Analytical robustness assessment for sensor fusion module of self-driving cars

Jin Ding
Department of Computer Engineering
University of Virginia
Charlottesville, Virginia 22903
Email: jd7jz@virginia.edu

Jiefu Liang
Department of Computer Science
University of Virginia
Charlottesville, Virginia 22903
Email: jl2dz@virginia.edu

Xin Wen
Department of Computer Science
University of Virginia
Charlottesville, Virginia 22903
Email: xw8ng@virginia.edu

***Abstract*—Artificial intelligence has been developing fast since ten years ago. Among the branches of artificial intelligence, the innovation of self-driving cars make a great contribution to push forward the development of artificial intelligence. Self-driving cars probably will be widely applied to the market in a few years later. Hence, security issues regrading self-driving cars have been proposed by scientific researchers during past several years. Among the security issues proposed, potential security problems of sensor fusion module are easily ignored. This module plays an important part in collecting data from various sensors and processing data of different formats. In this paper, we proposed to test the robustness for sensor fusion module of self-driving cars system. We designed an experiment by changing input data with different steps, magnitude and sensors to verify whether the module can defense the different attacks.**

Keywords*—*self-driving cars, sensor fusion, robustness, assessment.

I. Introduction

Nowadays, self-driving car is becoming a potential transportation tool in the future. Shenzhen has started to use self-driving car as bus used in Futian tax-free district. Even though self-driving car has a pile of advantage compared with traditional transportation tools, we still need to take care of its potential risk, especially security risk. There are several different security risks in self-driving car. For example, risk of using fake training dataset to train its control engine, risk of using physical methods to block self-driving cars system sensors like Lidar, Radar and camera and risk of breaking down control system to make it lose control of the self-driving car. In this paper, we focus on the security assessment for sensor fusion module of self-driving cars. By conducting several experiments of changing the input of different sensors, we observed the lag effect of changing Lidars input value and Radars input value and changing the two sensors input together. Chapter 2 states the related work for security issues of self-driving cars. Chapter 3 introduces the operating principle for sensor fusion module. Chapter 4 and 5 introduce the experiment idea and present experiment result. In the last chapter, we analyze the experiment results and propose several potential attacks.

II. Related work

Generally, the system of self-driving cars consists of three phases: *Sense*, *Understand* and *Act* [2]. A set of sensors will be used to sense surrounding environment of the automated vehicles. These sensors can be termed as the interfaces between outside environment and self-driving cars' system. These devices will provide the raw collecting data to the *Understand* phase. Then the *Understand* phase constructs a representation of the environment by fusing and processing raw sensors data. Finally, the *Action* phase instruct self-driving cars to take correct operations.

Since a self-driving car will unconditionally rely on sensors to make short-term or long-term driving decisions. Under this situation, sensors play a significant role in the whole system. At the same time, since camera and Lidar can be used in laboratory environment for controlled experiments, without being integrated in an actual vehicle, Petit et al [2] proposed a remote attacking method on camera and Lidar of automated vehicle sensors. Actually, the reason why investigations focus on Sense phase is researchers think sensor fusion algorithms cannot process raw sensor data properly and effectively [1].

Sun and Deng [3] proposed a general multi-sensor optimal information fusion decentralized Kalman filter with a two-layer fusion structure. It is for discrete time linear stochastic control systems with multiple sensors and correlated noises.

III. Sensor Fusion

The target system we aim at is a Udacity course project demo[1]. This project includes a whole sensor fusion module which combines the input data from lidar with radar. The core algorithm of sensor fusion module is Kalman filter. In this system, sensor fusion module mainly focuses on predicting the position and velocity of pedestrians. Figure1 shows the features of pedestrians which will be estimated by sensor fusion module.

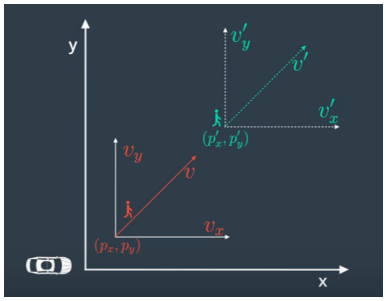

Fig. 1: Functionality of sensor fusion

[1]Github link:https://github.com/jessicayung/self-driving-car-nd

When sensor fusion module firstly receives the input data from lidar or radar, it will initialize the state of pedestrians and covariance matrices based on the first input data. In the following prediction phase, the sensor fusion module can separately set up covariance matrices for lidar and radar, at the same time predict the state of pedestrians based on the corresponding input data and covariance matrices. Finally the pedestrians' state will be updated with the new prediction. Figure2 shows the processing flow of the sensor fusion module.

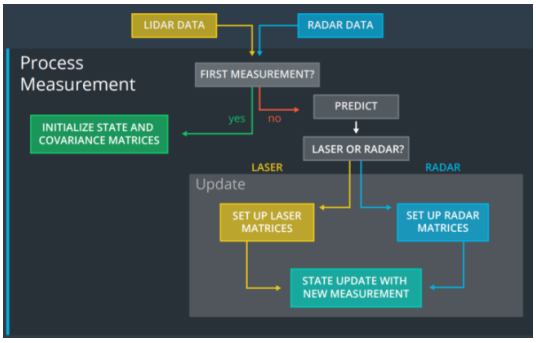

Fig. 2: Processing Flow

In this sensor fusion module, each sensor has its own prediction and update scheme. With multiple sensors, state of pedestrians and prediction schemes of sensors are updated asynchronously. In figure 3, we have probability distribution mean X_k and covariance matrix P_k at time k. At time $k+1$, the sensor fusion module computes a new measurement based on previous state at time k. At the same time, the sensor fusion module combines the prediction state with the input data from lidar to compute a new measurement state X_{k+1} and P_{k+1}. At time $k+2$, another new measurement X_{k+2} and P_{k+2} can be predicted by same computation methods. At time $k+3$, the sensor fusion module combines the state $k+1$ with state $k+2$ to generate a measurement state X_{k+3} and P_{k+3}.

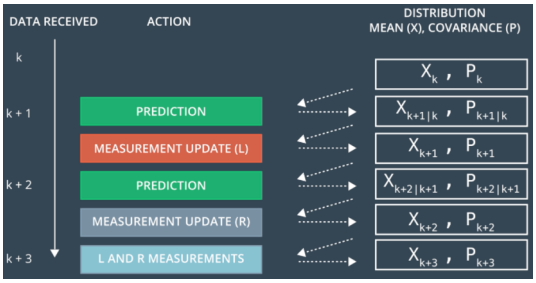

Fig. 3: Asynchronously Processing Flow

The above materials about Kalman filter and the operating principle can be found at here[2].

IV. Experiment Design

A. Format of input data

The input data format of lidar and radar is slightly different. TableI partly shows the parameters and format for lidar data. TableII partly shows the parameters and format for radar data.

[2]https://github.com/jessicayung/self-driving-car-nd/tree/master/term-2

meas_{px}	means_{py}	timestamp	...
8.45	0.25	1477010443349642	...

TABLE I: Format of Lidar Data

In tableI, $meas_{px}$ means the distance between cars and pedestrians in x direction. $meas_p$ means the distance between cars and pedestrians in y direction.

meas_{rho}	meas_{phi}	meas_{rhodot}	timestamp	...
8.60363	0.0290616	-2.99903	1477010443399637	...

TABLE II: Format of Radar Data

In tableII, the data is expressed in polar coordinates. $meas_{rho}$ means the distance between the cars and pedestrians. $meas_{phi}$ means the polar angle between the cars and pedestrians. In our experiments, we mainly focus on testing a lag effect of sensor fusion module by changing the $meas_{px}$, $meas_{py}$ and $meas_{rho}$ three parameters.

B. Experiment Design

In the experiment, we set three variables: magnitude, steps and sensor type. Magnitude has three levels: 1, 2 and 5, which means the number we will add to the original input data. Steps also three levels: 10, 100 and 300, which means the lines number of input data we want to modify. Sensor type has three levels: Lidar(L), Radar(R), Lidar and Radar(LR), which means we will change the input data of the sensor of this type. Here if we choose magnitude to 1, steps to 100, sensor type to R, in this experiment we will add the radar's input data parameters $meas_{rho}$ to 1, in the first 100 lines of input data.

The method to test the robustness of the sensor fusion module is to quantify the lag effect. One of the most important features of Kalman filter is its robustness. In other words, even though we modify some input data of lidar or radar, after a period of time Kalman filter algorithms can still accurately predict the state of pedestrian based on other un-modified input data. And the length of this period of recovering time can help us test the robustness of the sensor fusion module. Here we convert this period of recovering time to the number of lines. If we change the first 100 lines of input data, we will compute in the following how many lines the sensor fusion module will inaccurately predict the state of pedestrians. And we define a threshold as 0.01, which means if the difference between prediction results of modified input data and the previous original prediction results exceeds 0.01, we think this prediction result is inaccurate. Finally, we show the robustness of the sensor fusion module based on the quantified lag effect (number of lines which sensor fusion make inaccurate predictions).

V. Experiment Results

A. Experiment Results

In the experiment, we mainly use the method of controlling variables. Firstly we fix the variable step to 100, then change the sensor type to L, R and LR, the magnitude to 1, 2, 5. Therefore, we get 9 experiment results, shown in the table III.

Then we fix the variable magnitude to 1, then change the sensor type to L, R and LR, the step to 10, 100, 300. We can get 9 experiment results, shown in the table IV.

Sensor \ Magnitude	1	2	5
L	34	40	46
R	22	29	35
LR	31	38	45

TABLE III: Fixed Step: 100

Sensor \ Step	10	100	300
L	18	34	28
R	20	22	22
LR	26	31	27

TABLE IV: Fixed Magnitude: 1

B. Result Analysis

Based on the experiments shown above, we find as the magnitude increases, the lag effect increases obviously. Take the first line of table III for instance, when magnitude is 1, 2 and 5, the number of lines for inaccurate predictions are 34, 40, 46, respectively. we also find the variable step can also have a influence on the lag effect. Take the first line of TableIV for instance, as the step going up, the sensor fusion module can recover from slowly to quickly. When the step is 10, 100 and 300, the lines number of inaccurate predictions are 18, 34 and 28. Obviously, the lag effect is better when the step is 100, rather than 300.

Apart from the above two findings, we also find the lag effect will be more significant if we only modify the input data of L, compared with that of L and LR. Take the first column of table III for instance, when the sensor type is L, R and LR, the number of lines for inaccurate predictions are 34, 22 and 31. We can reach a conclusion that the input data of lidar probably have more influence on the computation of the sensor fusion module. However, it is hard to explain this weird experiment results because in theory modifying the input data of LR is supposed to have more influence on sensor fusion than modifying the input data of lidar alone. Here we guess the reason is the input data of radar contains a directional information which we did not extract in our experiments. In other words, when we modify both the input data of lidar and radar, probably the directional parameters of radar will weaken the influence even reverse the influence, which could lead the sensor fusion module to recover more quickly than we think it should be.

C. Potential Attacks

Here we propose several potential attacks for the sensor fusion module, in perspective of maximizing the lag effects and minimizing the cost. Here are three potential attacks:

- Periodically attack (every ten lines)
- Increase magnitude
- Modify lidar's input data alone

The reason why we recommend modify the input data in a period of ten lines is that it will cost less if we periodically attack the sensor fusion, but we can still get a good lag effect. The lag effect for attacking sensor fusion every ten lines for ten times is better than that for attacking sensor fusion by 100 lines once. For the remaining two potential attacks, we can easily reach the conclusions from the previous experiment results.

VI. Conclusion

In this paper, we proposed to test the robustness of sensor fusion module of self-driving cars. We focused on the project demo on Udacity self-driving car course and Baidu Apollo version 1.0 and 1.5. We find Baidu Apollo demo is a more complete self-driving system which contains a UI interface and corresponding hardware. However, the Apollo platform only contains a perception module but no sensor fusion module. Therefore, we only conduct experiments on Udacity course project demo. In the future, researchers can try the similar experiments on Baidu Apollo Version 2.0 and we think the experiment results will be more persuasive. In our experiment, we design a control-variable experiment and define a series of variables, as well as the quantified lag effect. Finally, based on the experiment results, we proposed three potential attacks for sensor fusion in perspective of maximizing lag effect and minimizing attacking cost. In the future work, we plan to extract the directional information of radar input data and apply our experiments on Baidu Apollo 2.0.

References

[1] Marcus Obst, Laurens Hobert, and Pierre Reisdorf. Multi-sensor data fusion for checking plausibility of v2v communications by vision-based multiple-object tracking. In *Vehicular Networking Conference (VNC), 2014 IEEE*, pages 143–150. IEEE, 2014.

[2] Jonathan Petit, Bas Stottelaar, Michael Feiri, and Frank Kargl. Remote attacks on automated vehicles sensors: Experiments on camera and lidar. *Black Hat Europe*, 11:2015, 2015.

[3] Shu-Li Sun and Zi-Li Deng. Multi-sensor optimal information fusion kalman filter. *Automatica*, 40(6):1017–1023, 2004.