measures angular velocity and acceleration at a high frequency, typically in the range between 100 Hz to 200 Hz. Based on real-time angular velocity and acceleration, the position of an AV can be predicted via the kinematics model of the car. However, IMU readings often contain noises and biases because of sensor dynamics and the MEMS structural imperfection. To the best of our knowledge, there is no IMU that can achieve 100% accuracy of angular velocity and acceleration measurements. As a result, IMU biases accumulate over time without the correction from other sensors, and hence cause drift in localization estimations. Consequently, some location measurement sensors (*e.g.*, GPS and LiDAR) are used to correct localization estimation drift.

3.1.3 LiDAR

Light Detection and Ranging (LiDAR) sensor scans the surrounding environment of an AV. The ranging process starts by shooting lasers at different vertical and horizontal directions. When the lasers are blocked by obstacles, they are reflected back to the LiDAR. Based on the duration between shooting and receiving the lasers, LiDAR calculates the distances to the obstacles around the AV. Eq. 3.1 shows how to calculate the distance. The *time* represents the laser's traveling time and *c* is the speed of light ($2.99 * 10^8$ m/s).

$$distance = \frac{time \times c}{2} \tag{3.1}$$

The reflected lasers, which form a point cloud, effectively create a mapping of the surrounding area. As shown in Fig. 3.2, the LiDAR locator searches the point cloud in a high-definition map (HDMap) to find the best matching position and orientation. Such an HDMap is built offline before operating the AV. Currently, the HDMap can be provided either by the AV developing company (*e.g.*, Waymo and Baidu) or some mapping services (*e.g.*, DeepMap). When the weather is clear, LiDAR is generally much more reliable than GPS. Also, the LiDAR locator operates at a higher frequency (5 Hz in Apollo).

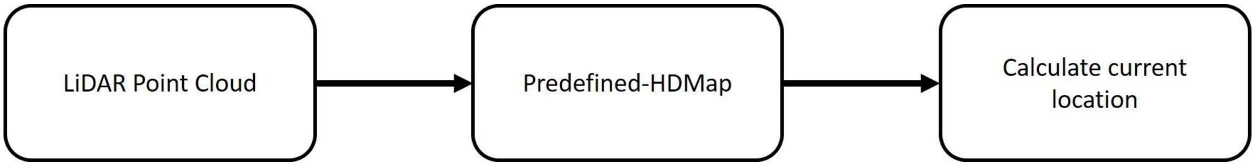

Figure 3.2: LiDAR process

3.2 Kalman filter and outlier detection

In MSF, the Kalman filter (KF) [18] is used for integrating different sensors. KF is commonly used for guidance, navigation, and control of vehicles. It defines a state model for the system (position, velocity, and attitude, *i.e.*, PVA). A KF operates recursively between two steps. The first step is the prediction. IMU's acceleration and angular velocity are integrated for predicting the next PVA state. The other step is the update, in which GPS and LiDAR update the state model asynchronously with the measurements of the PVA. Different kinds of kinematic models are used in both the prediction and update steps [13, 16].

Apart from the KF state estimation, outlier detection techniques (*e.g.*, Chi-squared test) are often used before feeding the measurements to the KF [3] in MSF. Such an outlier detector filters out abnormal measurements to prevent inaccurate sensor inputs. It increases the robustness of the result. The Chi-squared test is a widely used method for outlier detection in the KF [2, 14]. The Chi-squared test is a statistical method for determining the outlier measurement based on its Chi value, which reflects the measurement deviation from the state prediction. When the LiDAR and GPS try to update the KF, the KF will discard or perform a smaller update if the measurement Chi is larger than a statistical threshold (3.841 for the Chi distribution).

Fig. 3.3 shows an overview of the MSF components. With the outlier detector and multiple localization sources, the localization estimation from the MSF can be more robust. Consequently, it makes traditional sensor spoofing attacks hard to succeed.

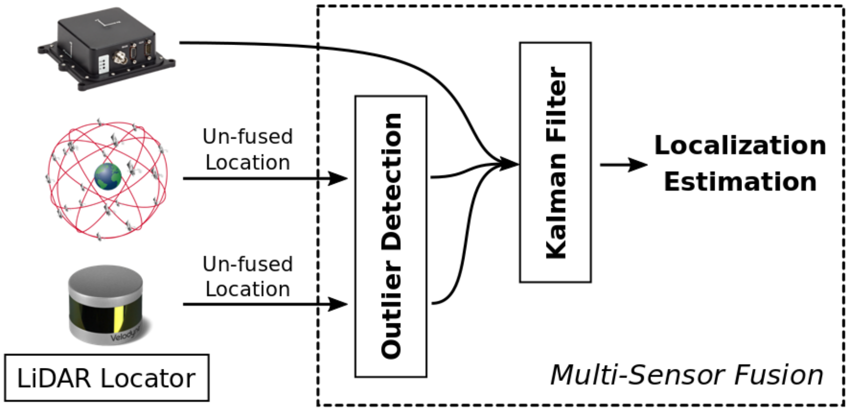

Figure 3.3: MSF algorithm using Kalman filter and outlier detection. The sensors on the left from top to bottom are IMU, GPS, and LIDAR

CHAPTER 4. THREAT MODEL

We assume that the attacker has a precise knowledge of the victim AV's real-time location. This can be achieved in two scenarios. The first scenario is when the victim AV is a taxi. The attacker can track the location of the AV by putting a suitcase that possesses the GPS device into the victim's trunk or backseat. To achieve this, the attacker simply takes the AV taxi and leaves the spoofer in the AV. In the second scenario, we assume that the attacker is driving a car and following a victim AV while launching the GPS spoofing attack. The attacker does not need to follow the victim AV closely. Zeng *et al*. [19] reported that the effective spoofing range can be as far as 50 meters. The attacker drives the car with the same speed as the victim's AV. Using the same speed is realistic because normally AVs drive at a constant speed and, nowadays, many cars are equipped with adaptive cruise control systems, which can maintain the same speed. After calculating the attacker car's current location, the attacker calculates the distance between the attacker's car and the victim AV. Affordable laser ranging sensors can be used for calculating distance precisely. To calculate the distance, Eq. 3.1 can be used again. In addition, we assume that the spoofed GPS signals can be as stable as natural signals. This is realistic since the spoofed signals are directly sent from the spoofer and thus, are free of the error sources that would occur in natural satellite signal propagation. We used the GPS data set provided by Apollo [1] and profiled the standard deviation of GPS. Then, we use the median value as the standard deviation for the spoofed GPS locations. Similar to prior work, we assume the attacker can spoof arbitrary locations to the victim's GPS receiver. Please note that we do not assume GPS spoofing can directly change the output of the MSF algorithm.

As the first step towards understanding the security properties, in this work, we assume that the attacker has the implementation of the MSF algorithm used in the victim AV. This is possible when 1) the victim AV adopts a representative algorithm implementation that is publicly available, such as Baidu Apollo MSF, or 2) the attacker owns an AV of the same model as the victim and can reverse engineer it to analyze the binary of the MSF. However, we do not assume the attacker can access the *real-time state* (*e.g.*, localization estimation in MSF) and sensor readings of the AV during the attack. As a result, the attacker has to infer the state during the attack.

The last assumption is that the attacker possesses the same IMU used in the victim AV. With the IMU, the attacker can collect IMU traces by driving the car offline. Meanwhile, the attacker measures the trajectory during the trace collection. The trajectory can be calculated precisely using some post-processing software such as GrafNav, which can reach < 5 cm positioning accuracy.

CHAPTER 5. METHODOLOGY

In this Chapter, we will present a systematic methodology for conducting security analysis on MSF designs.

5.1 Analysis Methodology Overview

At first, we explain the spoofing parameters, spoofing distance and spoofing degree, that are used in a GPS spoofing attack. By using the spoofing parameters, we can spoof a faked location to GPS. We apply exhaustively spoofing distance and spoofing degree to the data set provided by Baidu Apollo. That is, we apply every possible spoofing parameter to two GPS points. The purpose of this analysis is to try to see the MSF trend and compare the results between the two points. Based on the analysis, we find that even though we use the same spoofing distance, the deviation size is different. Since we align GPS close to LiDAR measurement location to get reliable result in this experiment, the IMU is likely the cause of the different size of the deviation. This phenomenon is called IMU drift. It is hard to predict IMU drift because of its nature. Because of its unpredictability, we propose practical attack design to filter out the point which has a large IMU drift. In our practical attack design, we introduce two attack parameters, initial spoofing distance and scaling factor. Initial spoofing distance is used for finding large IMU drift and scaling factor is used for making a larger deviation when large IMU drift is detected. To get a promising combination of initial spoofing distance and scaling factor, we use a profiling-based attack methodology. In the profiling, we apply every possible combination of two parameters and find the most common combination. After we get the most promising combination, we have to determine realistic goals that the attacker wants to achieve, such as the deviation size.

We will introduce spoofing parameters in §5.2. Next, we explain point-based analysis and its observation in §5.3. The IMU drift will be introduced in §5.4. Our practical attack design and profiling-based attack methodology are introduced in §5.5 and §5.6, respectively. Lastly, we introduce the goal of the attacker in §5.7.

5.2 Spoofing parameters and problem formulation

We define two spoofing parameters in GPS spoofing. The first parameter is the *spoofing degree* of the spoofed GPS location from the original GPS location. Another is the *incremental spoofing distance* along the degree. We represent parameter p_i as a pair (θ_i, D_i) for the spoofing point *i*. The θ_i represents the spoofing degree and D_i represents the incremental spoofing distance. The deviation dev_i presents the deviation size caused by spoofing point *i*. To calculate dev_i, it is defined as the lateral deviation of the fused location to the ground truth trajectory. Since we use the deviation size as lateral direction, we use θ_i as 90 or 270.

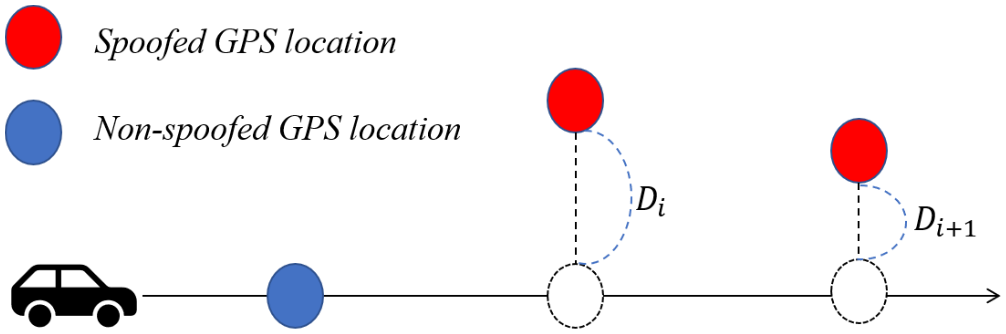

Figure 5.1: Description of spoofing distance *D* and degree *270 or 90*.

Fig. 5.1 shows the example of applying spoofing parameters. The *D* represents the spoofing distance. The two red circles are the spoofed GPS locations and the two dotted circles are the original non-spoofed GPS locations. As a result, using different spoofing distance D_i and D_{i+1}, the attacker can change the GPS location from the dotted circle to the red circles.

5.3 Point-based analysis by using spoofing parameters

AV systems make driving decisions by using real-world sensor data. So, we apply the spoofing parameters introduced in §5.2 to the real-world sensor data set provided by Baidu Apollo. Since the data set has noise, we align the GPS location close to the LiDAR measurement location. After every location of two sources are closely aligned, we choose two GPS data. We fix the spoofing degree as 270 and change the spoofing distance with fine granularity.

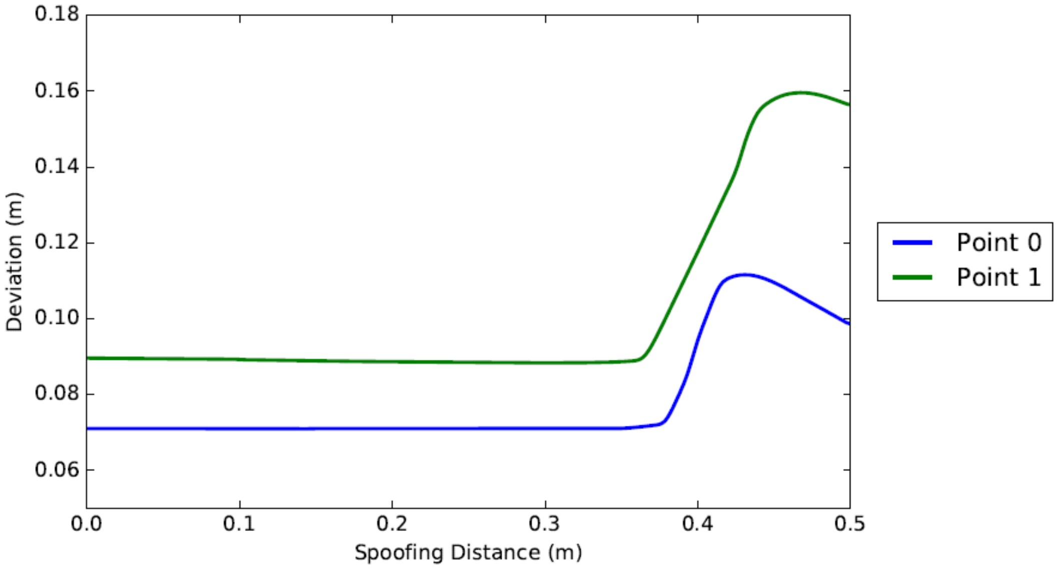

Figure 5.2: Deviation size on two points with different spoofing distance.

Fig. 5.2 shows the deviation size on two points with different spoofing distance. Based on Fig. 5.2, we can see that even though we use the same spoofing distance, the size of the deviation is different. For point 1, it achieves the highest deviation when the spoofing distance is close to 0.4m. But for point 2, it achieves the highest deviation when the spoofing distance is close to 0.45m. Since we align the GPS close to the LiDAR, the reason why such difference happens is very likely because of IMU. So, we will explain IMU drift as the reason of such difference in §5.4.

Another thing we have to analyze in Fig. 5.2 is when the spoofing distance is large. The deviation size decreases when the spoofing distance becomes larger than the specific distance. This specific distance varies on the point. For example, at point 1 in Fig. 5.2, the size of the deviation decreases when the spoofing distance is larger than 0.45m. Such decreasing happens because of an outlier detector. The outlier detector will filter out or make small updates to the Kalman filter when the spoofing distance is too large. As a result, the attacker has to choose the spoofing distance very carefully so as not to be detected by the outlier detector.

5.4 IMU drift

Inertial Measurement Unit (IMU) has a known property called drift, which is the difference between the real-world senor measurement and the ideal measurement. This is a result of the noises and biases in the raw IMU readings. For example, when the car drives with constant speed, the acceleration reading must be zero in an ideal case. However, in the real world, the acceleration reading is not exactly zero. This trend happens to the angular velocity as well. One source of the IMU bias is the manufacturing process. There are many related works to deal with IMU drift [7, 2]. In MSF, such IMU drift will be translated to the localization drift between the estimated location and the ground truth location. We find that this drift in localization estimation is a key for achieving large deviations in GPS spoofing. Since LiDAR localization is quite accurate in normal weather conditions and has been used as the ground truth in much researches, we measure the drift by calculating the distance between the localization estimation and the LiDAR location. We call this distance *LiDAR disagreement*. When the LiDAR disagreement is large, the Chi-value of the LiDAR becomes large. As a result, the LiDAR will provide a small update in the KF because of its large Chi-value. To show the LiDAR disagreement, we use the KAIST data set [6]. The KAIST data set provides the ground truth obtained by the LiDAR SLAM algorithm, so we set the LiDAR and the GPS location to the ground truth and compare the localization estimation from MSF to the LiDAR location.

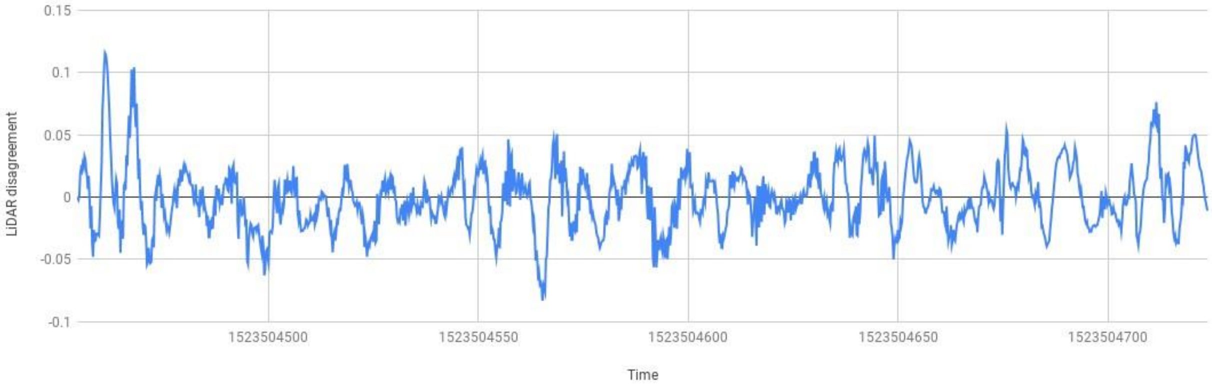

Figure 5.3: LiDAR disagreement in the KAIST dataset

Fig. 5.3 shows that the size of the LiDAR disagreement is large in some ranges. The spoofed GPS location is close to the localization estimation when the size of the LiDAR disagreement is large, and the direction is the same as the GPS spoofed location. This means that when the LiDAR disagreement increases, the distance between spoofed GPS and localization estimation decreases. Thus, the GPS can provide a larger update to the KF compared to the LiDAR because of the Chi-value. However, the IMU drift is unpredictable, so it is hard to calculate the exact timing of the GPS spoofing attack.

5.5 Practical Attack Design

Based on §5.3 and §5.4, the attacker has to choose the spoofing distance carefully. To achieve this, we introduce two parameters used in the attack. The first parameter of the attack is the *initial spoofing distance*. Whenever the attacker starts the GPS spoofing attack, the attacker changes the GPS location with a predefined initial spoofing distance with a spoofing degree 270 or 90. Since IMU drift is unpredictable, the attacker has to wait until the IMU drift happens. The initial spoofing distance helps the attacker predict when the IMU drift will happen. The valid initial spoofing distance can cause LiDAR disagreement to be larger. This is because when IMU drift causes a larger LiDAR disagreement, the spoofed GPS will be close to the localization result. Therefore, the initial spoofing distance helps the attacker filter out or detect the points that have a large IMU drift.

The second parameter is the *scaling factor*. The scaling factor applies to the spoofing distance.

$$\textit{spoofing distance} = \textit{previous spoofing distance} \times \textit{scaling factor} \tag{5.1}$$

Eq. 5.1 shows how to apply the scaling factor to the spoofing distance. The scaling factor is always larger than 1. The scaling factor is applied when the attacker discovers vulnerable points. We define vulnerable points as whenever the AV behaves incorrectly while applying the initial spoofing distance. We will define this odd behavior in §5.7. We use the scaling factor when the localization estimation is getting close to the spoofed GPS. It means that the distance between LiDAR and localization estimation is large and the localization estimation is closer to the spoofed GPS location. Therefore, we can increase the spoofing distance. However, the proper scaling factor has to be chosen such that the spoofed GPS will not be detected by the outlier detector and, at the same time, achieve a large measurement update.

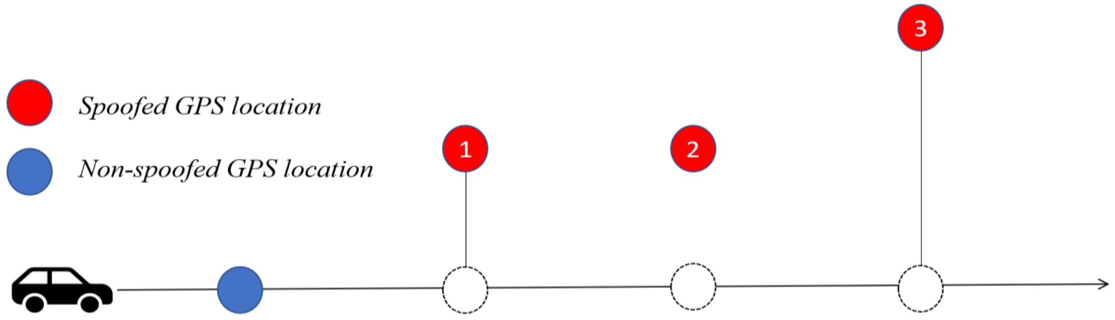

Figure 5.4: Examples for using initial spoofing distance and applying scaling factor

Fig. 5.4 represents how to apply the initial spoofing distance and scaling factor to the attack. The attacker starts the GPS spoofing on the red circle with number one. The attacker spoofs the GPS with the initial spoofing distance. Then, the attacker keeps using the same initial spoofing distance. Whenever the attacker discovers the AV's odd movement, he starts to apply the scaling factor to initial spoofing distance. In Fig. 5.4, the attacker discovers the

AV's odd behavior between red colored number 2 and 3 circles. Then, he applies the scaling factor to the red circle with number 3. He keeps applying the scaling factor. However, if the AV stops its invalid behavior or is going back to the original location, the attacker resets the scaling factor and restarts GPS spoofing with the initial spoofing distance. Fig. 5.5 shows an overview of the attack process.

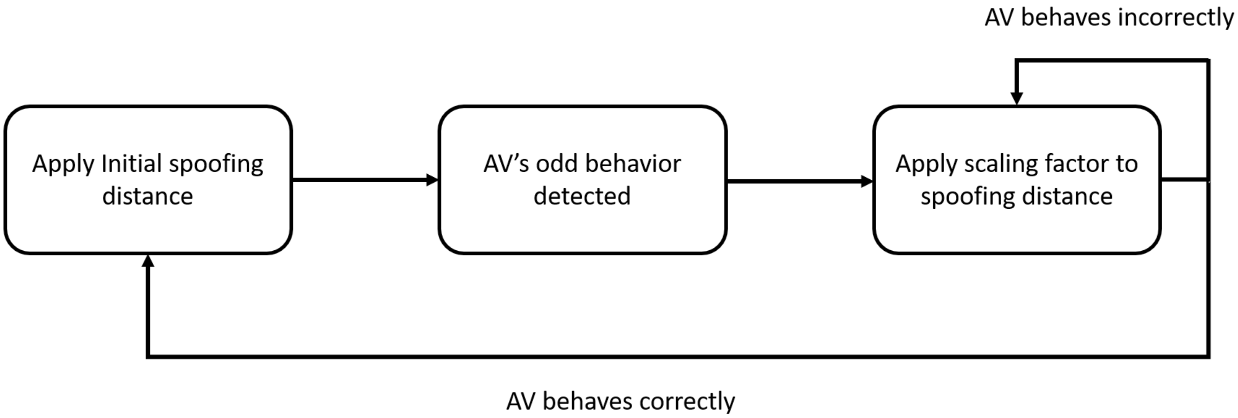
AV behaves incorrectly
Apply Initial spoofing distance
AV's odd behavior detected
Apply scaling factor to spoofing distance
AV behaves correctly

Figure 5.5: Control flow of using initial spoofing distance and scaling factor

The attacker keeps applying the scaling factor to the spoofing distance if the car goes in the wrong direction. Otherwise, the attacker stops applying the scaling factor and begins applying the initial spoofing distance again.

5.6 Offline profiling-based attack methodology

Based on §5.3 and §5.4, it is very important to choose the proper initial spoofing distance and scaling factor. If not, the location of the spoofed GPS will be considered as an outlier and LiDAR will correct the localization estimation. Due to the nature of IMU bias and noise, the drift in MSF is hard to predict. Thus, we propose a profiling-based attack method to predict the scaling factor and initial spoofing distance. The intuition is that the similar drift, which appears in the profiling segment, will likely appear in other segments as well. We apply every possible initial spoofing distance and scaling factor to the profiling segment. From the offline profiling, we obtain the most common spoofing parameters, initial spoofing distance and scaling factor, which can cause large deviation. In this process, we have to decide the profiling duration.

The attacker can conduct a profiling-based attack using his own IMU sensor. In Chapter 4, the attacker can collect IMU measurement data with the car's trajectory. Hence, the attacker can profile and find the initial spoofing distance and scaling factor from the collected data.

5.7 The attacker's goal

The attacker's goal is to make a larger deviation than the threshold from the ground truth. We define this threshold as the *desired deviation*. The possible desired deviation moves the AV to the adjacent lane or disturbs other cars driving in the adjacent lane. In this case, the attacker has to consider the lane width, AV's width and, other car's width. From the U.S Transportation Department, the width of the typical lane is 2.7m and the width of the highway is 3.6m. The width of an AV varies depending on the model of the car. In Baidu Apollo, they use the 2019 Lincoln MKZ model as an AV and its width is 2.11m. For bothering the car, the AV partially goes to the next lane and causes a crash. In this case, we assume that the car in the next lane drives very close to the line of the lane to avoid being hitted by the AV. The average width of car is 1.905m. So, the way to calculate the desired deviation for making a crash is following Eq. 5.2. As a result, when the AV's width is 2.11m, the car's width is 1.905m and the lane width is 2.7m, the desired deviation is 1.09m. Also, the attacker has to choose the duration. It means that the attacker achieves the desired deviation in the defined duration.

$$desired\ deviation\ to\ cause\ crash = (lane\ width - car's\ width) + \frac{lane\ width - AV's\ width}{2} \tag{5.2}$$

The last thing the attacker has to define is the pattern of the *AV's odd behavior or movement*. In §5.5, the attacker applies the scaling factor when the AV moves incorrectly. Such an odd movement is defined differently based on the AV system. However, the AV follows the center of the lane in a normal case. The attacker can observe the AV when it touches the line of the lane. Hence, the attacker can calculate the minimum deviation that makes the car move towards the boundary of the lane. We define this minimum deviation as the *observation deviation threshold*.

$$observation\ deviation\ threshold = \frac{lane\ width - AV's\ width}{2} \tag{5.3}$$

Eq. 5.3 shows how to calculate the observation deviation threshold. For example, if the AV's model is 2019 Lincoln MKZ and drives in the typical road that the width is 2.7m, the observation deviation threshold is 0.295m. So, when the attacker moves the AV to 0.295m, the AV will touch the line of the lane and the attacker starts to multiply the scaling factor to the initial spoofing distance, or previous spoofing distance.

CHAPTER 6. ATTACK EVALUATION

We use the production-grade open-source AV system, Baidu Apollo, in our analysis. In Apollo, MSF is implemented by the error-state Kalman filter, which fuses the sensor data from IMU, GPS, and LiDAR. In our evaluation, we use the MSF sensor trace data provided by Baidu. Baidu provides this MSF sensor trace data for testing the localization module so that the data is much reliable and accurate than other data sets. And our attack goal is that we achieve a deviation to specific threshold in the duration defined by the attacker.

At first, we will introduce the way to calculate the ground truth on Apollo's MSF sensor data set. Next, we will show the result of offline profiling. Lastly, we will evaluate our attack method by applying the most common combination of the initial spoofing distance and scaling factor.

6.1 Ground truth for evaluating the attack

The sensor trace data set provided by Baidu is approximately 3 minutes (exactly 219 seconds). We apply every possible combination of the initial spoofing distance and scaling factor to every segment. In this case, we vary the time to start applying the initial spoofing distance, the time to apply the scaling factor and initial spoofing distance. One segment consists of duration defined by the attacker. When at least one combination of the initial spoofing distance and scaling factor makes deviation, which is larger than the desired deviation threshold (*e.g.*, 2.7m and 1.09m), we define the segment as a vulnerable segment.

We use 4 different durations, 60, 90, 120, and 150 seconds. Table 6.3 shows the number of segments on the different duration. We calculate the ground truth for all segments except the segment that the starting point is included in the offline profiling segment introduced in §6.2. After we apply every possible combination to every segment on different duration, we find that every segment is vulnerable to the attack with the specific initial spoofing distance and scaling factor.

6.2 Offline profiling result

We choose one segment from the Apollo MSF sensor trace data set. This segment consists of 1 minute. The driving scenario of the segment we chose is that the car drives in a curvy road, which is a very common case in the real world. We apply every possible combination of the initial spoofing distance, scaling factor and vary the time to start spoofing. Since it is 60 seconds duration, the attacker has 59 choices to choose the time to apply the scaling factor. The range of the initial spoofing distance we used is from 0.1m to 2.0m with 0.1m granularity. Also, the range of the scaling factor is from 1.1 to 2.0 with 0.1 granularity. Next, we calculate the frequency of the combination of the scaling factor and initial spoofing distance that achieves a larger deviation than the desired deviation threshold (e.g.2.7m and 1.09m). As a result, the possible maximum frequency of a combination is 59 and the minimum frequency is 0. If the frequency is 59, it means the attacker can always achieve the desired deviation threshold when the attacker applies specific initial spoofing distance and scaling factor regardless of the time applying the scaling factor.

Table 6.1 and Table 6.2 show the profiling results with different desired deviation threshold. We set two desired deviation thresholds, 2.7m and 1.09m. Based on profiling results, 0.6m initial distance and 1.4, 1.3 scaling factor are the most common combinations. Also, 0.5m initial distance and 1.4 scaling factor is the common combination from the profiling result.

Initial distance (m)	Scaling factor	Frequency
0.6	1.3	30
0.6	1.4	23
0.6	1.2	20
0.6	1.5	18
0.5	1.4	15
0.5	1.5	15

Table 6.1: The result of offline profiling when desired deviation threshold is 1.09m

Initial distance (m)	Scaling factor	Frequency
0.6	1.4	13
0.6	1.3	13
0.5	1.4	11
0.6	1.2	11
1.0	1.3	10
1.1	1.2	10

Table 6.2: The result of offline profiling when desired deviation threshold is 2.7m

6.3 Attack evaluation

Based on the most frequent combination of the initial spoofing distance and scaling factor calculated on offline profiling, we apply those combinations to the Apollo MSF sensor trace data set. We vary the duration of the attack and exclude the profiling segment in the data set for the evaluation. The durations we used in the attack evaluation are 60, 90, 120, and 150 seconds. Also, we set the observation deviation threshold as 0.295m.

Table 6.3 shows the number of segments based on different durations. It means there are 99 segments the attacker can explore when the attack duration is 60 seconds. When the duration is 150 seconds, the number of segments is small because the data set we used is approximately 3 minutes (exactly 219 seconds). Then, there are 69 segments, but we have to exclude segments with starting time overlapped with the profiling period. That is why there are only 29 segments left.

Duration (s)	Number of segments
60	99
90	69
120	39
150	29

Table 6.3: The number of segments based on the different duration

We apply three different combinations of attack parameters, the initial spoofing distance and scaling factor; 1) initial spoofing distance: 0.6m, scaling factor: 1.3, 2) initial spoofing distance: 0.6m, scaling factor: 1.4, 3) initial spoofing distance: 0.5m, scaling factor: 1.4. Those combinations are most common cases from the profiling result when the desired deviation is 2.7m and 1.09m.

Table 6.4 and 6.5 show the success rate on different desired deviations. Table 6.5 is the result of success rate from 1.09m desired deviation and Table 6.4 shows the result of success rate when desired deviation is 2.7m. As the attack duration increases, the success rate increases as well. The combination of 0.6m initial spoofing distance and 1.4 scaling factor has larger success rate than others based on the result of success rate. The success rate in Table 6.5 is larger than Table 6.4 because the desired deviation is smaller. However, the difference of success rate is trivial. That is, when the attacker achieves 1.09m deviation, he can easily achieve 2.7m deviation as well. In both 2.7m and 1.09m desired deviation cases, the attack achieves 100% success rate when the duration becomes 150 seconds.

Duration (s)	Initial distance (m)	Scaling factor	Success rate (%)
60	0.6	1.3	45.45
	0.6	1.4	45.45
	0.5	1.4	44.44
90	0.6	1.3	62.31
	0.6	1.4	62.31
	0.5	1.4	60.86
120	0.6	1.3	64.1
	0.6	1.4	66.66
	0.5	1.4	66.66
150	0.6	1.3	100
	0.6	1.4	100
	0.5	1.4	100

Table 6.4: The success rate with different attack parameters. The desired deviation is 2.7 m.

Duration (s)	Initial distance (m)	Scaling factor	Success rate (%)
60	0.6	1.3	45.45
	0.6	1.4	47.47
	0.5	1.4	45.45
90	0.6	1.3	62.31
	0.6	1.4	65.21
	0.5	1.4	62.31
120	0.6	1.3	69.23
	0.6	1.4	76.92
	0.5	1.4	74.35
150	0.6	1.3	100
	0.6	1.4	100
	0.5	1.4	100

Table 6.5: The success rate with different attack parameters. The desired deviation is 1.09 m.

However, when the duration is 60 seconds, it has lower success rate which is slightly larger than 45%. But, in §6.1, we showed that all segments we explored are vulnerable to the attack. We try to find the reason why our attack cannot achieve higher success rate in 60 seconds duration. We apply different possible parameter combinations to the failed segment. Each segment has its own characteristics because the degree of IMU drift is usually unique. That is, the combination of initial spoofing distance and scaling factor that makes larger deviation in the vulnerable segment varies between segments. For example, the attacker has to use 0.9m initial distance to attack the specific segment. As a result, the combinations used in the evaluation cannot make the segment being attacked. But, when the duration is getting longer, it normally covers the segment which is attack-able by attack parameters we used in Table 6.4 and 6.5. It means that similar IMU drift, which is explored in profiling segment appears frequently.

CHAPTER 7. DISCUSSION OF THE POTENTIAL DEFENSES

There are possible ways to prevent GPS spoofing attack on the localization module of AV. The first possible defense is using the other car's location or infrastructure's location. In the near future, it will be possible for an AV to communicate with other vehicles including AVs and normal cars. It is called vehicle-to-vehicle communication. Also, the communication between the infrastructure and AV can be used. The infrastructure includes traffic signal. It is called vehicle-to-infrastructure communication. Based on this communication, the AV can receive the location of another car or infrastructure. The AV can compare its own perception result and the received location from the communication. It can detect the problem of localization since the perception result comes from localization estimation. So, if incompatibility exists between perception or localization estimation and the received location, the AV can detect that an attack happens on the localization module.

The second possible solution makes MSF algorithm to trust more on LiDAR than GPS. Attacking LiDAR is more difficult compared to GPS in the current stage. So, when incompatibility between GPS and LiDAR exists, trusting more on LiDAR will render the GPS spoofing attack ineffective. That is, LiDAR becomes the main source of localization and GPS is used for compensating the LiDAR. However, when the attacker can achieve a deliberate attack on LiDAR, such defense method will be ineffective. To the best of my knowledge, attacking moving LiDAR elaborately is hard to achieve in the current stage.

The last possible solution is about the IMU. When the IMU has no bias and drift, it can automatically solve the GPS spoofing attack. In current state, it is impossible to make completely perfect IMU. But, in the near future, the perfect IMU could be made and it would prevent GPS spoofing attacks. When the IMU becomes perfect, there won't be larger LiDAR disagreement, which is the main cause of GPS spoofing attack.

CHAPTER 8. CONCLUSION AND FUTURE WORK

To calculate the current location of AV and robot is challenging. This task is implemented by using different kinds of sensors. This process is the basic function for making the AV secure. To achieve this functionality, AV uses MSF algorithm which fuses different sensors such as GPS, IMU, and LiDAR. However, GPS, a source of localization, is vulnerable to spoofing attacks. Based on our analysis, the GPS spoofing attack still makes localization precariously. Even though it is difficult for the attacker to attack MSF using GPS spoofing, the attack is still valid. We also find a possible reason why the GPS spoofing attack still works on MSF. It is very likely because of IMU drift. We apply our attack methodology to data set provided by Baidu Apollo. We achieved 100% success rate within 150 seconds. Also, we make a deviation which is larger than 2.7m within 2 minutes with 66% success rate.

Because of the lack of time, we evaluate our methodology on only one data set. In the future, we will apply our most common attack parameters in Table 6.1 to different data sets such as KAIST [6] and KITTI [4]. After that, we will calculate the success rate on those data sets and evaluate our attack more precisely.

Also, we will choose different segments to get a profiling result and compare the result to our current result. Since each segment has its own characteristics, profiling results could differ between segments. After getting a new profiling result, we will apply it to data sets.

Apollo's MSF localization is a closed source so that there is a difficulty on analyzing the cause why MSF is affected by GPS spoofing attacks. We will do reverse engineering on Apollo's MSF algorithm to do cause analysis. If we can know the direct cause why the GPS spoofing attack works on MSF, we can devise the advanced attack method and achieve higher success rate. We will run different experiments to verify the reason.

Finally, we will explore other possible attack vectors. Different kinds of attack exist in each sensor such as the IMU and LiDAR. Even though they need complicated attack methods than GPS, attacks on the IMU and LiDAR can cause huge effect on MSF because IMU and LiDAR make greater data contribution than GPS.

REFERENCES

[1] Baidu Apollo team (2017), Apollo: Open Source Autonomous Driving, howpublished = https://github.com/apolloauto/apollo, note = Accessed: 2019-02-11.

[2] J. Borenstein, L. Ojeda, and S. Kwanmuang. Heuristic reduction of gyro drift in imubased personnel tracking systems. In *Optics and Photonics in Global Homeland Security V and Biometric Technology for Human Identification VI*, volume 7306, page 73061H. International Society for Optics and Photonics, 2009.

[3] B. Brumback and M. Srinath. A chi-square test for fault-detection in kalman filters. *IEEE Transactions on Automatic Control*, 32(6):552–554, 1987.

[4] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013.

[5] K. Jansen, M. Scha¨fer, D. Moser, V. Lenders, C. Po¨pper, and J. Schmitt. Crowd-gpssec: Leveraging crowdsourcing to detect and localize gps spoofing attacks. In *2018 IEEE Symposium on Security and Privacy (SP)*, pages 1018–1031. IEEE, 2018.

[6] J. Jeong, Y. Cho, Y.-S. Shin, H. Roh, and A. Kim. Complex urban dataset with multilevel sensors from highly diverse urban environments. In *The International Journal of Robotics Research*, 2019.

[7] A. R. Jim´enez, F. Seco, J. C. Prieto, and J. Guevara. Indoor pedestrian navigation using an ins/ekf framework for yaw drift reduction and a foot-mounted imu. In *2010 7th Workshop on Positioning, Navigation and Communication*, pages 135–143. IEEE, 2010.

[8] A. J. Kerns, D. P. Shepard, J. A. Bhatti, and T. E. Humphreys. Unmanned aircraft capture and control via gps spoofing. *Journal of Field Robotics*, 31(4):617–636, 2014.

[9] M. G. Kuhn. An asymmetric security mechanism for navigation signals. In *International Workshop on Information Hiding*, pages 239–252. Springer, 2004.

[10] S. Narain, A. Ranganathan, and G. Noubir. Security of gps/ins based on-road location tracking systems. *arXiv preprint arXiv:1808.03515*, 2018.

[11] J. Petit, B. Stottelaar, M. Feiri, and F. Kargl. Remote attacks on automated vehicles sensors: Experiments on camera and lidar. *Black Hat Europe*, 11:2015, 2015.

[12] J. Saarinen. Students hijack luxury yacht with gps spoofing. *Secure Business Intelligence Magazine*, 2013.

[13] S.-L. Sun and Z.-L. Deng. Multi-sensor optimal information fusion kalman filter. *Automatica*, 40(6):1017–1023, 2004.

[14] N. Thacker and A. Lacey. Tutorial: The kalman filter. *Imaging Science and Biomedical Engineering Division, Medical School, University of Manchester*, page 61, 1998.

[15] Y. Tu, Z. Lin, I. Lee, and X. Hei. Injected and delivered: fabricating implicit control over actuation systems by spoofing inertial sensors. In *27th {USENIX} Security Symposium ({USENIX} Security 18)*, pages 1545–1562, 2018.

[16] G. Wan, X. Yang, R. Cai, H. Li, Y. Zhou, H. Wang, and S. 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.

[17] J. S. Warner and R. G. Johnston. Gps spoofing countermeasures. *Homeland Security Journal*, 25(2):19–27, 2003.

[18] G. Welch, G. Bishop, et al. An introduction to the kalman filter. 1995.

[19] K. C. Zeng, S. Liu, Y. Shu, D. Wang, H. Li, Y. Dou, G. Wang, and Y. Yang. All your GPS are belong to us: Towards stealthy manipulation of road navigation systems. In *27th USENIX Security Symposium (USENIX Security 18)*, pages 1527–1544, 2018.