

**NUMERICAL INVESTIGATION OF AFFORDABLE 2D LIDAR  
SENSORS FOR AUTONOMOUS DRIVING UNDER A SPECIALLY-  
STRUCTURED DRIVING ENVIRONMENT**

A Dissertation  
Presented to  
The Academic Faculty

By

Xiaoshu Liu

In Partial Fulfillment  
Of the Requirements for the Degree  
Master of Science in Mechanical Engineering

Georgia Institute of Technology  
December of 2017

Copyright © Xiaoshu Liu 2017

**NUMERICAL INVESTIGATION OF AFFORDABLE 2D LIDAR  
SENSORS FOR AUTONOMOUS DRIVING UNDER A SPECIALLY-  
STRUCTURED DRIVING ENVIRONMENT**

Approved by:

Dr. William Singhose, Advisor  
School of Mechanical Engineering  
*Georgia Institute of Technology*

Dr. Oliver Sawodny  
Institut für Systemdynamik  
*Universität Stuttgart*

Dr. Kok-Meng Lee  
School of Mechanical Engineering  
*Georgia Institute of Technology*

Dr. Christina Tarin  
Institut für Systemdynamik  
*Universität Stuttgart*

Date Approved: September 11, 2017

To my parents

## ACKNOWLEDGEMENTS

First I would like to thank Dr. William Singhose for his advising at Georgia Tech; Mr. Bernd Schuler for his guidance at Daimler; and Mr. Florian Morlock for offering advice on my thesis and providing help throughout my stay in Stuttgart. I would also like to thank Dr. Paul Neitzel and Dr. Oliver Sawodny for making this joint degree program possible. A big thank you to my parents who have supported me from two different continents. Last, but not least, I would like to thank my friends who have filled these last two years with joy and laughter - Ahmed, Benni, Caroline, Chris, Eui, Marius, Thomas, and especially Wenjia.

# TABLE OF CONTENTS

<b>ACKNOWLEDGEMENTS</b>	<b>iv</b>
<b>LIST OF TABLES</b>	<b>vii</b>
<b>LIST OF FIGURES</b>	<b>viii</b>
<b>SUMMARY</b>	<b>x</b>
<b>I. INTRODUCTION</b>	<b>1</b>
1.1 Motivation	1
1.2 Sensors for Autonomous Driving	2
1.3 Light Detection and Ranging (Lidar)	3
1.4 Lidar Model of Interest	7
1.5 Specially-Structured Driving Environment	9
1.6 Thesis Outline	10
<b>II. BACKGROUND</b>	<b>11</b>
2.1 Background on Light	11
2.1.1 Intensity attenuation	11
2.1.2 Laser class 1	11
2.1.3 Reflectivity	12
2.2 Minimum Feature	12
2.3 Effect of Angle	12
2.4 Effect of the Windshield	13
2.5 Vehicle Model	14
2.5.1 Pitch effect	14
2.5.2 Model setup	15
2.5.3 Object Relative Location Calculation	17
2.6 Artificial Distance Calculation	19
2.7 Signal Pre-Processing	20
2.7.1 Kalman Filter Introduction	20
2.7.2 Kalman Filter modeling based on vehicle dynamics	22
2.7.3 Reset method	25
2.8 Lateral Speed Calculation	32
2.9 Braking distance criteria	34
<b>III. SIMULATION SETUP</b>	<b>36</b>
3.1 Lidar Configurations	37
3.2 Driving scenarios	40
3.2.1 Human-crossing	40
3.2.2 Same-direction lane-change	41
3.2.3 Opposite-direction lane-change	42
3.2.4 Turn-and-stop	43
3.3 Scenarios for Lateral Speed Calculation	44

<b>IV. RESULTS</b>	<b>46</b>
4.1 Lidar Collision Avoidance Results	47
4.1.1 Human-Crossing Results	47
4.1.2 Same-Direction Lane-Change Results	49
4.1.3 Opposite-Direction Lane-Change Results	51
4.1.4 Turn-and-Stop Results	52
4.2 Lateral Speed Results	54
<b>V. CONCLUSION AND FUTURE WORK</b>	<b>57</b>
5.1 Conclusion	57
5.2 Future Work	58
<b>VI. DESIGN OF A NEW SMART PEN</b>	<b>60</b>
7.1 Motivation	62
7.2 Design	63
7.3 Smart Pen Concept Based on Customized Components	75
7.4 Heat Transfer Analysis	84
7.5 Force analysis	87
7.6 Conclusion and Future Work	90
<b>APPENDIX A. CAD MODEL</b>	<b>93</b>
<b>APPENDIX B. INITIAL VALUES FOR SIMULATION CASES</b>	<b>95</b>
<b>APPENDIX C. BRAKING TIMES FOR EACH SIMULATION CASE</b>	<b>97</b>
<b>REFERENCES</b>	<b>102</b>

## LIST OF TABLES

Table 1. Comparison of sensors available for autonomous driving.....	3
Table 2. Comparison of Lidar types .....	6
Table 3. Specifications of a VU8 model.....	9
Table 4. Chi-square table .....	27
Table 5. Human-crossing performance (cases 1-6) .....	47
Table 6. Human-crossing performance (cases 7-12) .....	48
Table 7. Same-direction lane-change performance .....	50
Table 8. Opposite-direction lane-change performance.....	52
Table 9. Turn-and-stop performance .....	53
Table 10. Results from lateral speed calculation (path 1 and 2).....	54
Table 11. Results from lateral speed calculation (path 3).....	55
Table 12. Fillet analysis results.....	90
Table 13. Initial values for human-crossing scenario .....	95
Table 14. Initial values for same-direction lane-change scenario.....	95
Table 15. Initial values for opposite-direction lane-change scenario .....	95
Table 16. Initial values for turn and stop scenario.....	96
Table 17. Initial values for lateral speed calculation .....	96
Table 18. Braking times for Lidar sensors in human-crossing scenario (cases 1-6) .....	97
Table 19. Braking times for Lidar sensors in human-crossing scenario (cases 7-12) .....	98
Table 20. Braking times for same-direction lane-change scenario.....	99
Table 21. Braking times for turn and stop scenario .....	100
Table 22. Braking times for opposite-direction lane-change scenario. ....	101

## LIST OF FIGURES

Figure 1. Waymo car with Lidar on top (left). Close-up of a Lidar sensor (right).	4
Figure 2. Lidar distance measurement principal	4
Figure 3. 3D Lidar signal (left) and 2D solid-state Lidar signal (right).	7
Figure 4. Distance measurement within a segment.	8
Figure 5. Leddartech VU8 Lidar model	8
Figure 6. Autonomous driving overview	10
Figure 7. Intensity vs angle for various colored and finish paint	13
Figure 8. Theoretical setup for pitch effect calculation.	15
Figure 9. Parameters and variables for ego vehicle and objects in the global coordinate.	16
Figure 10. Lidar placement and local coordinate	18
Figure 11. Filtered signal with small process noise covariance	25
Figure 12. Comparison of reset and non-reset KF	29
Figure 13. Reset with Mahalanobis distance and pure distance method	30
Figure 14. Reset under high noise variance	31
Figure 15. Missed detection by pure distance method	32
Figure 16. Lateral speed calculation example	33
Figure 17. Simulation structure	36
Figure 18. Single <b>20°</b> Lidar configuration.	37
Figure 19. Dual parallel <b>20°</b> Lidars configuration.	38
Figure 20. Cross-beam <b>20°</b> Lidars configuration.	39
Figure 21. Dual Lidars with <b>40°</b> FoV configuration.	39
Figure 22. <b>20°</b> and <b>48°</b> Lidars configuration.	40
Figure 23. Human-crossing scenario	41
Figure 24. Same-direction lane-change	42
Figure 25. Opposite-direction lane-change	43
Figure 26. Turn-and-stop	44
Figure 27. Driving scenarios for lateral speed test.	45
Figure 28. Difference in distance measurement caused by Lidar configuration.	49
Figure 29. Popular smart pens available on the market.	62
Figure 30. Infrared camera inside a smart pen	63
Figure 31. Mechanism for refill and camera extension.	65
Figure 32. Downward-facing camera.	65
Figure 33. Comparison on retracted and extended camera bar	66
Figure 34. Y-shaped camera bar.	67
Figure 35. Opening for easy refill removal	68
Figure 36. Side view of design I.	69
Figure 37. Design I appearance.	69
Figure 38. Front view of design I.	70
Figure 39. Airbus Beluga freighter	71
Figure 40. Design II front view	72
Figure 41. Design II side view	72
Figure 42. Design III side view	73



Figure 43. Design III front view. ....	74
Figure 44. Design III curved contour for finger placement. ....	74
Figure 45. Battery with hollow center. ....	75
Figure 46. Side view of concept pen. ....	75
Figure 47. Uni-ball Eye. ....	76
Figure 48. Side view of concept pen appearance. ....	76
Figure 49. Rear view of concept pen version 1. ....	77
Figure 50. Refill stopper. ....	78
Figure 51. Stylus tip on pen cap. ....	79
Figure 52. Force transfer from stylus to refill. ....	79
Figure 53. Side view of stacked board. ....	79
Figure 54. Vision through window. ....	80
Figure 55. Transparent pen tip. ....	81
Figure 56. Transparent pen cap. ....	81
Figure 57. New vision area of concept pen. ....	81
Figure 58. Side view of concept pen with a few changes. ....	82
Figure 59. Ventilation slits on concept pen. ....	82
Figure 60. Heatsink in pen. ....	83
Figure 61. Appearance of concept pen with heatsink. ....	83
Figure 62. Electric board used inside smart pen. ....	84
Figure 63. Small copper heatsink. ....	84
Figure 64. Test cylinder for misalignment analysis. ....	84
Figure 65. Heat analysis setup. ....	85
Figure 66. One result from heat analysis. ....	85
Figure 67. Maximum steady-state temperature vs. angle. ....	86
Figure 68. Comparison between contact and no-contact setup. ....	87
Figure 69. Finite Element Analysis setup. ....	88
Figure 70. Displacement of spring. ....	88
Figure 71. Displacement of refill shown in animation. ....	88
Figure 72. Part with high stress concentration. ....	89
Figure 73. Round fillet. ....	89
Figure 74. Force analysis with 1 mm fillet. ....	90
Figure 75. Front view of case. ....	93
Figure 76. Lidar case. ....	93
Figure 77. Side view of case. ....	94

## SUMMARY

Light Detection and Ranging (Lidar) sensors are widely used in autonomous driving test vehicles for object detection. The prices of Lidar sensors, especially 3D rotating Lidar sensors, are extremely expensive. They can range from a few hundred dollars to tens of

In this thesis, the use of one, or a combination of several, more affordable 2D solid-state Lidar(s) for object detection is proposed for use in a specially-structured driving environment. A set of systematic tests were conducted to investigate the Lidars' performance. Several Lidar combinations and installation configurations are proposed. Based on real-world observations, scenarios that lead to potential accidents are constructed in Matlab simulation. The simulation produces artificial Lidar signals using a similar distance calculation algorithm as the actual sensors. The goal of the simulation is to determine the combination and installation configuration that yields the fastest and most accurate detection, while also bearing in mind the cost of the corresponding physical system. A Kalman Filter (KF) is used for filtering noisy signals and providing state estimation from the artificial Lidar signals. While the Lidar signals are modeled in a linear fashion, there are frequent discontinuities in the signals, requiring the KF to constantly reset itself to reduce converge time and overshoot. Mahalanobis Distance is used as a basis for the Kalman Filter's reset criterion.

The thesis contributes to a better understanding of the Lidar sensor's capability; proposes various Lidar sensor combinations and configurations; compares their

performance under different dangerous driving scenarios; and finally, provides a best Lidar combination solution for the scenarios.

# I. INTRODUCTION

Light Detection and Ranging (LiDAR or Lidar) is a remote-sensing method for detecting objects and measuring their distance using light. It is widely used on autonomous test vehicles. However, the cost of Lidar sensors has limited their use on mass-production passenger vehicles. This thesis presents a study of using much more affordable and compact 2D Lidar sensors for a specially-structured autonomous driving scenario.

## 1.1 Motivation

Currently at Daimler's Sindelfingen production plant, the average daily production rate is 1000 vehicles per day. Once a new vehicle comes off the assembly line, a driver has to drive the car for up to 1.5 km distance to a parking lot where all new cars are loaded onto trucks or trains. The driver, then, takes a shuttle bus from the parking lot back to the assembly line to drive the next cars. It is estimated that driving each new car from the assembly line to the parking lot takes 8 minutes of the employee's time. Assuming employees get paid 35 euros per hour [1], then in one year, the company has to pay over 1.7 million euros in salary to the drivers.

One solution to saving cost in transporting the vehicles is to add sensors to them and make them drive autonomously. If the average budget for one set of such sensor is limited at 10,000 euros, and a total of 100 sets are deployed within the plant, then that is equivalent to one million euros. Automation cost is usually compared to two years of salary paid to workers that would be replaced. That means the cost of one million euros for autonomous-driving systems should be compared to 3.4 million euros of labor cost.

However, not all the 10,000-euro budget for each set will be used for purchase of hardware. Instead, it includes the cost of research and development, hardware required for installation onto cars, for example, robot arms, and safety installations, such as barriers and fences around the robot, etc. Therefore, it is important that the cost for autonomous driving sensors be limited.

## **1.2 Sensors for Autonomous Driving**

Most commonly used sensors for object detection in autonomous driving are radar sensors, ultrasonic sensors, camera, and Lidar sensors. A comparison of their performance concerning range, resolution, and Field-of-View, etc. is shown in Table 1 [2-5]. Lidar sensors generally have good range, high resolution, and can still be used under light rain and fog conditions. High-capability 3-D Lidars can scan an area of 360° in the horizontal plane with a vertical Field-of-View (FoV) of up to 30°. A Lidar sensor is chosen to perform the autonomous driving task in this thesis.

**Table 1. Comparison of sensors available for autonomous driving**

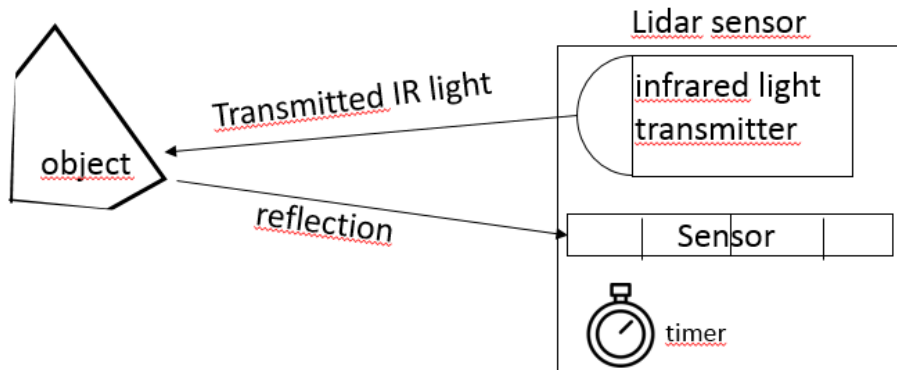
Sensor	2D Lidar	Radar	Camera	Ultrasonic
Range	+	++	+	-
Resolution	+	-	++	-
Field of View	+	+	++	
Bad weather	+	++	-	++
Poor lighting	++	++	-	++

### **1.3 Light Detection and Ranging (Lidar)**

Lidar sensors, as seen in Figure 1, measure the distance to an object using light, usually in the infrared spectrum. The distance is measured based on the Time-of-Flight (ToF) principal, which measures the time delay from when the light is transmitted to the time its reflection is received. Figure 2 shows the distance measurement principal used by Lidar sensors.



**Figure 1. Waymo car with Lidar on top (left). Close-up of a Lidar sensor (right).**



**Figure 2. Lidar distance measurement principal**

The distance measured using ToF is calculated by

$$D = \frac{c \times t}{2} \tag{1}$$

where  $D$  is the measured distance between the object and the sensor,  $c$  is the speed of light, which is  $2.998 \times 10^8 \text{ m/s}$  in a vacuum and slightly slower inside Earth's atmosphere, and  $t$  is the time delay.

There are currently 3 types of market-available Lidar sensors – 3D rotating Lidar, 2D rotating Lidar, and 2D solid-state Lidar. A rotating Lidar has a motor that rotates a transmitter or a mirror about the vertical axis, thereby allowing the Lidar to scan the entire horizontal plane. Rotating Lidars have high-resolution horizontal FoV of up to 360°, with some vertical FoV, typically around 30°, split between angles pointing above and below the horizontal plane. But, due to their construction, they are also costly and susceptible to vibrations.

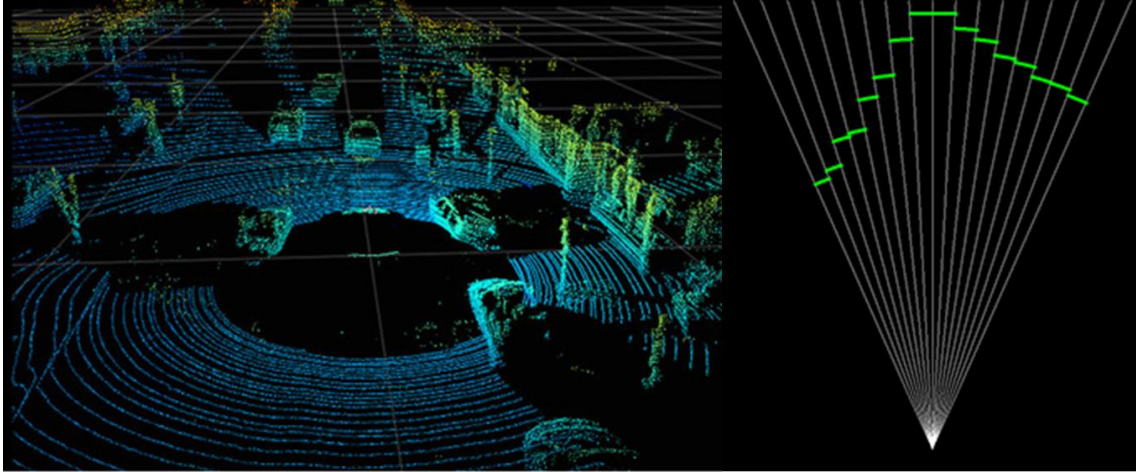
A comparison of the three types of Lidars in terms of performance and price is shown in Table 2 [2, 6, 7]. A more advanced 3D solid-state Lidar based on phased array technology is currently under development and not yet publicly available. From the row of “retail price”, we can see that the price of a 2D solid-state Lidar is \$540 while a rotating Lidar costs \$4700. The price difference is very significant and we wish to explore the possibility of using a 2D Lidar for the project.



**Table 2. Comparison of Lidar types**

Lidar Type	3D rotating	2D rotating	2D solid-state
Lidar Model Example	Velodyne VLP-16	Sick LMS1xx	Leddartech Vu8
Horizontal FoV	360°	270°	20°
Vertical FoV	30°	0.15°	0.3°
Horizontal Resolution	0.1°	0.25°	2.5°
Vertical Resolution	2°	N/A	N/A
Retail Price	\$ 7999	\$ 4700	<b>\$ 540</b>
Weight	830 g	1100 g	<b>110.3 g</b>

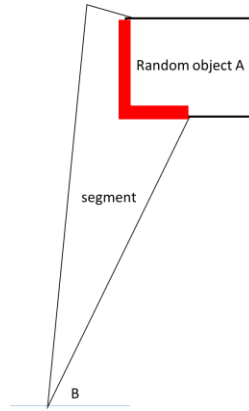
Compared to rotating Lidars, solid-state Lidars have the advantage of being more affordable and compact. Without moving parts, the Lidar is less likely damaged under harsh vehicle movement. The measurement precision and accuracy is also less likely to be affected in such conditions. A solid-state Lidar has coarse resolution and limited functionality. For example, achieving object classification with such sensors is highly unlikely. Figure 3 compares the signal of a 3D rotating Lidar and a 2D solid-state Lidar [2]. However, a more capable rotating Lidar is probably not necessary given a specially structured driving environment. This idea forms the basis of this thesis: the use of affordable and compact solid-state Lidars for autonomous driving.



**Figure 3. 3D Lidar signal (left) and 2D solid-state Lidar signal (right).**

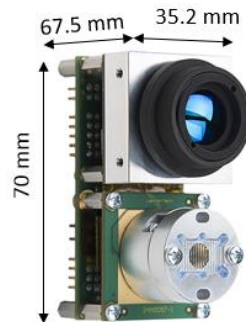
#### **1.4 Lidar Model of Interest**

The specific model of interest is the Leddartech VU8, as shown in Figure 5. The VU8 comes with three horizontal FoV configurations. They are  $20^\circ$ ,  $48^\circ$ , and  $100^\circ$ , with horizontal resolution of  $2.5^\circ$ ,  $6^\circ$ , and  $12.5^\circ$  respectively. The distance reading from each segment is the averaged distance of all measurements within the segment. For example, as shown in Figure 4, the distance reading for this segment is the averaged distance of every point on the red, L-shaped edge, to point B.



**Figure 4. Distance measurement within a segment.**

Key specifications from the user's manual of the 20° model are given in Table 3 [2]. This thesis provides an investigation of this inexpensive and lightweight 2D solid-state Lidar sensor.



**Figure 5. LeddarTech VU8 Lidar model**

**Table 3. Specifications of a VU8 model**

Specification	Leddartech VU8-020-003-02
Horizontal FoV	20°
Number of segments	8
Horizontal resolution	2.5°
Vertical FoV	0.3°
Detection range	215 m (retro-reflector) 60 m (white Target) 38 m (grey target)

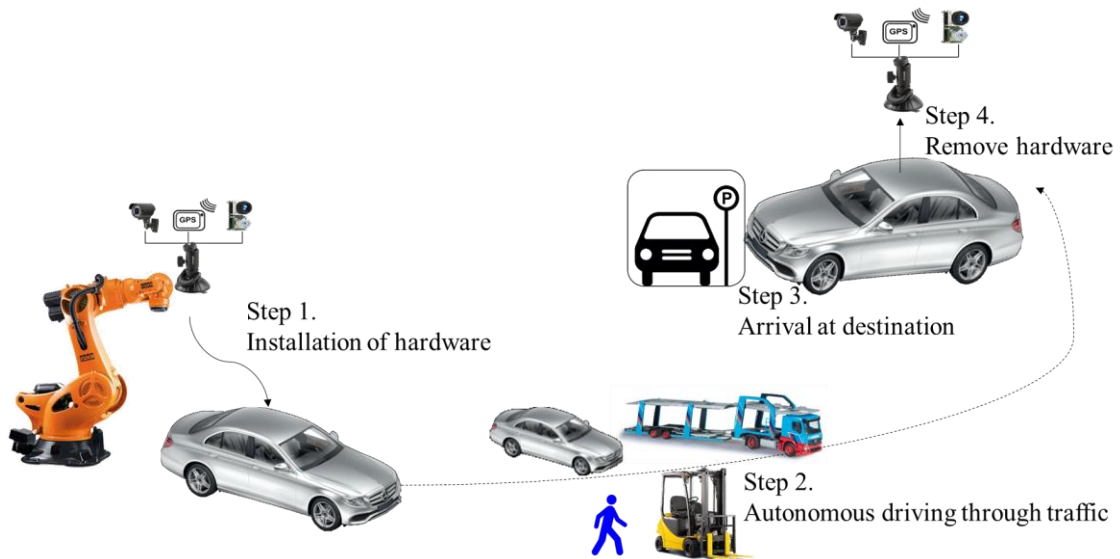
### **1.5 Specially-Structured Driving Environment**

The structured driving environment is introduced in section 1.1.

At the start of the journey, the Lidar sensor, as well as sensors for navigation, will be attached to a car. The car drives autonomously to a target location up to 1.5 km away through a predefined and fixed route. Along the route, the vehicle encounters traffic, including both autonomously and non-autonomously driven passenger vehicles, trailer trucks, utility vehicles (e.g. forklifts), pedestrians, and intersections. At any part of the

road, the speed limit is no higher than 40 km/h. The Lidar sensor is responsible for object detection, while the navigation task is performed by a GPS system.

As the vehicle arrives at the target location, the set of hardware is removed from the vehicle. The process is illustrated in Figure 6.



**Figure 6. Autonomous driving overview**

## 1.6 Thesis Outline

Chapter 2 investigates Lidar technology and introduces the vehicle model and the Kalman Filter used. Chapter 3 explains the simulation framework. Chapter 4 discusses the results from the simulation. Finally, Chapter 5 presents the conclusions and future work. Additionally, Chapter 6 provides an overview of a smart-pen project done in Georgia Tech.

## II. BACKGROUND

Both lab tests and field tests are conducted to investigate the behavior and capability of the Lidar sensor.

### 2.1 Background on Light

#### 2.1.1 Intensity attenuation

The intensity of light attenuates as it travels. This attenuation is described by the inverse square law [8]

$$I \propto \frac{1}{D^2} \quad (2)$$

Where  $D$  is the distance of the measurement point from the origin of the light source, and  $I$  is the intensity measured at this point.

#### 2.1.2 Laser class 1

A Lidar is categorized as a laser class 1 system, which means it is “*incapable of producing damaging radiation levels during operation*” for the naked eye [9]. This is essentially a limitation of the power emanating from the laser source. Because a wide FoV Lidar covers more area than a narrower FoV Lidar with the same amount of energy, its intensity, or energy per area, is therefore lower, and its detection range is correspondingly shorter.

### 2.1.3 Reflectivity

A surface appears dark to the human eye because it reflects less visible light than a light-color surface. Similarly, dark-color surfaces also reflect less infrared light, except that human cannot observe this difference directly. This means that a black car is less detectable than a white car, at the same distance. Many tests presented in this thesis are conducted using black objects to explore the Lidar's absolute minimum capability.

## 2.2 Minimum Feature

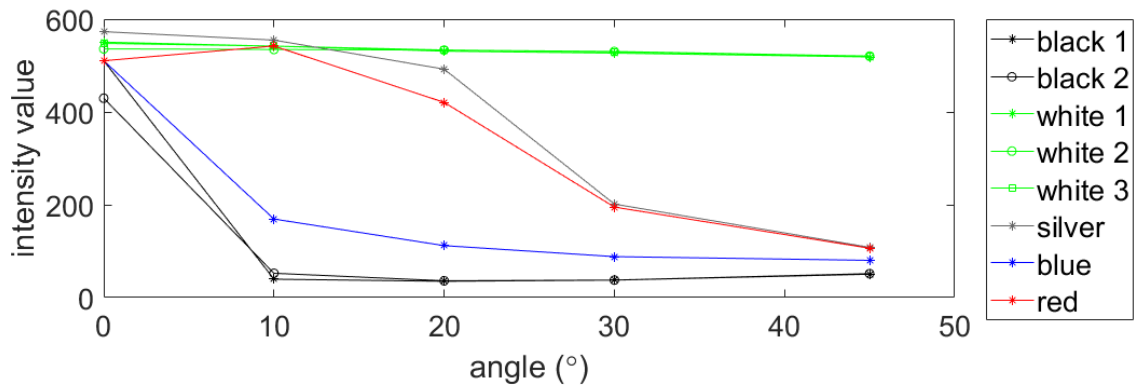
According to the user manual, the horizontal FoV of the VU8 is  $20^\circ$  and, with 8 segments, each segment covers  $2.5^\circ$ . However, when an object occupies only a small portion of a segment, the total reflected light from that object may not have enough intensity to be a detectable reflection, and therefore, it may be discarded as noise. To test the minimum feature size that can be detected, a black cable of 5 mm diameter was hung vertically along the laser's path. The black cable was moved away from the Lidar sensor and the distance at which the cable was no longer detected by the sensor was recorded.

The maximum distance for consistent detection was 2.67 meters. The angle can be calculated using  $Arc\ Length = Radius \times \theta_{rad}$ .  $\theta_{min}$  is  $0.107^\circ$ . Therefore, the simulation presented in Chapter 3 uses  $0.1^\circ$  as a distance measurement interval.

## 2.3 Effect of Angle

We wish to explore the energy loss when the laser beam hits a surface at various angles and for some of the more popular vehicle colors.

Besides distance measurement, the Lidar sensor also provides measurement for intensity value of reflected light. Figure 7 shows the intensity value of the reflected light plotted against the incident angle of the laser beam. Results show that dark colors reflect less light, and that the difference between colors is even greater when the light beam is not normal to the surface.



**Figure 7. Intensity vs angle for various colored and finish paint.**

## 2.4 Effect of the Windshield

Placing the Lidar sensor behind the windshield would protect the Lidar from rain and snow. And without having to attach it externally, it also means that the vehicle's appearance will not be affected. However, the effects of reflection from, and refraction through, the glass must be considered. Reflection causes significant loss of detection range, especially when the emitted light enters glass at a high incident angle. Moreover,



some car models may come with infrared filters in their windshields to reduce irradiation from the sun.

Test results show that a black car can be detected up to 30 meters when the Lidar sensor is placed behind the windshield, and up to 65 meters when placed outside the windshield. Therefore, the Lidar sensors should be placed outside the windshield.

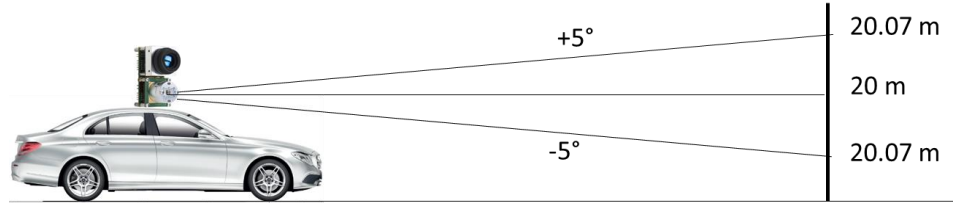
## **2.5 Vehicle Model**

It is desired that the vehicle model be simplified and the simulation be constructed in 2D. However, there is an inevitable vehicle movement in the third dimension that affects the Lidar's measurements. As a vehicle accelerates and decelerates, the vehicle pitches up and down. The road, not being perfectly flat, also slightly affects the vehicle's pitch movement. As the Lidar pitches with the vehicle, the Lidar's distance reading will show incorrect distance measurements between the object and the vehicle. We wish to know how much variation the pitch movement has on the readings and whether the pitch movement can be neglected.

### *2.5.1 Pitch effect*

Based on my personal observation inside the plant, and the fact that the speed limit is low, there is not much variation in the vehicles' speeds. Most vehicles travel steadily around the speed limit. Therefore, the pitch angle is very small. A theoretical calculation is set up based on this assumption. If the car is static and parked 20 meters away from a flat wall, as shown in Figure 8, then the Lidar has an ideal measurement of 20 meters. If

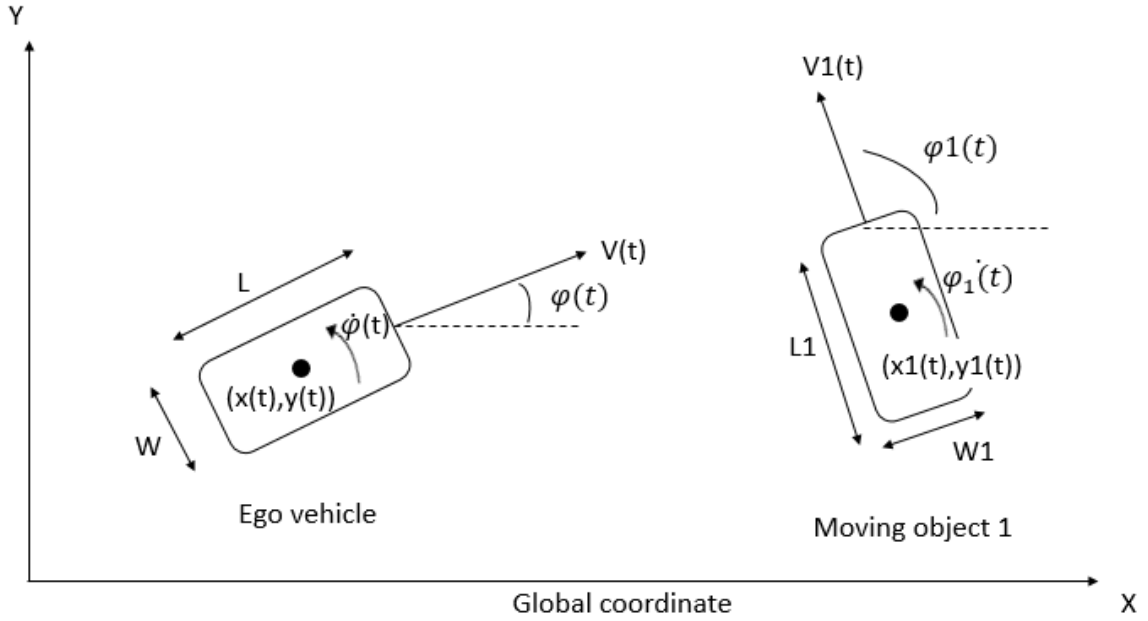
the vehicle has a pitch angle of + or - 5°, then the reading would instead be 20.07 meters, which reflects an error of less than 0.4%. To simplify the model, the pitch effect is neglected and the vehicles are modeled as 2D objects.



**Figure 8. Theoretical setup for pitch effect calculation.**

### 2.5.2 Model setup

Vehicles and pedestrians are represented in the simulation as rectangular shapes. A vehicle model has length of 4.8 m and width 1.9 m, which are taken from the actual dimension of the current E-Class sedan. Based on several measurements, it is estimated that the average thickness of a German grown-up's upper body is 30 cm, and the average shoulder-width measures 50 cm. Therefore, a pedestrian is modeled as a rectangular shape of 30 cm by 50 cm. The position  $(x(t), y(t))_G$  of an object is defined at its geometric center in the global coordinate. Each moving object also has the following variables – speed  $V(t)$ , heading  $\varphi(t)$ , and yaw rate  $\dot{\varphi}(t)$  [10]. Figure 9 shows the autonomous-driving vehicle and another moving object in the global coordinate, as well as all their respective variables.



**Figure 9. Parameters and variables for ego vehicle and objects in the global coordinate.**

Global coordinates of a moving object are calculated as follows

$$\varphi(t) = \varphi_0 + \int_{t_0}^t \dot{\varphi}(t) dt \quad (3)$$

$$V(t) = \text{func}(t) \quad (4)$$

$$x(t) = x_0 + \int_{t_0}^t V(t) * \cos(\varphi(t)) dt \quad (5)$$

$$y(t) = y_0 + \int_{t_0}^t V(t) * \sin(\varphi(t)) dt \quad (6)$$

Where  $func(t)$  can be any arbitrary user-defined function that is realizable and reasonable in real-world driving.

A differential GPS will be navigating the autonomous-driving vehicle along the route, so unless the GPS is giving very inaccurate positioning, the vehicle should not collide with a building or wall. However, it happens that a car is parked on the right-most lane of the route, which indeed happens very often along the route. Therefore, any static object is modeled as a vehicle but without the dynamic variables.

### 2.5.3 Object Relative Location Calculation

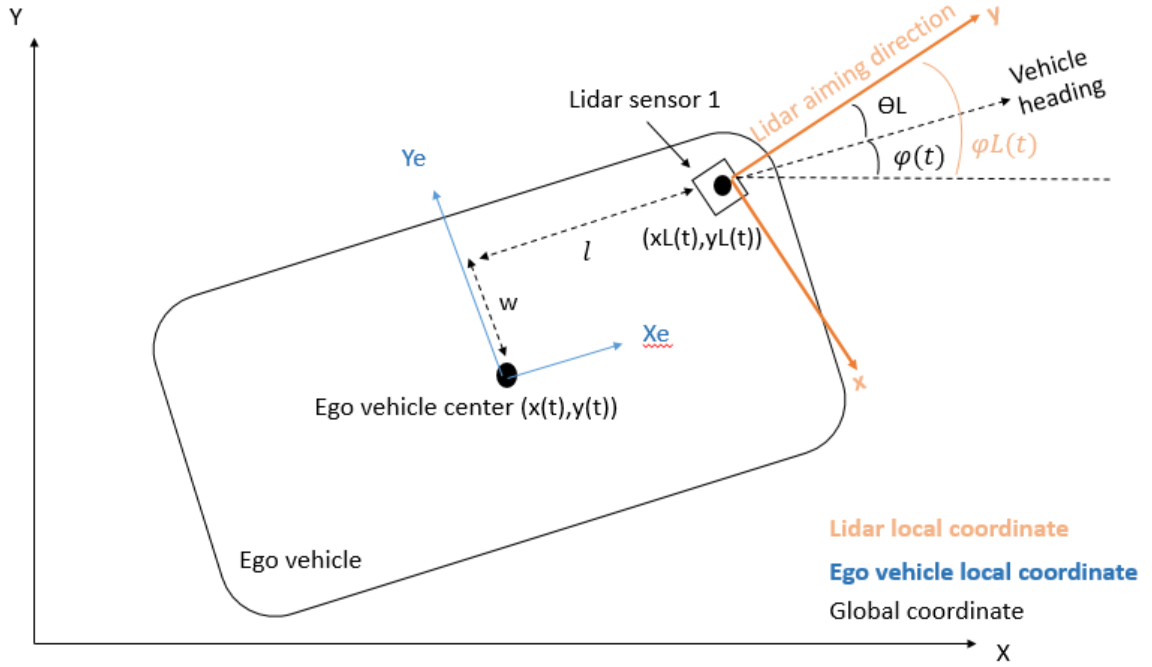
To calculate the relative distance of the object from the Lidar sensor, in part 2 of the simulation, an object's location is converted from the global coordinate to the Lidars' local coordinates. The artificial Lidar distance measurement, the output, is passed to part 3 for signal pre-processing.

To calculate the object's location in the local coordinates, the location and heading of each Lidar should be known. Figure 10 shows the Lidar's relative location and heading to the vehicle. The Lidar position is calculated as follows

$$x_L(t) = x(t) + l * \cos(\varphi(t)) - w * \sin(\varphi(t)) \quad (7)$$

$$y_L(t) = y(t) + l * \sin(\varphi(t)) + w * \cos(\varphi(t)) \quad (8)$$

$$\varphi_L(t) = \varphi(t) + \theta_L \quad (9)$$



**Figure 10. Lidar placement and local coordinate**

Let  $P$  be a point whose location is  $(x_p, y_p)_{\text{global}}$  in the global coordinates, and  $L$  a Lidar whose location and aiming direction are  $(x_L, y_L)_{\text{global}}$  and  $\varphi_L(t)$ , then the coordinate of  $P$  in the Lidar's local coordinates can be calculated as

$$P_{\text{Lidar}} = R(\varphi_L) \times (P_{\text{global}} - L_{\text{global}}) \quad (10)$$

Where  $P$  and  $P^*$  are coordinates of point  $p$  in global - and Lidar coordinates,  $L$  the Lidar location in global coordinates, and  $R$  the rotation matrix. The coordinates of  $P$  first goes through a translation, then rotates to get  $P_{\text{Lidar}}$ . The new coordinates are then  $(x_p, y_p)_{\text{Lidar}}$  where

$$\begin{bmatrix} x_p \\ y_p \end{bmatrix}_{\text{Lidar}} = \begin{bmatrix} \cos(\varphi_L(t)) & \sin(\varphi_L(t)) \\ -\sin(\varphi_L(t)) & \cos(\varphi_L(t)) \end{bmatrix} \left( \begin{bmatrix} x_p \\ y_p \end{bmatrix}_{\text{global}} - \begin{bmatrix} x_L \\ y_L \end{bmatrix}_{\text{global}} \right) \quad (11)$$

or

$$\begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix}_{\text{Lidar}} = \begin{bmatrix} \cos(\varphi_L(t)) & \sin(\varphi_L(t)) & 0 \\ -\sin(\varphi_L(t)) & \cos(\varphi_L(t)) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -x_L \\ 0 & 1 & -y_L \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix}_{\text{global}} \quad (12)$$

## 2.6 Artificial Distance Calculation

Instead of calculating the transformed coordinates for each of the discretized points on the rectangle shape, only the coordinates of the four corners are transformed.

Unlike how a real Lidar sensor measures distance, the calculation of distance in the simulation is done in a discrete fashion. Within each segment, distance is measured at a  $0.1^\circ$  interval, and all measurements in one segment are averaged to form the distance reading of the one segment. And each measurement is made by calculating the distance to the rectangle.

Assuming the coordinates for two consecutive corners of a rectangle are  $C_1$  and  $C_2$ . Then a point  $P$  on the line formed by  $C_1$  and  $C_2$  will satisfy

$$P = C_1 + \alpha * (C_2 - C_1) \quad (13)$$

Where  $\alpha$  is a scalar. And if  $0 < \alpha < 1$ , then point  $P$  is in between  $C_1$  and  $C_2$ . And  $P$  coincides with  $C_1$  or  $C_2$  if  $\alpha = 0$  or  $\alpha = 1$ .

Measurement-beams with  $0.1^\circ$  interval are used to calculate distance and they are defined as

$$y = \tan^{-1}(80^\circ + 0.1^\circ \times n) * x, \quad n = 1 \dots 200 \quad (14)$$

The intersections of each measurement-beam and the lines formed by  $C_1$  through  $C_4$  are found using Equation 33. Distances of all measurement-beams within one segment are averaged to find the distance reading of the segment.

Measurement-beam with  $n = 100$  is neglected, because  $\tan^{-1}(90^\circ) = \infty$ .

## 2.7 Signal Pre-Processing

A Kalman Filter is selected to provide state estimation for the object distance and the relative speed. Because the signal is not always continuous, the Kalman Filter requires a few time steps to converge. During the transient, estimation value can yield large errors and overshoot. Both the time delay and the error are problematic in applications that target safety. The data refresh rate of the Lidar sensor is slow at high accuracy settings, and a delay of 0.1s can turn a near-miss into a collision. It is therefore beneficial to explore a statistical way to reset the initial estimation values of the Kalman Filter when the signal experiences discontinuity, while also avoiding false alarms.

### 2.7.1 Kalman Filter Introduction

The Kalman Filter, introduced by R. E. Kalman in 1960, is commonly used for linear state estimation [11, 12]. A linear model is set up in state space form and it consists of two

parts. The first part estimates the model states at the current time step – the a priori estimation. And the second part calculates the measurement state based on the a priori estimation. Finally, the posterior estimation is given as a combination of the a priori estimation and the measurement state using a Kalman Gain.

First, to construct a Kalman Filter, a system model is expressed in state space form [13]

$$x_k = F_{k-1}x_{k-1} + B_{k-1}u_{k-1} + w_{k-1} \quad (15)$$

$$z_k = H_k x_k + v_k \quad (16)$$

Where  $x_k$  is the state at time  $k$  and  $z_k$  is the observable states.  $F_k$  is the transition matrix,  $B_k$  the input matrix, and  $H_k$  the measurement matrix.  $w \sim N(0, Q_k)$  and  $v \sim N(0, R_k)$  are additive, white, and Gaussian noise, with zero mean and covariance matrix

$$Q_k = E[w_k^T w_k] \quad (17)$$

and

$$R_k = E[v_k^T v_k] \quad (18)$$

Equation 16 gives an a priori estimate of the state at time  $k$ , denoted  $\hat{x}_k^-$ . The estimate of the measurement state is denoted  $\hat{y}_k$  and the actual measurement at time  $k$  is  $y_k$ . The Kalman Filter provides the a posterior estimation of the state, or, state estimation update, at time  $k$  by



$$\hat{x}_k = \hat{x}_k^- + G_k(y_k - H_k \hat{x}_k^-) \quad (19)$$

Where  $G_k$  is the *Kalman Gain*. To calculate  $G_k$ , the a priori covariance for the estimation error,  $P_k^-$  is needed.

$$P_k^- = F_{k-1} P_{k-1} F_{k-1}^T + Q_{k-1} \quad (20)$$

And the Kalman Gain is subsequently found by

$$G_k = P_k^- H_k^T [H_k P_k^- H_k^T + R_k]^{-1} \quad (21)$$

Finally, the error covariance update is given by

$$P_k = (I - G_k H_k) P_k^- \quad (22)$$

### 2.7.2 Kalman Filter modeling based on vehicle dynamics

The time-continuous relation between distance, velocity, and acceleration is simply

$$v_2 = v_1 + \int_{t_1}^{t_2} a(t) dt \quad (23)$$

and

$$d_2 = d_1 + \int_{t_1}^{t_2} v(t) dt \quad (24)$$

Where  $d_1, d_2$  and  $v_1, v_2$  are the distance and velocity at time  $t_1$  and  $t_2$ . And  $v(t)$  and  $a(t)$  are the time varying velocity and acceleration. Because the Lidar returns a distance measurement after every time interval, the above relation shall be written in discrete form as

$$v_k = v_{k-1} + a_{k-1} \times T \quad (25)$$

$$d_k = d_{k-1} + v_k \times T \quad (26)$$

$$d_k = d_{k-1} + v_{k-1} \times T + \frac{1}{2} a_{k-1} \times T^2 \quad (27)$$

Where  $a_k, v_k$ , and  $d_k$  are the relative acceleration, relative speed, and distance between the vehicle and the detected object at time  $k$ , with  $T$  being the data refresh time for the Lidar sensor.

In the state space representation, instead of modeling the acceleration  $a_k$  as a state, it is modeled as a perturbation to the velocity state  $x_2 = v_k$ , thus  $a_k$  is represented as  $w_k$  [13]. This is based on the observation that most of the time, the object vehicles drive around the speed limit without much velocity variation, and humans usually walk at constant speeds.

The state vector of the model is

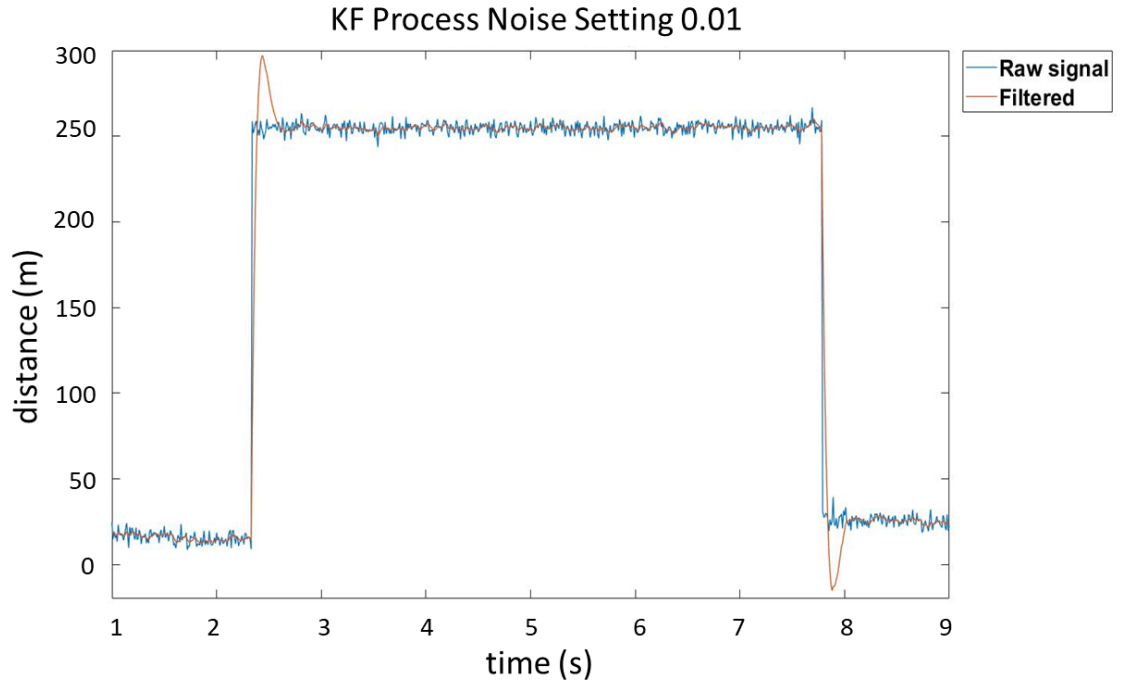
$$x_k = \begin{bmatrix} d_k \\ v_k \end{bmatrix} \quad (28)$$

Since the Lidar provides only one measurement, the distance, the state estimation for the Kalman Filter can be represented as

$$x_k = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} x_{k-1} + \begin{bmatrix} \frac{1}{2} T^2 \\ T \end{bmatrix} w_{k-1} \quad (29)$$

$$y_k = [1 \quad 0] x_k + v_k \quad (30)$$

Figure 11 shows the filtered signal compared to the observed signal. The observed signal is the artificial signal from the simulation, with measurement noise added. As can be seen, when process noise covariance  $Q$  is set to a small value, the Kalman Filter effectively smoothens the noisy signal based on the linear dynamic model. However, this results in the estimation not being able to effectively learn from the measurement when there is a discontinuity, and the filtered signal yields large estimation errors and time delays. Of course, setting  $Q$  to a larger value can reduce this unfavorable effect, but that would also increase the uncertainty and reduce the effect of filtering. At some point, increasing  $Q$  will make the filter useless. One method to tackle this problem is to make  $Q$  time-varying, depending on the observed signal. Another method is to reset the initial estimation value  $x_0$  for the Kalman Filter. This last approach is used in this thesis.



**Figure 11. Filtered signal with small process noise covariance**

### 2.7.3 Reset method

When an object at close range suddenly comes into the field of view of the Lidar sensor, the distance measurement will suddenly jump to a smaller value, causing a discontinuity in the signal. This causes the estimated error to increase. The estimation may need a long settling time to converge to the actual values. This transition is harmful in two ways. First, it will display a distance value that is farther than the actual object, making dangerous scenarios seem less severe. Second, when the distance does get corrected, some time may have passed. Furthermore, at high relative speeds, a split second means a few meters of travel and losing valuable braking time. Therefore, the Kalman Filter needs to be reset when such things happen. However, such sudden jumps in signal measurement

is also be caused by noise. Two methods to define the reset criterion are explored in this section.

#### 2.7.3.1 Reset Method A: Distance Difference

The first method is to reset whenever the difference between the current value and the preceding value is greater than a maximum possible distance based on real-time speed. Assuming an object is driving towards the autonomous vehicle at 30 km/h, with the autonomous vehicle also driving at 30 km/h. Then, the relative speed is 60 km/h. In a time-interval of 0.1s, the maximum relative distance change is 1.67 meters. Therefore, if distance changed more than 1.67 meters, the new signal must be from a new, and closer, object. Therefore, at this point, a reset to the KF should be performed. However, even when the Lidar is measuring the distance of the same object, but with the influence of large noise signal, the difference in distance still exceeds 1.67 meters. This causes a false alarm.

#### 2.7.3.2 Reset Method B: Mahalanobis Distance

Another method is to use the Mahalanobis Distance [15]. The Mahalanobis distance was introduced by P. Mahalanobis in 1936 [16]. The Mahalanobis distance is a measurement of the distance between one observation  $x$  and a distribution of past observations  $D$ . As  $x$  moves away from the mean of  $D$ , the Mahalanobis distance increases. And the Mahalanobis distance is 0 when  $x$  is at the mean of  $D$ . The calculation of Mahalanobis distance is given by

$$D_M(x) = \sqrt{(x - \mu)^T S^{-1} (x - \mu)} \quad (31)$$

Where  $x$  is the observation point,  $\mu$  is the mean of the distribution, and  $S$  is the covariance matrix of the distribution. Finally,  $D_M$  is the Mahalanobis distance.

To determine the Mahalanobis distance for reset, data in Table 4 is used.

**Table 4. Chi-square table**

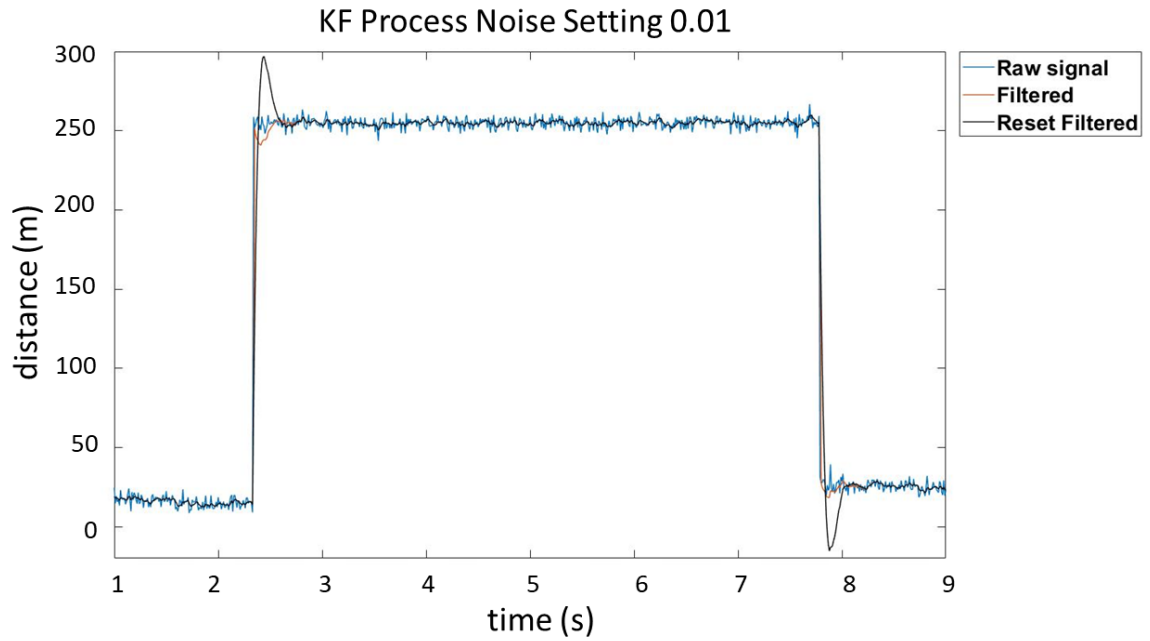
	$P(X \leq x)$							
	0.010	0.025	0.050	0.100	0.900	0.950	0.975	0.990
$r$	$\chi_{0.99}^2(r)$	$\chi_{0.975}^2(r)$	$\chi_{0.95}^2(r)$	$\chi_{0.90}^2(r)$	$\chi_{0.10}^2(r)$	$\chi_{0.05}^2(r)$	$\chi_{0.025}^2(r)$	$\chi_{0.01}^2(r)$
1	0.000	0.001	0.004	0.016	2.706	3.841	5.024	6.635
2	0.020	0.051	0.103	0.211	4.605	5.991	7.378	9.210
3	0.115	0.216	0.352	0.584	6.251	7.815	9.348	11.34
4	0.297	0.484	0.711	1.064	7.779	9.488	11.14	13.28
5	0.554	0.831	1.145	1.610	9.236	11.07	12.83	15.09
6	0.872	1.237	1.635	2.204	10.64	12.59	14.45	16.81
7	1.239	1.690	2.167	2.833	12.02	14.07	16.01	18.48
8	1.646	2.180	2.733	3.490	13.36	15.51	17.54	20.09
9	2.088	2.700	3.325	4.168	14.68	16.92	19.02	21.67
10	2.558	3.247	3.940	4.865	15.99	18.31	20.48	23.21

Table 4 is called a Chi-square Table, or  $\chi^2$  Table. The values,  $P$ , in the second row, ranging from 0.010 to 0.990, are the probability that a point in a distribution is found within a certain distance from the mean of the distribution.  $\chi^2$  is  $1 - P$ . The  $r$  value in the first column is the degree of freedom. The  $x$  values in the main part of the table are the distances. Since our only observable variable is the relative distance, our degree of freedom is 1. For a safety-related project, we wish to achieve high confident levels,

therefore, the  $\chi^2$  value is chosen to be 0.01, or, equivalently,  $P = 0.990$ , and the corresponding  $x$  value from the table is 6.635. This means that, for a 1 degree of freedom system, 99% of the distribution is less than 6.635 away from the distribution mean. Therefore, when the calculated Mahalanobis distance is greater than 6.635, the Kalman Filter will reset.

### 2.7.3.3 Comparing Performance of Two Methods

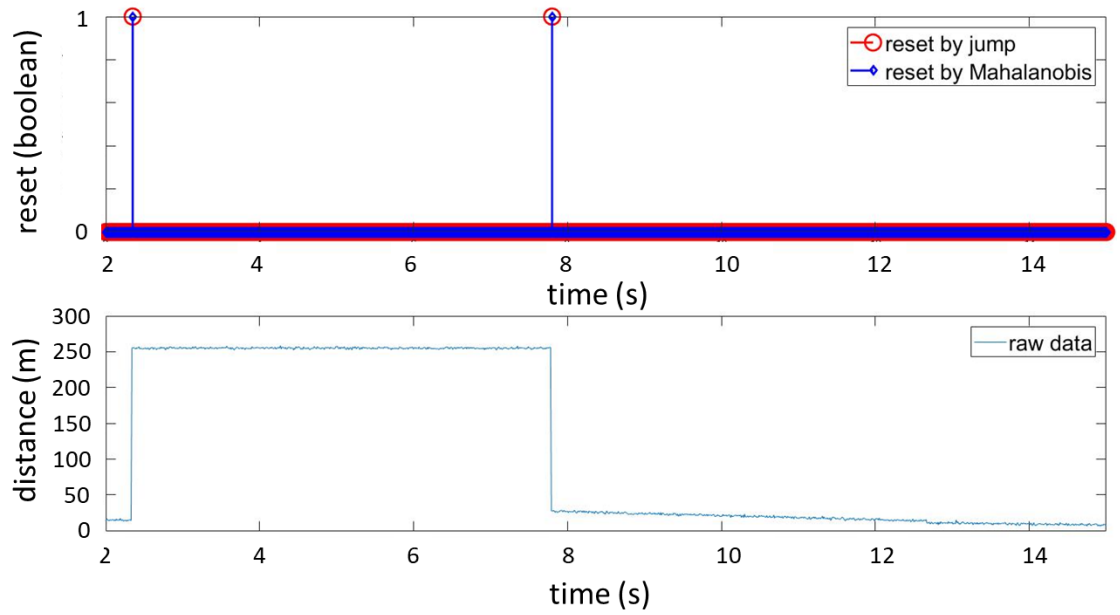
Figure 12 compares the filtered result with and without reset function. For this set of data, where the noise amplitude is relatively small compared to the ground truth, both methods A and B perform as expected and function identically. The estimation error of the KF with reset function is significantly smaller than the KF without reset function, and the converging time is also smaller.



**Figure 12. Comparison of reset and non-reset KF.**

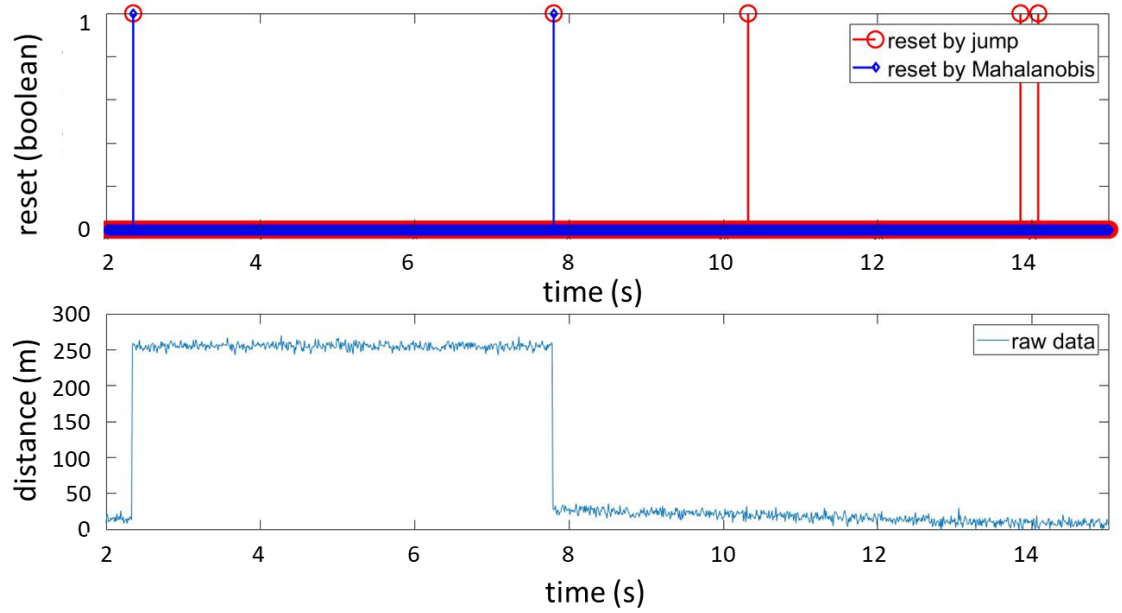
Figure 13 shows the correct reset commands generated by the Distance Difference method and the Mahalanobis Distance method for the data in Figure 12. The reset command is a Boolean value. When the value is 1, the KF is reset. As explained earlier, both methods function identically for this set of data.





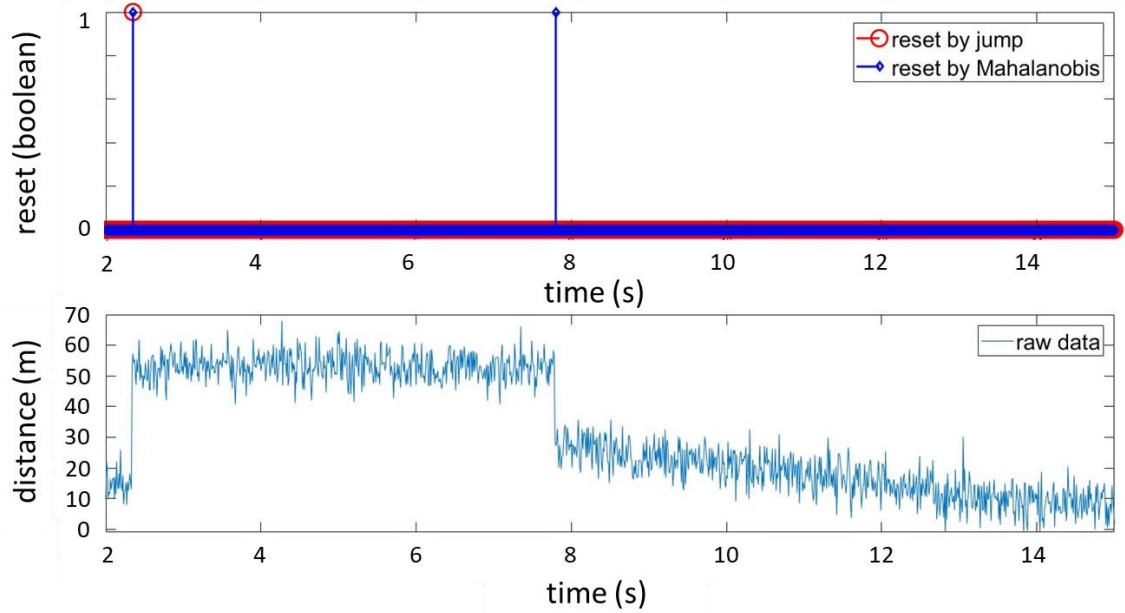
**Figure 13. Reset with Mahalanobis distance and pure distance method**

Measuring pure distance difference is problematic when noise is high. Figure 14 shows false alarms when the pure distance difference method is used.



**Figure 14. Reset under high noise variance**

To counter the noise problem, the threshold for triggering a reset can be set higher, but this leads to a problem of not resetting the Kalman Filter when the distance change is not large, combined with noise of high amplitude. As seen in Figure 15, the distance difference method was not able to detect the signal change caused by a new object. The Mahalanobis distance method was able to track the changes.



**Figure 15. Missed detection by pure distance method.**

## 2.8 Lateral Speed Calculation

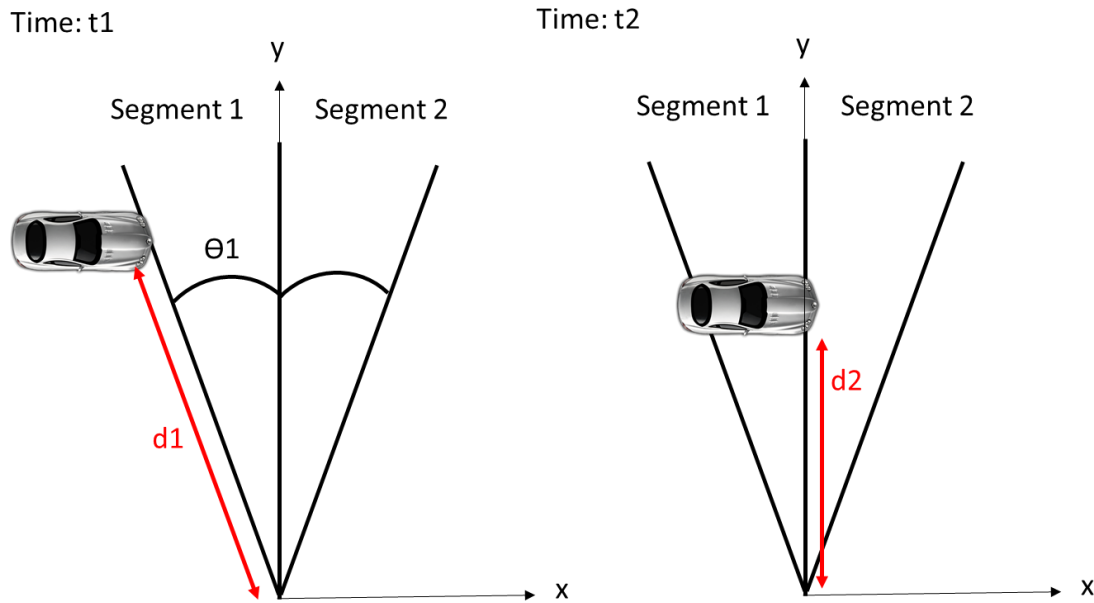
When the Lidar sensor is not capable of calculating the lateral speed of an object, the computer would have to command the autonomous vehicle to brake regardless of whether an object is on the collision path, or it is driving away from the vehicle. This reduces the efficiency of the vehicle. With coarse horizontal resolution, the accuracy of lateral speed calculation is limited, but we still wish to explore the feasibility and reliability of this function.

The horizontal distance is calculated using the time the object travels through a segment . A lateral speed can be calculated using the time between when the first signal appears in a segment and when the first signal appears in the next segment, or, the time between

when the previous segment loses its last signal to when the current segment loses its last signal.

The horizontal distance is calculated from the measured distance using simple trigonometry.

As shown in Figure 16, at time  $t_1$ , the object first appears in segment 1 with a distance reading of  $d_1$ . And at time  $t_2$ , the object appears in segment 2 with distance reading of  $d_2$ . If the autonomous vehicle is driving with a constant heading, the relative lateral movement of the object is then  $d_1 \times \sin(\theta) - d_2 \times \sin(0)$ . The lateral speed is then found by dividing this distance by  $t_2 - t_1$  [18].



**Figure 16. Lateral speed calculation example.**

## 2.9 Braking distance criteria

To determine the performance of each configuration, the times the systems start commanding the vehicle for emergency braking are compared. To determine when the vehicle should start braking, a braking zone is defined [19]. The braking zone is a real-time distance based on the autonomous vehicle's speed and object's speed. When the object is within the calculated braking zone, the vehicle starts emergency braking. The accuracy of the braking time depends on the accuracy of the distance and speed measurements and the FoV of the Lidar sensors.

The distance of the braking zone is calculated as

$$d_{\text{brake}} = \left(\frac{v_0}{2} - v_{\text{obj}}\right) t + 0.5 \times (v_0 - v_{\text{obj}}) + d_{\text{margin}} \quad (32)$$

Where  $d_{\text{brake}}$  is the distance,  $v_{\text{obj}}$  is the object's absolute speed in the longitudinal direction,  $t$  is the time required for the autonomous vehicle to come to a full stop. And  $0.5 \times (v_0 - v_{\text{obj}})$  is a term of 0.5 second delay to account for computation time, Lidar sampling time, and mechanical parts moving time.  $d_{\text{margin}}$  is an additional distance added for safety.

$$d_{\text{brake}} = \frac{\left(\frac{v_0}{2} - v_{\text{obj}}\right) v_0}{a} + \frac{v_0 - v_{\text{obj}}}{2} + d_{\text{margin}} \quad (33)$$

The system should start emergency braking when the Lidar measured distance satisfies

$$d_{\text{measured}} < d_{\text{brake}} + \Delta d_{\text{Lidar}} \quad (34)$$

Where  $\Delta d_{\text{Lidar}}$  is the position offset of the Lidar sensor from the ego vehicle's front bumper.

### III. SIMULATION SETUP

To study the performance of different Lidar combinations and installation configurations under different dangerous scenarios, a simulation was developed. Aside from the advantage of saving time later, reducing human effort, and saving cost, a greater advantage to this project is that it does not risk human injuries or property damages. To make the simulation as accurate as possible, the artificial Lidar signal should be able to represent the actual Lidar signal accurately. The structure of the simulation is shown in Figure 17. The simulation consists of three main parts. Part 1 of the simulation is setting up the autonomous-driving vehicle and other objects. This part was discussed in section 2.5.2. Part 2 of the simulation calculates relative distance and outputs artificial Lidar signal. Finally, part 3 of the simulation is the Kalman Filter as discussed in section 2.7.1.

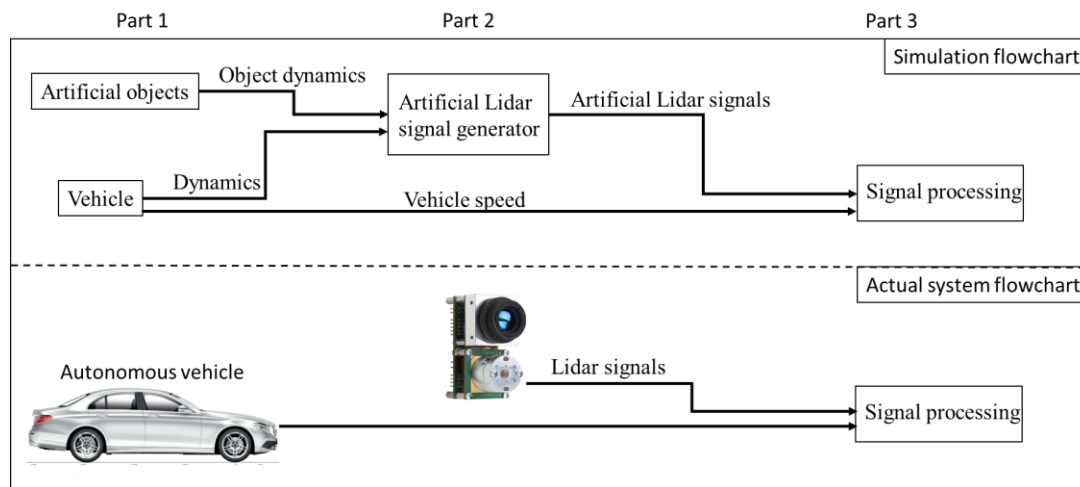
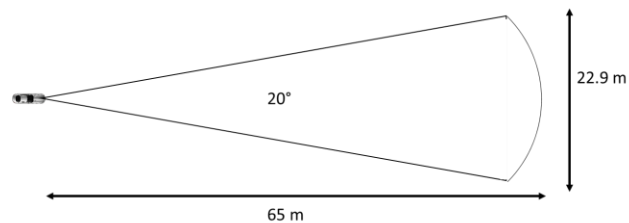


Figure 17. Simulation structure

### 3.1 Lidar Configurations

A long range 20° Lidar is always needed to detect objects at the farthest distance possible. Complementing this Lidar can be another 20° Lidar, or a 48° Lidar for more lateral detection at close range, or a 100° Lidar for wide angle within a few meters of the vehicle. Due to low detection range, it would almost be too late to take action when an object appears in the FoV of the 100° Lidar. Therefore, the 100° Lidar is neglected.

Five combinations are proposed, as shown from Figure 18 to Figure 22. In configuration 1, as shown in Figure 18, only one 20° Lidar is placed along the centerline of the car, and 1 meter behind the front bumper. This is the cheapest configuration. With a range of 65 meters, it is sufficient for detection of objects far ahead, and in front of the ego vehicle. But this configuration leaves blind spot in the areas directly in front of the car.

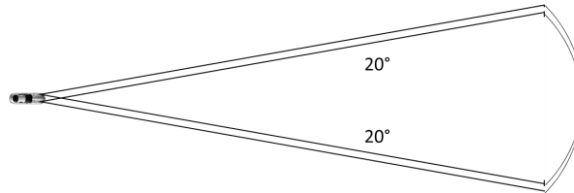


**Figure 18. Single 20° Lidar configuration.**

In configuration 2, as shown in Figure 19, two 20° Lidars are placed in parallel to each other. Each Lidar is 1 meter behind the front bumper, and 0.8 meter to the left or right of the centerline. Due to the vast area of overlapping, this combination does not

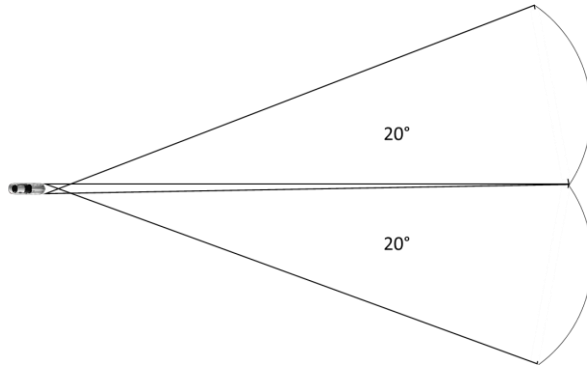


significantly increase FoV in the long range, but the greatest advantage is that it covers the area near the front of the car.



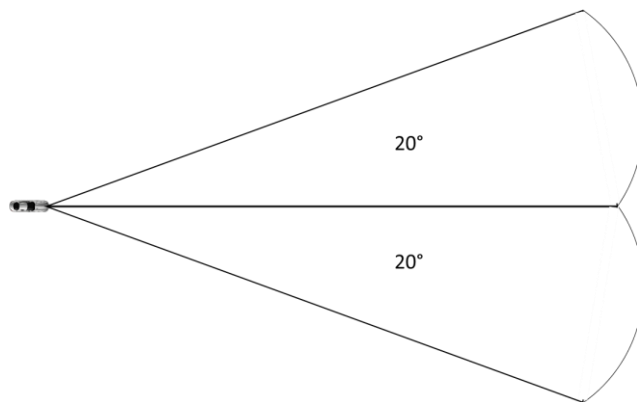
**Figure 19. Dual parallel 20° Lidars configuration.**

Configuration 3, as shown in Figure 20, is a cross-beam setup, where the Lidar on the right side points slightly to the left and captures objects on left side. The Lidar's installation angle relative to the car's longitudinal direction is set at  $\pm 10.8^\circ$  to minimize overlap of both FoVs while not letting FoVs of the Lidars to diverge until at 65 meters. This configuration increases coverage area at long range, while still maintaining sufficient coverage in the near field.



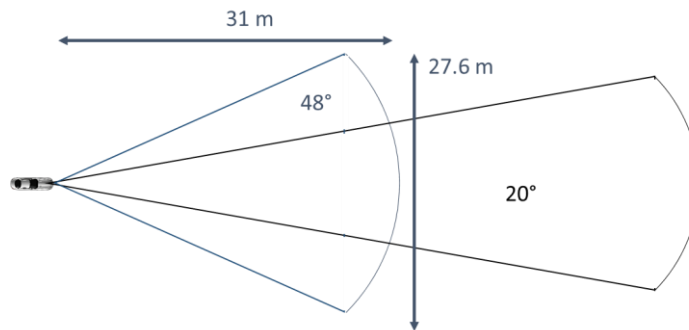
**Figure 20. Cross-beam 20° Lidars configuration.**

Configuration 4, as shown in Figure 21, is set up by placing two Lidars close to the centerline of the car, and pointing away from each other. This configuration is equivalent to using a 40° Lidar but with the range and resolution of a 20° Lidar. Similar to configuration 1, configuration 4 also has blind spots in the near field, but the area is smaller than that of configuration 1.



**Figure 21. Dual Lidars with 40° FoV configuration.**

Finally, as shown in Figure 22, configuration 5 uses a 48° Lidar to complement the 20° Lidar. The 20° Lidar is installed as in configuration 1, the 48° Lidar is also placed along the centerline, but 0.5 meter ahead of the 20° Lidar. This configuration has the widest FoV in the near-field. But due to the shorter range of the 48° Lidar, this configuration may not be useful when objects are traveling at high relative speeds.



**Figure 22. 20° and 48° Lidars configuration.**

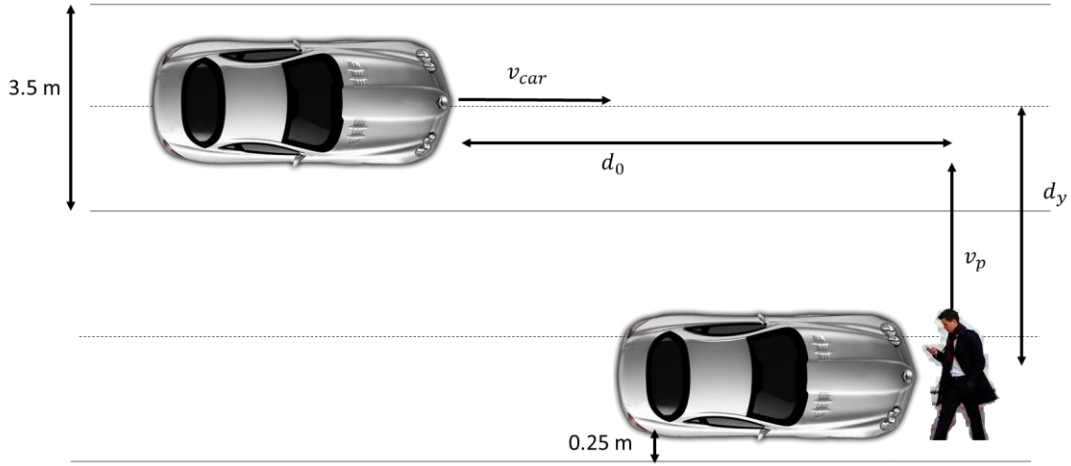
### 3.2 Driving scenarios

Four driving scenarios are created to find the best Lidar configuration. However, configuration best for one scenario may not be best for other scenarios.

#### 3.2.1 Human-crossing

Accidents involving humans are most severe, and the first driving scenario is chosen based on daily observation along the route. Even for a human driver, it is sometimes hard to detect when a pedestrian suddenly appears from behind a static object. Similarly, the

pedestrian's sight is also often obstructed because of the parked vehicles along the road, especially when the vehicles is a bus or truck. The simulation setup is shown in Figure 23.



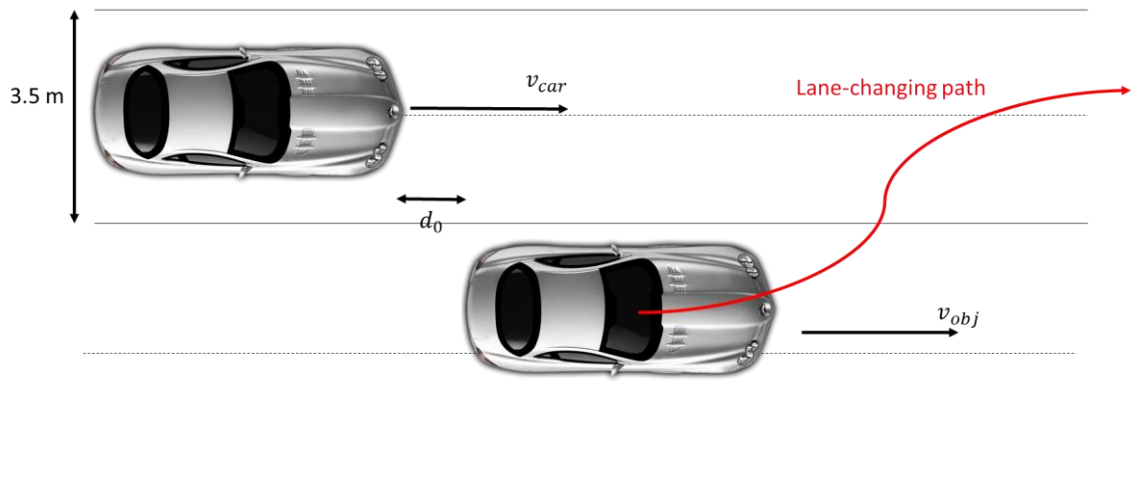
**Figure 23. Human-crossing scenario**

The car on top is the autonomous driving vehicle, traveling at speed  $v_{car}$ . Lane width  $w_l = 3.5m$  is a typical lane width for a low-speed-limit road in Germany. The person travels at  $v_p = 1.3 m/s$ , a typical speed for human.  $d_y$  is the initial lateral distance between the pedestrian and the vehicle. And  $d_0$  is the initial distance in the driving direction. Several simulations are done with different  $v_{car}$ ,  $d_0$ , and  $d_y$  values.

### 3.2.2 Same-direction lane-change

The second driving scenario, as shown in Figure 24, is very common in daily driving. A car on the neighbouring lane, traveling at the same direction, changes into the same lane as the autonomous vehicle [20]. If the initial distance  $d_0$  is too small, the autonomous

vehicle would have to brake to avoid following too closely, even if collision is not imminent. In this scenario,  $v_{car}$  is kept at 10 m/s, and  $v_{obj}$  is the object-vehicle's speed.  $d_0$  is the initial distance between the two cars. Parameters that vary for this simulation are  $v_{obj}$  and  $d_0$ .

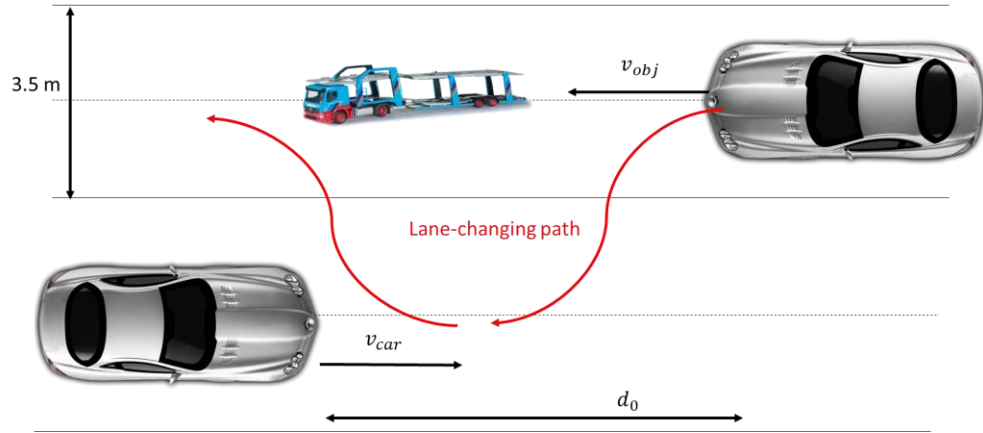


**Figure 24. Same-direction lane-change**

### 3.2.3 *Opposite-direction lane-change*

The third driving scenario is common on single-lane roads. A car would temporarily drive on the opposite-direction lane to pass a slower moving vehicle on its original lane. This scenario is very common in the plant, because on some part of the route the right lane for each direction is occupied by parked vehicles, leaving only one lane for driving. Fast-moving vehicles would then need to use the opposite lane to pass slower traffics

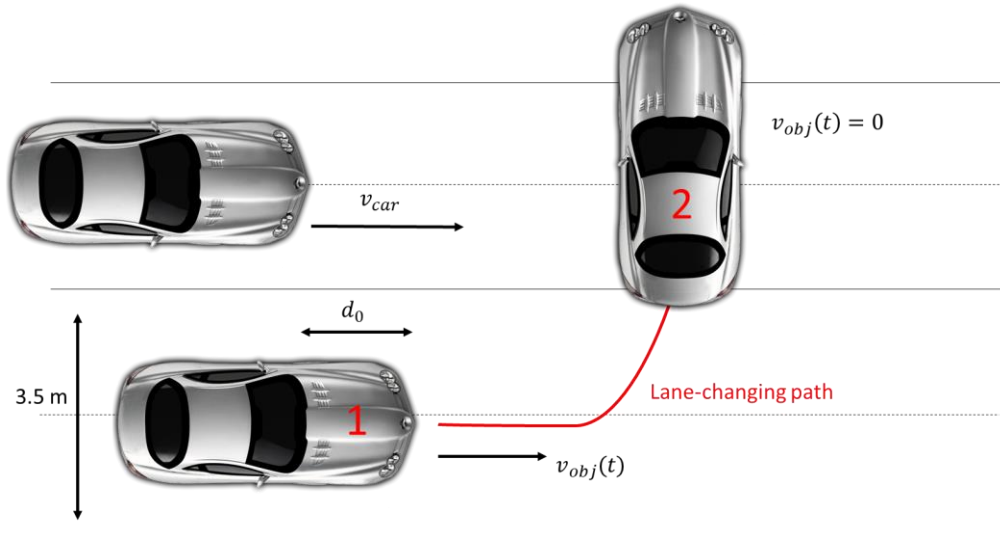
such as trucks and electric tugs. The simulation setup is shown in Figure 25 [21]. The initial distance,  $d_0$ , is the varying parameter for this scenario.



**Figure 25. Opposite-direction lane-change**

### 3.2.4 Turn-and-stop

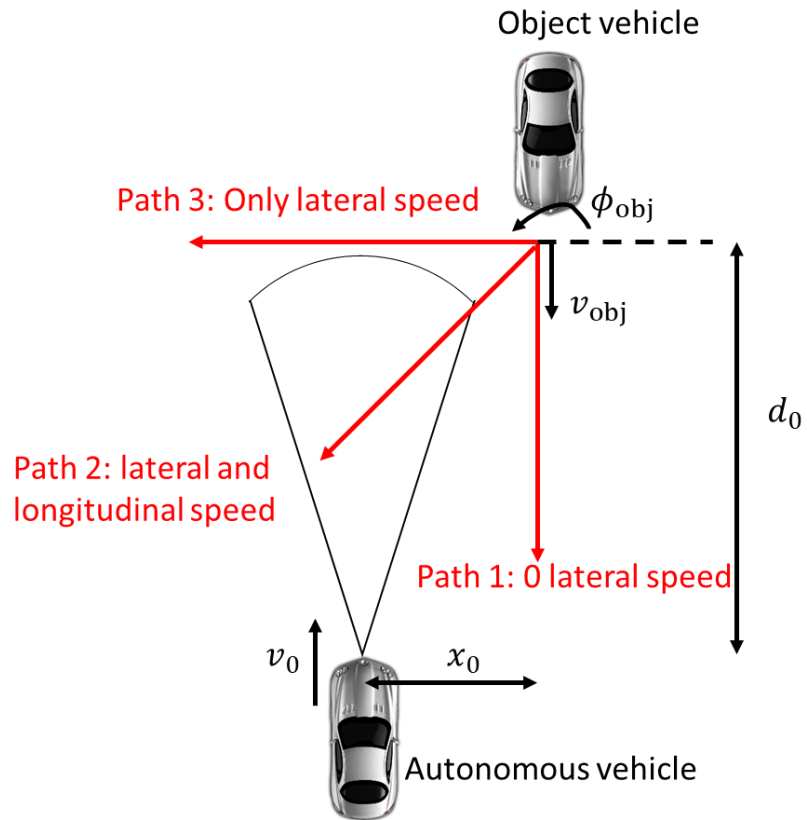
The fourth scenario, as shown in Figure 26, is created to simulate aggressive maneuvers. The object vehicle traveling on the right lane turns left in front of the autonomous vehicle, comes to a stop and occupies the left lane. This scenario usually happens when the object vehicle wants to turn left or make a U-turn but must wait for traffic on the other lanes to clear. The object vehicle's speed,  $v_{obj}(t)$ , decreases at a constant rate [22]. The initial distance,  $d_0$ , is the varying parameter. At large  $d_0$ , the scenario is similar to the human-crossing scenario. The object vehicle is almost traveling perpendicular far ahead of the autonomous vehicle. But at small  $d_0$ , the scenario simulates an aggressive lane-changing, and fast-stopping, maneuver.



**Figure 26. Turn-and-stop**

### 3.3 Scenarios for Lateral Speed Calculation

To test how accurate the lateral speed calculation is, three simple driving scenarios are used. In all scenarios, the autonomous vehicle is driving at a constant 10 m/s speed. As shown in Figure 27, the object vehicle is placed at certain distance away from the autonomous vehicle. The object vehicle travels along three different paths. In path 1, the object vehicle travels parallel to the autonomous vehicle but in the opposite direction. The lateral speed should be 0. In path 2, the object vehicle travel at 45° angle in front of the autonomous vehicle. And in path 3, the object's path is perpendicular to the autonomous vehicle's.



**Figure 27. Driving scenarios for lateral speed test.**



## IV. RESULTS

In total, 26 cases are simulated based on the four driving scenarios explained in section 3.2. The different cases of the same scenario differ in their initial value settings, such as initial speed and initial distance. Initial values for each case are shown in Table 13 through Table 16 in Appendix B. Each Lidar sensor's performance of avoiding a collision is tested individually. And the performance of configurations 2 – 4 is determined by the performance of the two Lidar sensors combined. The performance of each Lidar sensor is one of the three results – “brake too early”, “brake in time”, and “failed to prevent collision”. Braking too early, although will not cause a collision, is not favorable as it slows the traffic and reduces time-efficiency of the autonomous vehicle. To determine whether the system is braking too early, the time the system starts braking is compared to a theoretical time, which is the first time-instant an object is within the braking zone. The braking time for each Lidar sensor for each case is shown in Table 18 through Table 22 in Appendix C.

For testing the lateral speed calculation, seven cases are simulated for three driving scenarios. Initial values are shown in Table 17 in Appendix B.

## 4.1 Lidar Collision Avoidance Results

### 4.1.1 Human-Crossing Results

Table 5 and Table 6 show the performance of each Lidar sensor and each Lidar configuration under the Human-crossing scenario. The upper half of the table lists the performance of each individual Lidar sensor, while the lower half of the table lists the performance of each Lidar configuration. For each block in the tables, green color stands for “brake in time”, and yellow color stands for “brake too early”, while red color stands for “fail to prevent collision”.

**Table 5. Human-crossing performance (cases 1-6)**

Scenario	Human-crossing					
	1	2	3	4	5	6
Cases						
20° middle	Green	Green	Green	Red	Green	Green
20° left	Green	No Signal	No Signal	No Signal	No Signal	Red
20° right	Yellow	Yellow	Yellow	Green	Green	Green
20° cross left	Red	No Signal	No Signal	No Signal	Red	Red
20° cross right	Yellow	Yellow	Yellow	Green	Yellow	Yellow
20° split left	Yellow	Yellow	Yellow	Green	Yellow	Yellow
20° split right	Red	Red	Red	Red	Red	Red
48° middle	Red	Red	Red	Red	Red	Red
configuration 1	Green	Green	Green	Red	Green	Green
configuration 2	Yellow	Yellow	Yellow	Green	Green	Green
configuration 3	Yellow	Yellow	Yellow	Green	Yellow	Yellow
configuration 4	Yellow	Yellow	Yellow	Green	Yellow	Yellow
configuration 5	Green	Green	Green	Red	Green	Green

**Table 6. Human-crossing performance (cases 7-12)**

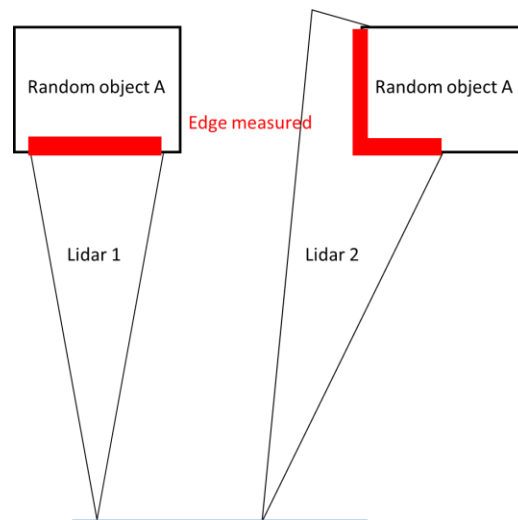
Scenario	Human-crossing					
	7	8	9	10	11	12
20° middle	Green	Green	Red	Red	Green	Green
20° left	Green	Green	No Signal	No Signal	No Signal	No Signal
20° right	Green	Green	Yellow	Yellow	Yellow	Yellow
20° cross left	Red	Green	No Signal	No Signal	No Signal	No Signal
20° cross right	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow
20° split left	Yellow	Green	Yellow	Yellow	Yellow	Yellow
20° split right	Red	Green	Red	Red	Red	Red
48° middle	Red	Red	Red	Red	Red	Red
configuration 1	Green	Green	Red	Red	Green	Green
configuration 2	Green	Green	Yellow	Yellow	Yellow	Yellow
configuration 3	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow
configuration 4	Yellow	Green	Yellow	Yellow	Yellow	Yellow
configuration 5	Green	Green	Red	Red	Green	Green

From the table, we can see that the 48° sensor is almost useless, due to its limited range.

This explains why configuration 5 behaves the same way as configuration 1. In this scenario, the single 20° Lidar is accurate and sufficient in most cases. However, as explained earlier, the blind spot of the single 20° Lidar sensor means it does not detect objects that are close to, and on the side of, the car. This is reflected by cases 9 and 10. In both cases, the pedestrian starts very close to the car when the simulation starts and is outside the FoV of the 20° Lidar. The advantage of having less or little blind spot in configurations 2, 3, and 4 can be reflected in cases 9 and 10 as well.

Even though more than one Lidar sensor are rated as “brake in time”, the accuracy of their braking time may differ. That is, how close their braking time is to the theoretical braking time. When studying the time accuracy for each Lidar sensor using values from

Appendix C, the single 20° performed best. This is perhaps because the Lidar sensor points straight ahead. The effect is shown in Figure 28. Both Lidar 1 and Lidar 2 measure the same random object A, but Lidar 2 gives a larger, and in our case, less favorable, distance reading. This may also be the reason why configuration 2 performs better than configurations 3 and 4. Configuration 2 has two 20° sensor and both are pointing in the vehicle's driving direction.



**Figure 28. Difference in distance measurement caused by Lidar configuration.**

#### *4.1.2 Same-Direction Lane-Change Results*

Table 20 shows the results for scenario 2, same-direction lane-change scenario. In this scenario, the Lidars generally performed poorly. Especially when the vehicle had little time to respond. In cases 1 and 4, the object vehicle turned into the autonomous vehicle's lane quiet aggressively, and since the Lidar sensors do not provide lateral speed

measurement, the system does not recognize the object vehicle as a potential danger that is moving towards its own lane. In case 3, larger initial distance is given between the object and the autonomous vehicle, therefore, the system has ample time to respond to the situation, and the result is rather good.

**Table 7. Same-direction lane-change performance**

Scenario	Same-direction lane-change			
	1	2	3	4
20° middle	Red	Yellow	Green	Red
20° left	Red	Yellow	Red	Red
20° right	Green	Yellow	Green	Red
20° cross left	Red	Yellow	Red	Red
20° cross right	Red	Yellow	Green	Red
20° split left	Red	Yellow	Green	Red
20° split right	Red	Red	Red	Red
48° middle	Red	Red	Red	Red
configuration 1	Red	Yellow	Green	Red
configuration 2	Green	Yellow	Green	Red
configuration 3	Red	Yellow	Green	Red
configuration 4	Red	Yellow	Green	Red
configuration 5	Red	Yellow	Green	Red

Note that in case 3, both the “20° left” and the “20° cross left” Lidar sensors fail to prevent the collision. This is not a performance problem, rather, it is because both Lidars are pointing towards the left side of the vehicle and are not intended for objects emerging from the right. In case 2, all configurations brake too early. Such false alarm occur when the object first appears in a new segment. The error is caused by the large error from the

Kalman Filter's initial value setting of relative speed. Recall that the second state variable in the KF is the relative speed. And the equation is given as

$$v_{k+1} = v_k + w_k \times T \quad (35)$$

And because the process noise is set to a very small value, therefore, it takes a few time steps for the relative speed estimation to converge to the actual value from the initial value. This error in relative speed measurement causes incorrect braking zone calculation, and therefore, false alarms.

#### *4.1.3 Opposite-Direction Lane-Change Results*

Table 22 shows the results for the opposite-direction lane-change scenario. Case 2 and 3 are set up with small initial distance values, therefore, the object vehicle also allows the Lidar sensors little time to react. More precisely, the relative speed estimation has little time to converge to an accurate value. Cases 1 and 4 are set up with larger initial distance values, and the system has more time to react. The simulation results show that the converging time for relative speed measurement must be reduced. Proposed methods are discussed in future work section.

**Table 8. Opposite-direction lane-change performance**

Scenario	Opposite-direction lane-change			
	1	2	3	4
Cases				
20° middle	Green	Red	Red	Green
20° left	Green	Red	Red	Green
20° right	Green	Red	Red	Green
20° cross left	Green	Red	Red	Green
20° cross right	Green	Red	Red	Green
20° split left	Green	Red	Red	Green
20° split right	Green	Red	Red	Green
48° middle	Red	Red	Red	Red
configuration 1	Green	Red	Red	Green
configuration 2	Green	Red	Red	Green
configuration 3	Green	Red	Red	Green
configuration 4	Green	Red	Red	Green
configuration 5	Green	Red	Red	Green

*4.1.4 Turn-and-Stop Results*

Table 21 shows the results for the turn-and-stop scenario. The Lidar configurations perform better than in the last two scenarios. This is possibly because towards the end of the object vehicle’s travel, it has almost 0 speed in the autonomous vehicle’s driving direction, and this reduces chance of getting large errors in relative speed estimation. Moreover, because the object has small relative speed and large initial distance to the autonomous vehicle, the system has more time to provide accurate calculation.

**Table 9. Turn-and-stop performance**

Scenario	Turn and stop					
	1	2	3	4	5	6
20° middle	Green	Yellow	Yellow	Green	Green	Green
20° left	Green	Green	Green	Green	Green	Red
20° right	Green	Yellow	Yellow	Green	Green	Green
20° cross left	Green	Yellow	Yellow	Green	Red	Red
20° cross right	Green	Green	Green	Green	Green	Green
20° split left	Green	Green	Yellow	Green	Green	Red
20° split right	Yellow	Green	Yellow	Green	Green	Red
48° middle	Red	Red	Red	Red	Red	Red
configuration 1	Green	Yellow	Yellow	Green	Green	Green
configuration 2	Green	Yellow	Yellow	Green	Green	Green
configuration 3	Green	Yellow	Yellow	Green	Green	Green
configuration 4	Yellow	Green	Yellow	Green	Green	Red
configuration 5	Green	Yellow	Yellow	Green	Green	Green

Results from the four driving scenarios show that the Lidar configurations are susceptible to the same problems. That is, it takes too much time for the relative speed estimation to converge to the actual value. And since the braking zone is based on the speed estimation, the braking zone also shows error. And this error either causes the system to fail to brake in time or brake too early. All configurations perform similarly under most scenarios, and they fail at the same time. A single 20° Lidar sensor has the problem of having blind spot in the near field of the car. Configurations 2 and 3 covers most, if not all, the area directly in front of the vehicle, while configuration 4 reduces the area in the blind spot. Even though configurations 3 and 4 have larger coverage area in the long range, the FoV of configuration 2 is already sufficient to cover the neighboring lanes. And as explained earlier, the Lidar distance measurement is more accurate when the



Lidar is pointing straight ahead in the vehicle’s driving direction, therefore, configuration 2 is best of the five configurations. This is also supported by the results from the four driving scenarios. Configuration 2 has at least two more “brake in time” ratings and at least one less “failed to prevent” rating than configurations 3 and 4.

#### 4.2 Lateral Speed Results

Since the best Lidar configuration is determined to be configuration 2, the lateral speed test is also conducted using this configuration. In total, 7 cases were simulated.

Results from the lateral speed calculation is shown in Table 10 and Table 11. In the tables, columns under “appear” are speed values calculated when the object enters a new segment, and columns under “disappear” are values calculated when an object exists the FoV of a segment.

**Table 10. Results from lateral speed calculation (path 1 and 2)**

Actual speed (m/s)	Path 1 parallel				Path 2 -135°			
	0				3.54		10.61	
	Appear	Disappear	Appear	Disappear	Appear	Disappear	Appear	Disappear
Min. speed (m/s)	-3.66	-1.50	2.23	0.01	2.15	2.88	6.61	9.28
Max. speed (m/s)	-3.66	0.10	11.98	0.05	4.81	5.20	13.75	11.16
Mean speed (m/s)	-3.66	-0.46	7.10	0.03	3.49	4.00	9.61	9.92
Standard deviation	0.00	0.74	4.88	0.02	0.65	0.77	2.17	0.88

**Table 11. Results from lateral speed calculation (path 3)**

Actual speed (m/s)	Path 3 perpendicular					
	5		10		10	
	Appear	Disappear	Appear	Disappear	Appear	Disappear
Min. speed (m/s)	3.19	3.08	8.32	8.57	5.57	0.78
Max. speed (m/s)	8.13	7.27	11.53	15.52	11.24	9.60
Mean speed (m/s)	4.80	5.03	9.83	10.43	8.77	6.22
Standard deviation	1.37	1.17	0.96	1.91	1.73	3.11

Results from path 1 shows that lateral speed calculation are not accurate when the object is traveling parallel to the vehicle. Because the object is traveling towards the Lidar, the relative speed is high. This not only means uncertainty between each time step is large, but also means the time that the object crosses a segment is short. Both factor lead to a large error.

Lateral speed calculated from path 2 and path 3 are acceptable. In both cases, the objects have non-zero lateral speed relative to the vehicle. Comparing the last two results from path 3, where the theoretical speeds are both 10 m/s, we can see that the first set yields better estimation results. With all other parameters same, the initial distance setting in the first set is farther than in the second set. Therefore, the object in the first set uses more time to travel across a segment. And since the lateral speed is calculated as distance over time, the error is smaller.

The test shows that accurate lateral speed estimation is generally acceptable when the actual lateral speed is not close to zero. And the farther the object is from the vehicle, the smaller the estimation error. To improve accuracy of the lateral speed estimation,

overlapping segments from two different Lidar sensors can be used. This is discussed in future work section.

## V. CONCLUSION AND FUTURE WORK

### 5.1 Conclusion

The thesis first conducted tests on the Leddartech VU8 model Lidar sensor. Results from lab tests showed that black surface is more difficult to detect and, therefore, the detection range value used in the simulation is the actual detection range of the Lidar sensor on black objects. On-board test results showed that the windshield causes significant energy loss in the laser beam. Therefore, Lidar sensors are placed outside the vehicle in each of the proposed configurations for better detection range.

A theoretical calculation showed that vehicle pitch movement is not significant to Lidar measurement and therefore, the simulation can be created using 2D dynamic models.

Five Lidar configurations were proposed and four driving scenarios were created based on real-world observations. Results from the response time of each configuration are compared to the theoretical response time. Study of the results showed that the 48° Lidar sensor is not useful under most circumstances, due to its limited range. Lidar configuration 1, with only one 20° Lidar sensor is sufficient and provide reliable readings under most circumstances except when the object will appear near the vehicle and on the side. Comparing among configurations 2, 3, and 4, configuration 3 and 4 have greater coverage area in the long range, and should theoretically perform better. But this advantage of greater coverage is not observable from the simulations because the entire road width can be covered by a single 20° Lidar, and a wider FoV is redundant. Results from the simulations do support this explanation, that the performance of configuration 2,

3, and 4, under the four scenarios, are almost identical. Configuration 2 and 3 are superior to configuration 4 due to the extra coverage in the near field. By comparing the results from the simulation, it is determined that configuration 2 is slightly better. Therefore, configuration 2 with two parallel Lidar sensors is proposed as the best configuration.

Lateral speed calculation is tested with two 20° Lidar sensor installed as in configuration 2. Results show that the estimation is generally acceptable given the coarse resolution. The estimation yields large error when the actual lateral speed is, or close to, zero. And that estimation is more accurate when the object is farther from the vehicle. Since the lateral speed is calculated as distance over time, the more time it takes for the object to cross a segment, the less the error is.

## 5.2 Future Work

One main problem with the Kalman Filter is that the relative speed estimation takes long time to converge. This cause problem when braking is necessary within a very short time of an object appearing. The Kalman Filter can be better tuned using more complex sets of data, especially the process noise variance setting for the  $v_k$  state. Another possible way is to use real-time initial value updating. That is, before an object enters a segment, the segment will have its initial value updated using information from its neighboring segment. Otherwise, the processing noise covariance can also be time varying.

Another method to achieve more accurate relative speed estimation is to model acceleration as a state variable. And the acceleration can be estimated either linearly or non-linearly using past estimations. This would not only yield more accurate estimations, but also reduce converging time.

In the simulation, the autonomous vehicle is driving along a straight path. A more realistic model would be to have the vehicle driving through curves. For that purpose, an Extended Kalman Filter can be used.

For lateral speed calculation, even though two Lidars were utilized to provide more speed estimations, the calculations were performed independently by each Lidar. However, overlapping between two Lidars means their segments' boundaries will cross each other's, and such boundaries can be utilized. This increases horizontal resolution, at least locally, and allow for more estimations value as well as more frequent estimations. Moreover, using Lidar sensors that are not parallel to each other (configurations 3 and 4) may yield better results when the object is parallel to the vehicle.

## **VI. DESIGN OF A NEW SMART PEN**

## SUMMARY

A smart pen is an electronic pen that records the user's handwriting into electronic note. The camera inside the smart pen captures image on the paper and converts it digitally. The smart pen is not only convenient to use, but also time saving.

However, there already exist many restrictions to each of the components. For example, the refill capacity and battery charge are limited due to space limitation, and the processing power is limited to avoid overheating. Adding more sophisticated components and increasing pen capability is even more challenging.

This part of the thesis proposes several pen shell designs, as well as component placement for each design. A concept design based on custom-made battery is created. Furthermore, force analysis is conducted using Finite Element Method to study displacement of moving parts, and to determine ways to reduce stress concentration. Heat transfer analysis are performed to find solution to processor heating problem.

The main contribution of the thesis is suggesting optimized design for smart pen using conventional battery, and also suggesting a design concept using highly customized battery. The thesis also contributes by proposing ways to effectively reduce maximum temperature inside the smart pen.



## 7.1 Motivation

A smart pen comes with a camera that captures what is written on a paper and converts it into digital note. Three smart pens of different brands are shown in figure. The camera placement of one of the smart pens is shown in figure. Since the pens process the camera images on board, they need a battery and an electric board inside the pen. However, due to the limit of space and weight, current smart pens share common problems. They have limited ink capacity and limited battery charge. The refills of such smart pens are usually very short with low ink capacity. Therefore, a user is required to change the refills frequently. Smart pens on the market also use small batteries, and less charge means more frequent recharging. Both add to the inconvenience of smart pens.



**Figure 29. Popular smart pens available on the market.**



**Figure 30. Infrared camera inside a smart pen**

The idea of this thesis is to design a smart pen with greater capability, while tackling the abovementioned issues of a typical smart pen. A pen with more, and better, cameras will have more function and be more convenient to its user, while reducing the frequency for refill change or battery recharge reduces the inconvenience caused by common smart pens. The aim is to deliver a brand new smart pen that includes a dual-camera setup for more image capturing ability, a battery with greater charge, a refill with higher ink capacity, and a processor with greater power.

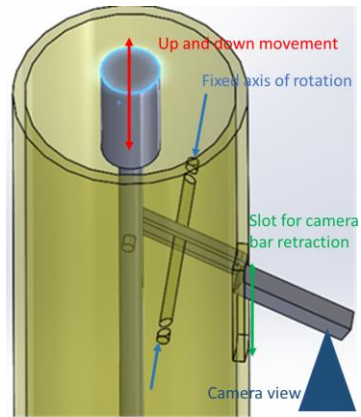
## **7.2 Design**

The first idea that came up with the new smart pen is to include two cameras. One camera would be placed towards the tip of the pen as in a normal smart pen. The function of the first camera is to capture hand writings and convert them into digital note. The second camera has a larger field-of-view, and captures a large section of the paper, if not the whole piece of paper, the user is writing on.

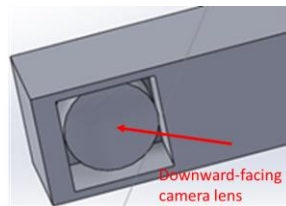
Due to the intended functionality of the second camera, it is ideal to place it on the end of the pen, so it has a bird's-eye-view of the paper. However, the hand holding the pen may block the view of the camera, so the camera should extend some distance away from the shell of the pen.

The camera should not be extended at all times to avoid accidentally damaging it. Therefore, it is desired that camera only extend when the pen is being used. A pen with a clicking mechanism requires the user to click only once to either extend the refill or retract it. A user may, however, find it frustrating to have to act once to extend the refill and act another time to extend the camera. Letting both the camera and the refill extend under one action is time saving and convenient. Therefore, the mechanism of the camera and the refill should be connected.

Figure 31 shows the refill-camera extension mechanism. The button at the end of the pen is like any clicking mechanism pen, except that it is connect to the camera bar through a joint. The camera bar is a lever about a fixed axis. One end of the bar is attached to the refill. The other end has the downward-facing camera mounted, as shown in Figure 32. As the refill moves downward, it pulls one end of the camera bar with it. The other end of the bar is lifted and extends outside the pen's shell. When the camera is retracted, it will be protected and not affect the appearance of the pen.

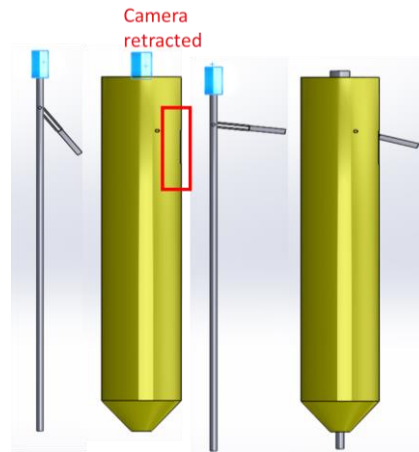


**Figure 31. Mechanism for refill and camera extension.**



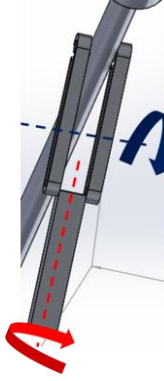
**Figure 32. Downward-facing camera.**

Figure 33 shows the comparison between when the camera bar is extended and retracted.



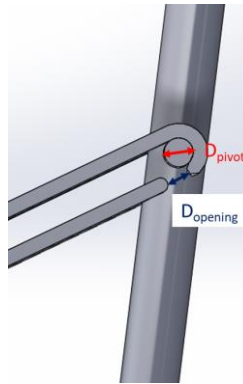
**Figure 33. Comparison on retracted and extended camera bar.**

The camera bar has a large aspect ratio. The length of the bar is relatively long compared to the cross-section dimension of the bar. If the camera bar is not stiff, the image captured may be blurred. To increase stiffness, the first segment of the original camera bar is modified into two parallel bars, and together with the second segment, form a Y-shape bar, as shown in Figure 34. This change increases stiffness of the camera bar, and reduces bending moment in the two directions shown in the figure.



**Figure 34. Y-shaped camera bar.**

Attaching the camera mechanism to the pen refill may cause additional work for the user when he/she changes the refill. To avoid this inconvenience, an opening is added to the camera bar, as shown in Figure 35. The dimension of the opening, labeled  $D_{opening}$ , is slightly smaller than the diameter of the pivoting point,  $D_{pivot}$ . This allows the refill to be securely held to the camera bar during normal use, but also easily detachable upon a light pull.

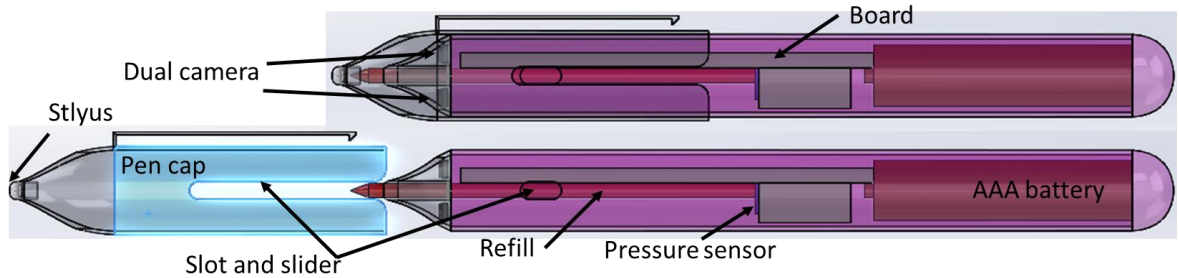


**Figure 35. Opening for easy refill removal.**

Even with the reinforced camera bar, the camera will not be perfectly stable. To guarantee stability of the camera and reduce the possibility of the camera being damaged, another method is chosen to place a wide-angle camera inside the pen at all time. This requires another opening window for the camera.

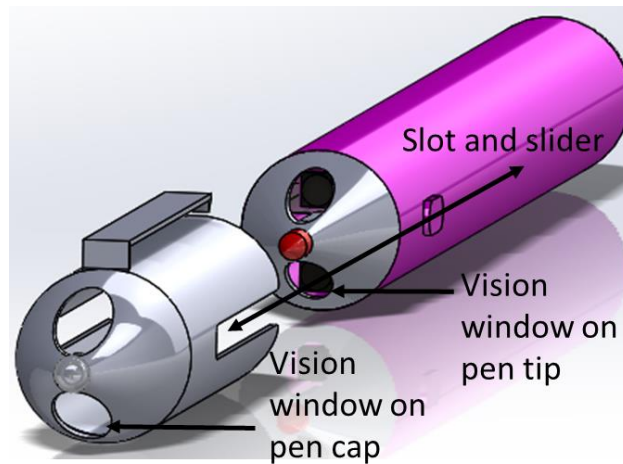
Following this idea, a new pen design is created as shown in Figure 36. Both cameras are close to the tip of the pen, and each camera has its own vision window on the pen tip. A stylus is added at the tip of the pen cap. The camera should be functional when the user is using the stylus. Therefore, two vision windows are also added on the pen cap. A slot is added on the pen cap and a slider on the pen shell. This design guarantees that the vision windows on the cap will always align with the window on the pen tip. To maximize the length, and therefore, capacity, of the refill, the battery is placed at the very end of the pen. The board is also placed in the configuration to maximize the refill's length. The outer diameter of the pen is 13.2 mm and the length without the refill is 131 mm.

In addition to the abovementioned features, a pressure sensor is installed. When the pen is pressed against a piece of paper, the force on the refill will transfer to the pressure sensor and activate the cameras.



**Figure 36. Side view of design I.**

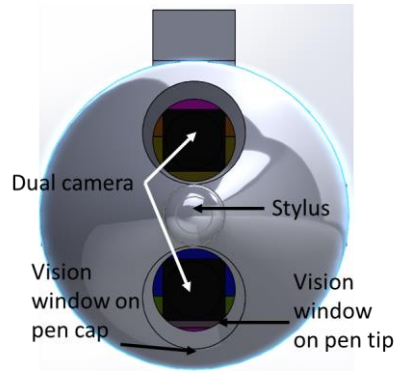
Both windows are at the tip of the pen, so the hand holding the pen will not obstruct the vision of the cameras.



**Figure 37. Design I appearance.**



As shown in Figure 38, the vision window on the cap has a slightly larger dimension compared to that of the vision window on the pen tip because there is gap between the cap and the tip in the longitudinal direction.



**Figure 38. Front view of design I.**

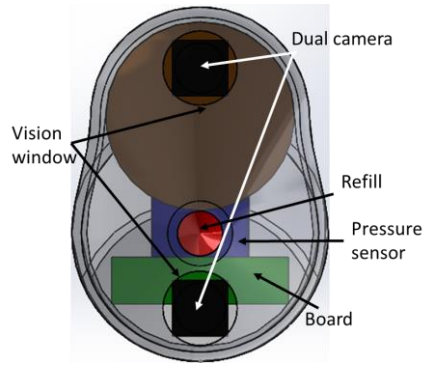
This smart pen design also has its drawbacks. With the battery, the heaviest component inside the pen, placed at the end of the pen, the center of gravity of the entire pen is shifted backwards. The pen, therefore, has the tendency to flip over. While a person may not experience discomfort at the beginning, he/she may be soon get annoyed by the increased effort required to keep the pen upright. Besides discomfort, the tendency of the tip to lift up may also be affect a person's handwriting.

One inspiration to solve this problem came from the Beluga airplane, a highly modified freighter by Airbus. The Beluga is designed to carry airplane fuselages that are wider than the original width of the plane it is based on - the A300 plane. The picture on the right in Figure 39 shows the plane carrying a section of the fuselage of an Airbus A350, which has a larger diameter than the A300.

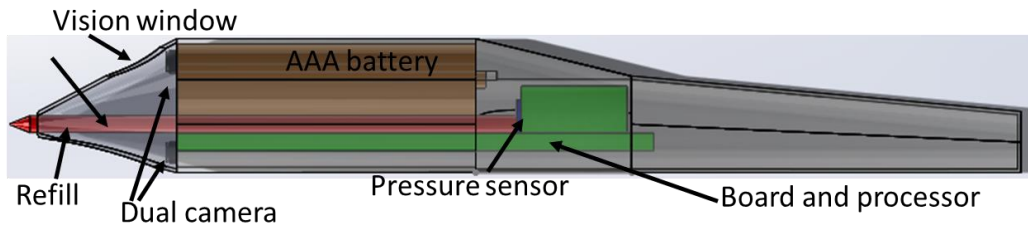


**Figure 39. Airbus Beluga freighter**

Design II, as shown in Figure 40, is similar to the design of the Beluga plane. The upper part of the pen is expanded for the placement of an AAA battery. As shown in Figure 41, the refill is placed between the board and the battery. In this way, the end of the pen is empty and the center of gravity is moved much closer to the tip of the pen. The maximum height of the cross-section area is 18.5 mm and the widest part is 13.2 mm



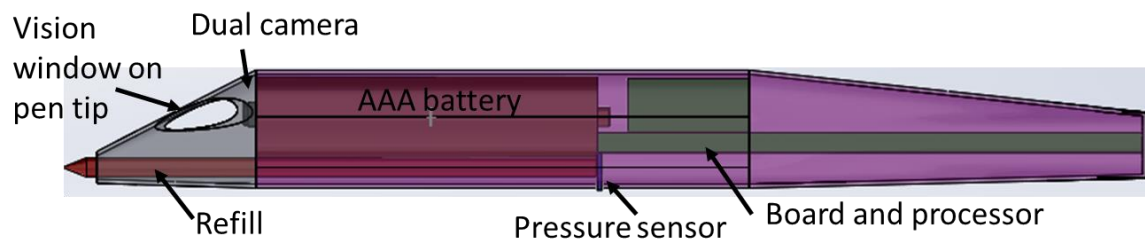
**Figure 40. Design II front view.**



**Figure 41. Design II side view.**

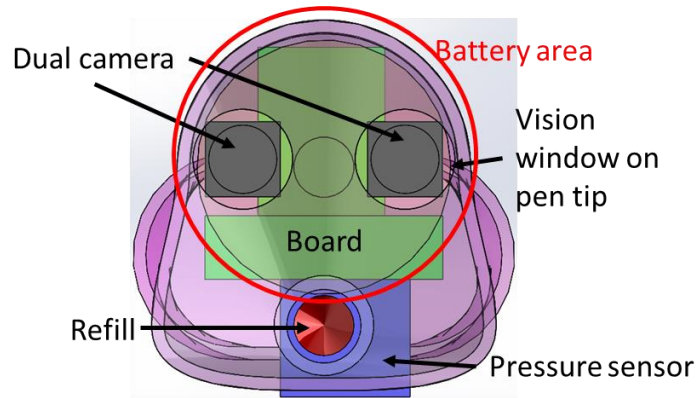
This design also has its own drawbacks. The grip part of the pen is too bulky for a person to hold comfortably. Such grip makes it hard for a person to maneuver the pen with his/her index finger and thumb without having to apply additional force. Hand writing may be affected. Long-time writing will most likely cause discomfort. Moreover, holding such bulky grip in hand will give the user a very different feel than holding a normal pen. A little difference in feel is desired to provide the user with a unique feeling of the pen, but too much difference is not desired.

To reduce the size of the grip part, a new design is created, as shown in Figure 42. In this design, the board is moved behind the battery. Because the board has smaller density compared to the battery, placing the board behind will not affect the center of gravity too much. This design reduces the cross-section dimension of the grip part while maintaining a relatively low center of gravity profile. Moreover, with nothing behind the refill, the length of the refill can be longer and, therefore, the capacity of the refill can also increase.

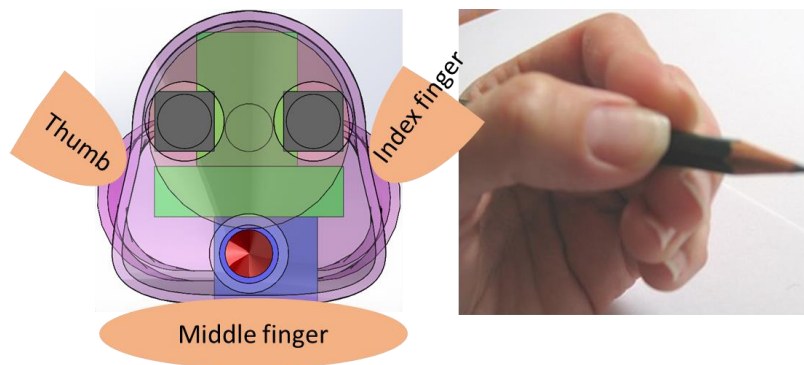


**Figure 42. Design III side view.**

As shown in Figure 43, the bottom contour of the smart pen is in the shape of a curve. The left and right sides of the pen grip area also have curved contours. These are designed based on the observation of the way a person normally holds a pen. As shown in Figure 44, the curved areas on the sides allow the fingers to rest naturally. The bottom area is not perfectly flat, so when the pen sits on the middle finger, the pressure the finger feels is reduced. It also makes the pen easier to roll on the middle finger than a flat surface.



**Figure 43. Design III front view.**



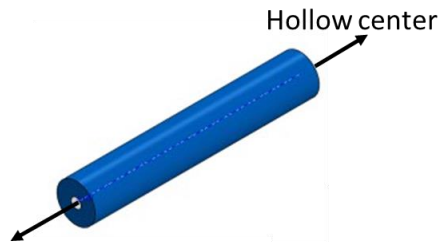
**Figure 44. Design III curved contour for finger placement.**

Compared to the last design, the maximum width in the cross-section area remains at 13.2 mm, but the height is reduced to 14.2 mm. The smaller grip area makes it easier for a user to hold.

For the use of conventional AAA batteries and a single-piece electric board, design III is the best solution compared to designs I and II.

### 7.3 Smart Pen Concept Based on Customized Components

For this specific project, one battery supplier has proposed an unconventional way of manufacturing batteries. As shown in Figure 45, the battery has hollow center that allows the refill to go through.



**Figure 45. Battery with hollow center.**

The pen is also designed with a new board with a lower profile. Without a tall processor extruding out of the board, it is now nearly flat.



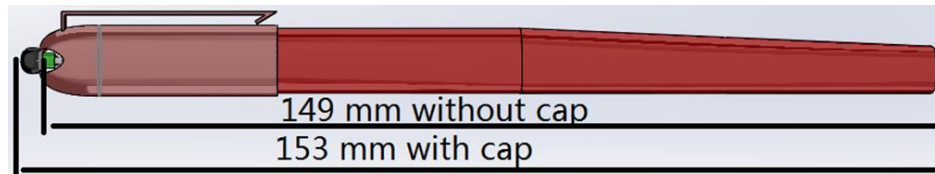
**Figure 46. Side view of concept pen.**

The Uni-ball Eye, as shown in Figure 47, is a very commonly used pen. The length of the Uni-ball is 140 mm and the diameter is 9.8 mm. As we can see in Figure 48 and Figure 49, the dimensions of the smart pen, 149 mm x 10.98 mm, are not much larger than a Uni-ball. The diameter of the custom-made battery is smaller than that of an AAA battery, therefore the diameter of this smart pen design is smaller than any previous design with an AAA battery. Another comparison can be made with the Montblanc

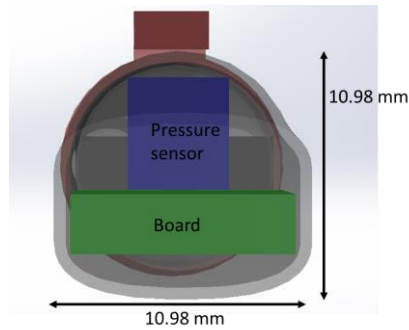
Meisterstück, one of the most popular high-end pen. The Meisterstück is a heavy pen made from metal and has large dimensions. The smart pen is thinner than a Meisterstück. One thing to note is that, with this configuration, the shell of the pen can be designed to have a round cross-sectional shape. However, some non-symmetry in the design is desired, as the orientation of the pen is then easily observable. The shape can guide the user to hold the pen in the correct orientation so that the two cameras will scanning the paper from the correct positions.



**Figure 47. Uni-ball Eye.**



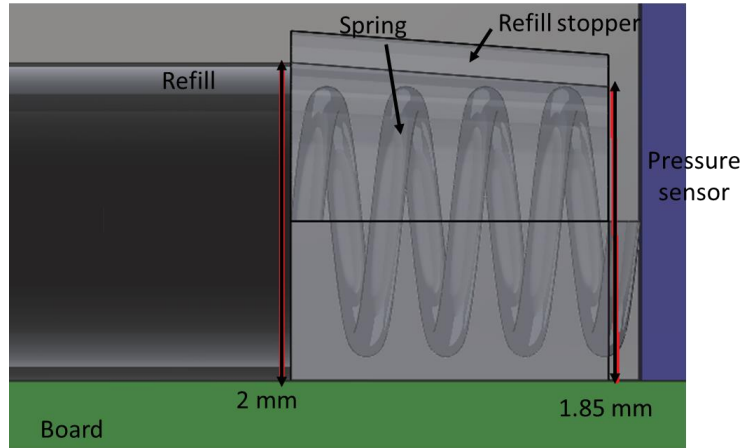
**Figure 48. Side view of concept pen appearance.**



**Figure 49. Rear view of concept pen version 1.**

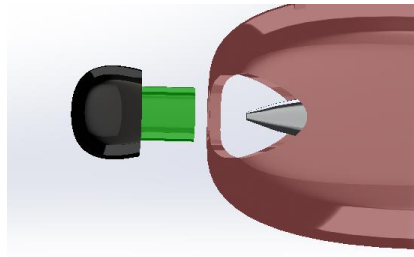
In the previous designs, the refill is in direct contact with the pressure sensor. But when the pen is accidentally dropped onto the floor, or when the user applies too much force, the refill may damage the pressure sensor. An additional to this new design is a refill stopper between the refill and the pressure sensor, as shown in Figure 50. The stopper is a tunnel with decreasing inner diameter, with the larger end having the same diameter as the refill. A spring inside the tunnel connects the end of the refill to the pressure sensor. The decrease in diameter means that the refill will experience increased resistance as it travels backwards. And at some point, the refill will not move back further. When the refill is pressed against the paper, it compresses the spring, which in turn exerts a force on the sensor. The spring will not exert extensive force onto the pressure sensor because the displacement is limited by the tunnel. The spring constant can be selected such that at the maximum allowed displacement by the tunnel, the pressure of the sensor will not exceed its limit.



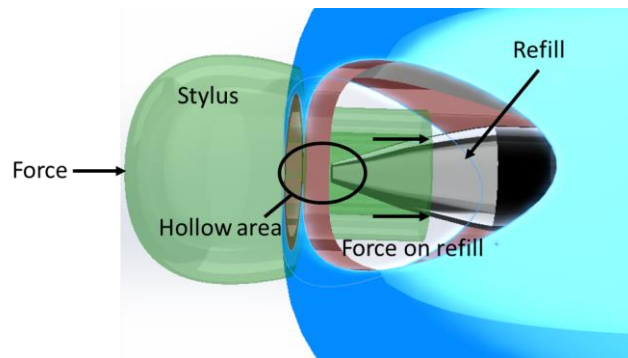


**Figure 50. Refill stopper.**

As shown in Figure 51, the stylus is a part that can translate relative to the cap. Therefore, pressing the stylus against a touch screen activates the pressure sensor. As shown in Figure 52, the force from the stylus is not applied directly onto the tip of the refill, since that may damage the roller ball. Instead, the stylus has a hollow area for the refill tip and exert the force onto the cone of the refill.

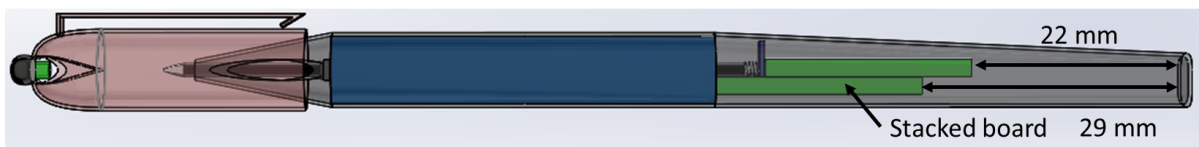


**Figure 51. Stylus tip on pen cap.**



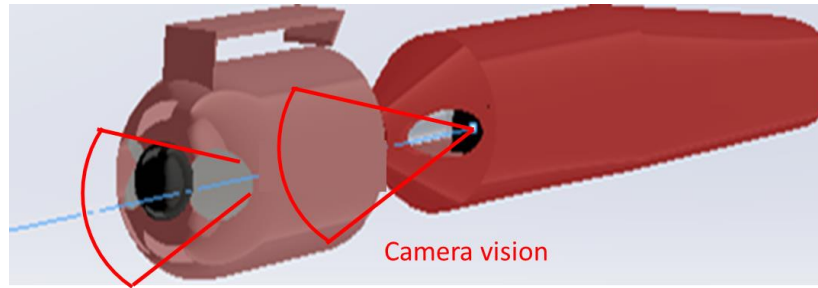
**Figure 52. Force transfer from stylus to refill.**

To shorten the length of the pen, the original one-piece long electric board can be produced into two pieces, and one piece can be stacked on top of the other. The decreased length of the pen not only lowers the center of gravity, but also reduces total weight.



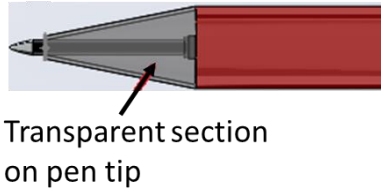
**Figure 53. Side view of stacked board.**

As with previous designs, the concept has two vision windows, as shown in Figure 54.

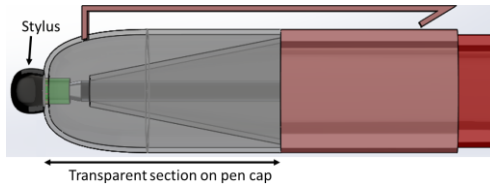


**Figure 54. Vision through window.**

The vision windows cannot be made too big because they would reduce the structure strength of the pen cap. But a window limits the field-of-view of the camera. Also, having openings on the pen shell will not make the pen water-resistant. The pen has electronics inside, and a coffee spill is sometimes inevitable. Based on these two concerns, a new design of the pen tip and pen cap is shown in Figure 55 and Figure 56. Large sections of the tip and cap are made transparent. This gives the camera a much larger field of view. And since the pen is nearly fully enclosed, it has some water-resistant capability. The transparent section can be made out of glass to give users a high-end feeling, or be made out of transparent plastics, such as acrylic, to reduce weight.

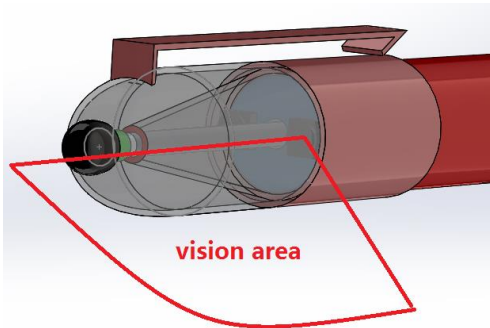


**Figure 55. Transparent pen tip.**



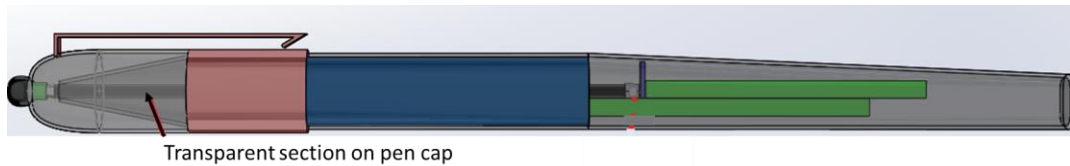
**Figure 56. Transparent pen cap.**

As seen in Figure 57, with transparent windows, the cameras have a much wider field of view.



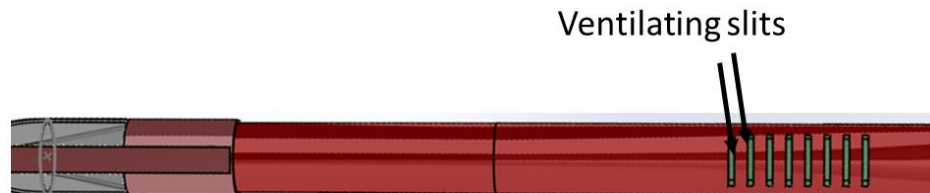
**Figure 57. New vision area of concept pen.**

After a few modifications, the concept pen design is shown in Figure 58.



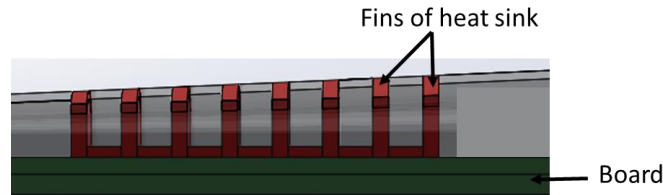
**Figure 58. Side view of concept pen with a few changes.**

The camera data is processed on-board, but the confined space makes it difficult for heat to dissipate. The maximum temperature inside the pen should be limited because high temperature may cause discomfort to the user. As seen in Figure 59, several ventilating slits are added towards the end of the pen. The slits allow convection between the hot air inside the pen shell and the ambient air. The slits are placed at the end of the pen because when the pen is upright, the end will be higher and low-density hot air will flow towards the end.

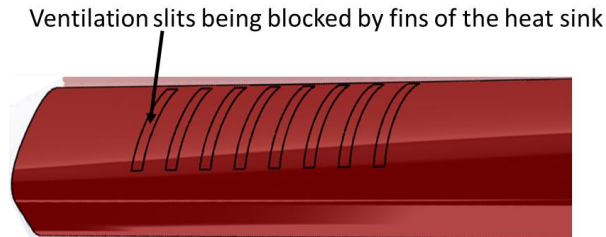


**Figure 59. Ventilation slits on concept pen.**

As mentioned earlier, the pen should have some water-resistant capability. Opening ventilation slits would weaken this capability. One solution to this problem is to block the slits with heatsinks. As seen in Figure 60, each fin of the heatsink can fit into the a slit. The fins are in direct contact with the ambient air and thus provide better heat transfer. The appearance of the pen with the heatsink is shown in Figure 61.



**Figure 60. Heatsink in pen.**

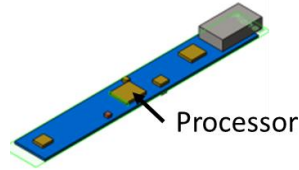


**Figure 61. Appearance of concept pen with heatsink.**

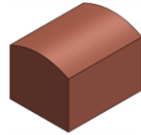
One drawback with the heatsink and slits idea is that it increases weight and raises the center of gravity of the pen.

Since most heat comes from the heat-generating processor, a smaller heatsink directly on top of the processor may be a better solution in terms of weight saving, although the heat transfer effect would not be as good. But since the shell of the pen is going to be made from metal, it is still a good medium for heat exchange between the heatsink and the ambient air.

The processor is shown in Figure 62. A metal heatsink out of copper is shown in Figure 63. The curved surface of the heatsink will be in contact with the metal pen shell, and the flat surface will be in contact with the processor.



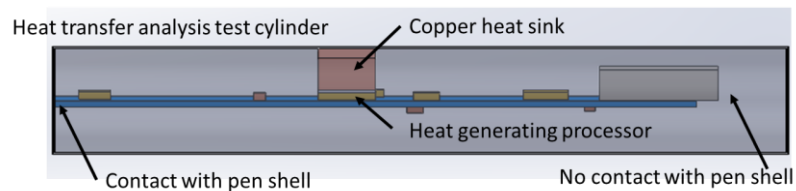
**Figure 62. Electric board used inside smart pen.**



**Figure 63. Small copper heatsink.**

#### **7.4 Heat Transfer Analysis**

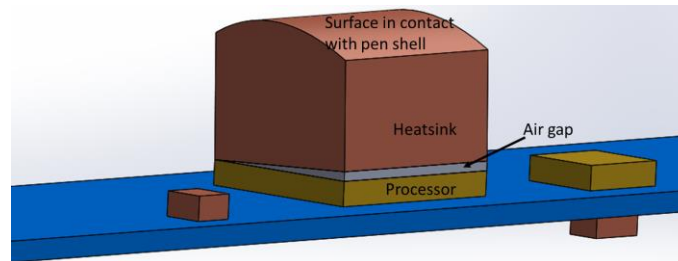
Due to internal component movement and misalignment, the surface of the heatsink may not align perfectly with the processor's surface, which would decrease the heat transfer effect. The effect the misalignment angle has on the maximum steady-state temperature is studied. A test cylinder, as shown in Figure 64, is created to simulate this effect [23].



**Figure 64. Test cylinder for misalignment analysis.**

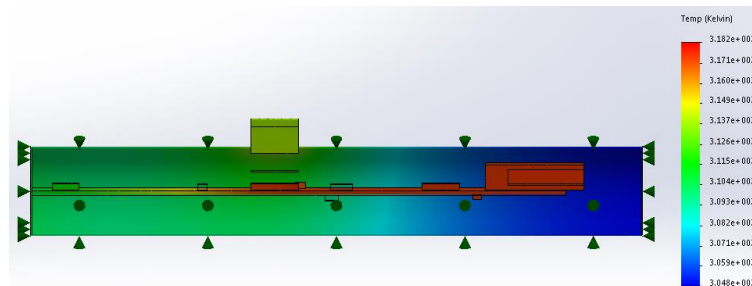
As seen in Figure 65, the air gap is modeled as a wedge between the heatsink and the processor. This model is chosen based on the assumption that there is no air flow within

the pen shell, so convection is neglected. The sharp angle of the wedge reflects the angle of the misalignment. The power of the processor is set at a constant 2 Watts. And the ambient temperature outside the pen shell is set at 30° C. In the analysis, the angle of the misalignment was increased from 0°, which is perfect alignment, up to 15°.



**Figure 65. Heat analysis setup.**

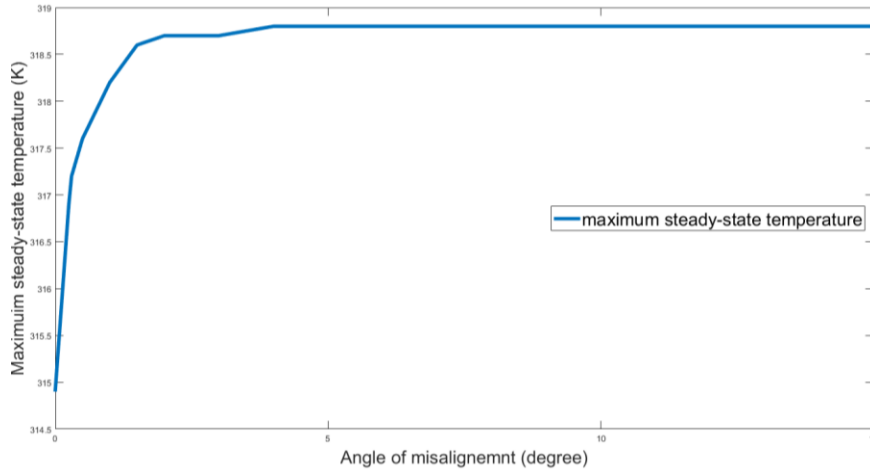
One example of the heat transfer analysis is shown in Figure 66.



**Figure 66. One result from heat analysis.**

The results are plotted in Figure 67. The maximum steady-state temperature increases dramatically as soon as there is a misalignment. But it doesn't increase further after the angle increases to 4°. At this point the air gap between the heatsink and the processor is essentially an insulator and heat generated by the processor is conducted through the electric board or through air inside the pen shell. The maximum steady-state temperature is around 45° C.



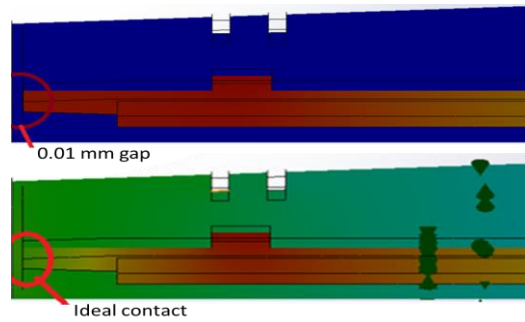


**Figure 67. Maximum steady-state temperature vs. angle.**

From Figure 66 we can also see that the left end of the board, which is in contact with the cylindrical shell, is  $8^{\circ} C$  cooler than the right end of the board, which is not in contact with the pen shell.

To further study the effect of heat conduction through contact, a simplified model of only a board and a shell is used. The result of contact and no-contact of the same end of the board is compared in Figure 68. Result shows that when there exists a 0.01 mm gap, the maximum temperature is 305.5 K, but only 296.8 K when there is contact.

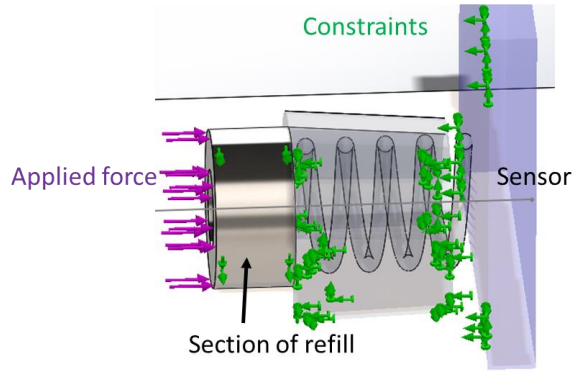
This means that, for better heat dissipation, the pen and its components, should be designed with large contact area.



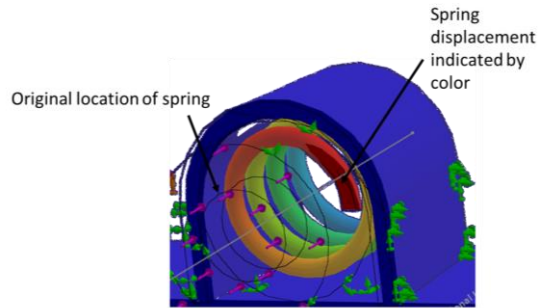
**Figure 68. Comparison between contact and no-contact setup.**

### **7.5 Force analysis**

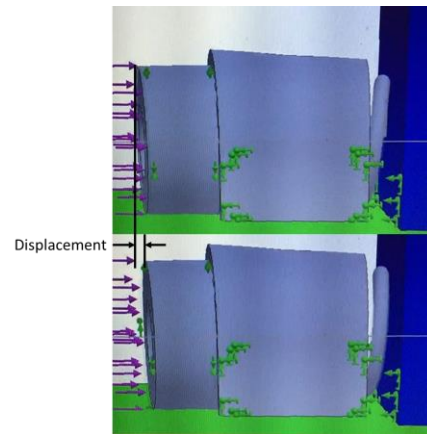
The only part inside the pen that is a moving part is the refill stopper. To study the displacement of the spring under compressive force, a simulation is conducted using Finite Element Analysis [24]. As shown in Figure 69, force is applied on a refill, which is only a short section of its original length. The displacement of the spring, as shown in Figure 70 and Figure 71, helps to determine the preferred spring constant.



**Figure 69. Finite Element Analysis setup.**



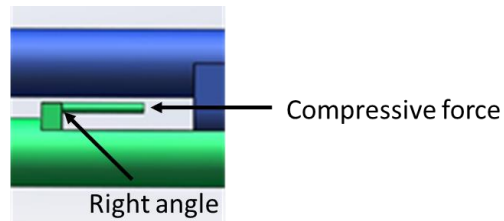
**Figure 70. Displacement of spring.**



**Figure 71. Displacement of refill shown in animation.**

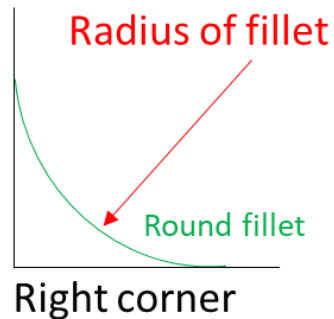
Finite Element method is also used to study stress within the parts. The part shown in Figure 72 was created by Georgia Tech student Yang Yujia. This part has a thin pole that

is used to push the refill. At the bottom of the pole is a  $90^\circ$  angle with a high stress concentration under compression. We wish to study how much a round fillet reduces the maximum stress.



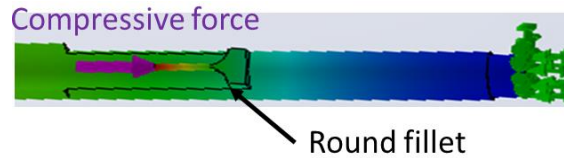
**Figure 72. Part with high stress concentration.**

A fillet is defined by the radius of its curvature, as shown in Figure 73.



**Figure 73. Round fillet.**

As shown in Figure 74, the round fillet is added to the pole. Results of the maximum Von Mises stress is shown in Table 12. As the results show, adding a 1.0 mm fillet reduces the maximum stress by 15%.



**Figure 74. Force analysis with 1 mm fillet.**

**Table 12. Fillet analysis results.**

	No fillet	0.5mm fillet	1.0mm fillet	1.2mm fillet
Max Von Mises Stress (N/m <sup>2</sup> )	6.94e006	6.152e006	5.898e006	5.627e006

## 7.6 Conclusion and Future Work

The new smart pen design has increased battery capacity, larger refill, and better cameras, all of which provides more convenience to the users. The thesis first explored a way of extending a second camera outside the pen shell, at the end of the pen. It is determined that with this design, the stiffness of the camera bar can cause a problem, as the instability of the camera can cause the image to blur. With the second camera being placed towards the tip of the pen, three smart pen designs were proposed using conventional batteries, and one concept design was proposed with a highly-customized battery. Out of the first three designs, design III is determined to be the best solution. It maintains a low center of gravity while limiting the size of the grip part.

With the custom-made battery, the concept pen was made thinner than the first three designs. Additional features were also added to the concept pen. These features are also applicable to the earlier designs. A refill stopper was added between the refill and the pressure sensor to protect the sensor from excessive force. The pen tip area and pen cap were made transparent to allow a wider field-of-view of the camera. Furthermore, an alternative way of stacking two shortened electric boards was used to reduce pen length and lower the center of gravity. Moreover, several methods for heat dissipation was also proposed and the use of a small heatsink directly on top of the processing unit was chosen as the best solution.

Heat analysis was conducted to study the effect of misalignment between the heatsink and the processor. Similarly, direct contact between the board and the pen shell was also studied. Results from the heat analysis showed that contact surface should be increased during the design phase, and a perfect alignment is critical during assembly.

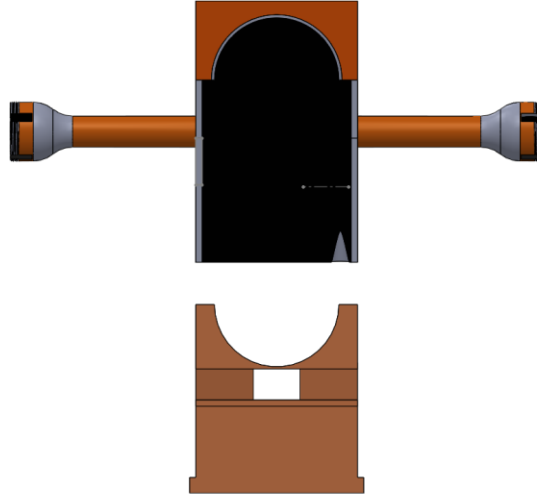
Force analysis was performed on the refill stopped. The results showed the amount of displacement of the spring under a given compressive force and provided guideline to the selection of spring constant.

One important thing for future work is a more thorough thermal analysis. The analysis should be conducted using the entire smart pen CAD, instead of just a test cylinder or part of the assembled CAD [25].

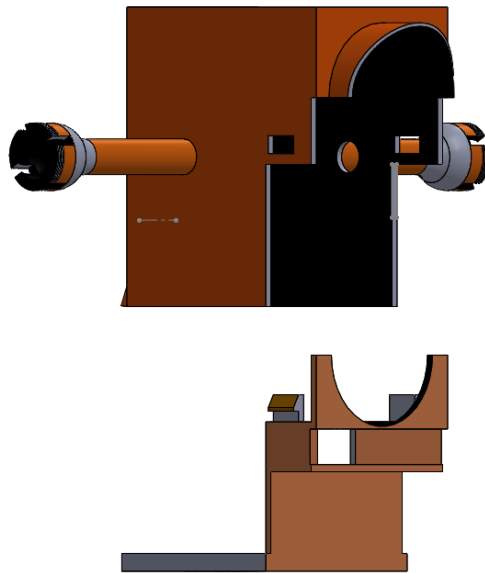
Camera vision should be tested using the prototyped pen. And the effect of the transparent material on the camera signal should be studied.

Finally, smart pen prototypes should be manufactured, possibly using rapid prototyping machines. The prototypes should be handed to real people to use, and feedback in terms of aesthetics, handling, feel, and comfort should be collected. The designs should be modified based on the feedback. Several iterations are required for an acceptable final design.

**APPENDIX A. CAD MODEL**

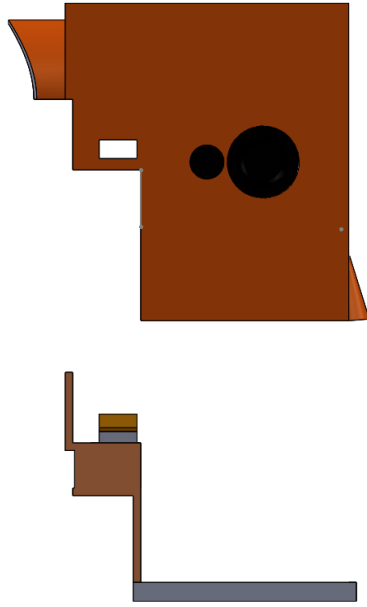


**Figure 75. Front view of case**



**Figure 76. Lidar case**





**Figure 77. Side view of case**

## APPENDIX B. INITIAL VALUES FOR SIMULATION CASES

**Table 13. Initial values for human-crossing scenario**

Cases	$v_{\text{car}}$ (m/s)	$d_0$ (m)	$d_y$ (m)
1	10	3.25	29.6
2	10	3.25	26.6
3	10	3.25	23.4
4	10	3.25	20.6
5	8	3.25	23.6
6	6	3.25	23.6
7	8	3.25	29.6
8	6	3.25	29.6
9	10	3.55	23.6
10	10	3.85	23.6
11	10	3.55	29.6
12	10	3.85	29.6

**Table 14. Initial values for same-direction lane-change scenario**

Case	$d_0$	$v_{\text{obj}}$
1	5	9
2	10	9
3	5	10
4	0	11

**Table 15. Initial values for opposite-direction lane-change scenario**

Cases	$d_0$
1	65
2	60
3	55
4	70

**Table 16. Initial values for turn and stop scenario**













Cases	$d_0$
1	25
2	30
3	35
4	20
5	15
6	10

**Table 17. Initial values for lateral speed calculation**

Cases	$v_{obj}$	$d_0$	$x_0$	$\phi_{obj}$
1	5	80	30	180
2	10	80	30	180
3	10	50	30	180
4	15	80	3	270
5	5	80	3	270
6	5	80	15	225
7	15	80	30	225

## APPENDIX C. BRAKING TIMES FOR EACH SIMULATION CASE

**Table 18. Braking times for Lidar sensors in human-crossing scenario (cases 1-6)**

Scenario	Human-crossing					
	1	2	3	4	5	6
Simulations						
Theoretical braking time (s)	0.9	0.6	0.3	0	0.8	1.5
20° middle braking time (s)	0.9	0.6	0.4		0.8	1.5
20° left braking time (s)	0.9					1.8
20° right braking time (s)	0.5	0.1	0	0	0.6	1.5
20° cross left braking time (s)	2.4				2.4	2.4
20° cross right braking time (s)	0.5	0.1	0	0	0.2	0.7
20° split left braking time (s)	0.4	0.1	0	0	0.2	0.8
20° split right braking time (s)	1.8	1.8	1.8	1.7	1.8	1.8
48° middle braking time (s)	2.1	1.8	1.5	1.2	1.9	2.4
Theoretical braking time (s)	0.9	0.6	0.3	0	0.8	1.5
configuration 1 braking time (s)	0.9	0.6	0.4		0.8	1.5
configuration 2 braking time (s) 	0.5 	0.1 	0	0 	0.6 	1.5
configuration 3 braking time (s) 	0.5 	0.1 	0	0 	0.2 	0.7
configuration 4 braking time (s) 	0.4 	0.1 	0	0 	0.2 	0.8
configuration 5 braking time (s)	0.9	0.6	0.4	1.2	0.8	1.5

**Table 19. Braking times for Lidar sensors in human-crossing scenario (cases 7-12)**

Scenario	Human-crossing					
	7	8	9	10	11	12
Simulations	7	8	9	10	11	12
Theoretical braking time (s)	1.5	2.5	0.3	0.3	0.9	0.9
20° middle braking time (s)	1.5	2.5			0.9	0.9
20° left braking time (s)	1.5	2.5				
20° right braking time (s)	1.3	2.5	0	0	0.5	0.5
20° cross left braking time (s)	2.4	2.5				
20° cross right braking time (s)	0.9	1.7	0	0	0.5	0.5
20° split left braking time (s)	0.9	2.4	0	0	0.4	0.4
20° split right braking time (s)	1.9	2.6	2		2	2.2
48° middle braking time (s)	2.6	3.5	1.5	1.5	2.1	2.1
Theoretical braking time (s)	1.5	2.5	0.3	0.3	0.9	0.9
configuration 1 braking time (s)	1.5	2.5			0.9	0.9
configuration 2 braking time (s)	1.3	2.5	0	0	0.5	0.5
configuration 3 braking time (s)	0.9	1.7	0	0	0.5	0.5
configuration 4 braking time (s)	0.9	2.4	0	0	0.4	0.4
configuration 5 braking time (s)	1.5	2.5	1.5	1.5	0.9	0.9

**Table 20. Braking times for same-direction lane-change scenario.**

Scenario	Same-direction lane-change			
	1	2	3	4
Simulations				
Theoretical braking time (s)	1.2	4.1	1.6	1.1
20° middle braking time (s)	1.8	1.6	1.7	1.7
20° left braking time (s)	2	1.9	1.9	1.8
20° right braking time (s)	1.4	1.6	1.5	1.5
20° cross left braking time (s)	2.1	2.2	2.1	1.9
20° cross right braking time (s)	1.5	1.6	1.6	1.5
20° split left braking time (s)	1.5	1.7	1.6	1.6
20° split right braking time (s)	1.8	4.7	1.9	1.7
48° middle braking time (s)	1.7	5	2	1.4
Theoretical braking time (s)	1.2	4.1	1.6	1.1
configuration 1 braking time (s)	1.8	1.6	1.7	1.6
configuration 2 braking time (s)	1.4	1.6	1.5	1.5
configuration 3 braking time (s)	1.5	1.6	1.6	1.5
configuration 4 braking time (s)	1.5	1.7	1.6	1.6
configuration 5 braking time (s)	1.7	1.6	1.7	1.6

**Table 21. Braking times for turn and stop scenario**

Scenario	Human-crossing					
	7	8	9	10	11	12
Simulations						
Theoretical braking time (s)	1.5	2.5	0.3	0.3	0.9	0.9
20° middle braking time (s)	1.5	2.5			0.9	0.9
20° left braking time (s)	1.5	2.5				
20° right braking time (s)	1.3	2.5	0	0	0.5	0.5
20° cross left braking time (s)	2.4	2.5				
20° cross right braking time (s)	0.9	1.7	0	0	0.5	0.5
20° split left braking time (s)	0.9	2.4	0	0	0.4	0.4
20° split right braking time (s)	1.9	2.6	2		2	2.2
48° middle braking time (s)	2.6	3.5	1.5	1.5	2.1	2.1
Theoretical braking time (s)	1.5	2.5	0.3	0.3	0.9	0.9
configuration 1 braking time (s)	1.5	2.5			0.9	0.9
configuration 2 braking time (s)	1.3	2.5	0	0	0.5	0.5
configuration 3 braking time (s)	0.9	1.7	0	0	0.5	0.5
configuration 4 braking time (s)	0.9	2.4	0	0	0.4	0.4
configuration 5 braking time (s)	1.5	2.5	1.5	1.5	0.9	0.9

**Table 22. Braking times for opposite-direction lane-change scenario.**

Scenario	Turn and stop					
	1	2	3	4	5	6
Simulations						
Theoretical braking time (s)	4.6	5.1	5.6	4.2	3.7	3.2
20° middle braking time (s)	4.4	4.8	5.3	4	3.7	3.4
20° left braking time (s)	4.4	4.9	5.4	4	3.7	3.6
20° right braking time (s)	4.4	4.8	5.3	4	3.6	3.3
20° cross left braking time (s)	4.4	4.8	5.3	4.2	4.1	4
20° cross right braking time (s)	4.8	5.2	5.6	4	3.7	3.4
20° split left braking time (s)	4.8	5.1	5.3	4.1	3.7	3.5
20° split right braking time (s)	4.3	4.9	5.3	4	3.9	3.8
48° middle braking time (s)	5.8	6.3	6.8	5.2	4.7	4
Theoretical braking time (s)	4.6	5.1	5.6	4.2	3.7	3.2
configuration 1 braking time (s)	4.4	4.8	5.3	4	3.7	3.4
configuration 2 braking time (s)	4.4	4.8	5.3	4	3.6	3.3
configuration 3 braking time (s)	4.4	4.8	5.3	4	3.7	3.4
configuration 4 braking time (s)	4.3	4.9	5.3	4	3.7	3.5
configuration 5 braking time (s)	4.4	4.8	5.3	4	3.7	3.4



## REFERENCES

1. *Global Salary Report*. The International Society for Optics and Photonics, 2015.
2. LeddarTech, *LeddarVu Spec Sheet*, LeddarTech, Editor.: [www.autonomoustuff.com](http://www.autonomoustuff.com).
3. Dhar, S.P., *From Google to Tesla, it's a war of LiDAR or RADAR*. [unitedlex.com](http://unitedlex.com), 2017.
4. Partners, W.C., *Beyond The Headlights: ADAS and Autonomous Sensing*. 2016.
5. Wilson, T., *Sensing Technology for the Autonomous Vehicle*, in *Embedded Vision Summit*. 2016.
6. Velodyne LiDAR, I., *VLP-16 Specification Sheet*.
7. SICK, *LMSxx Product Information*.
8. Silfvast, W.T., *Laser Fundamentals*. 2004: Cambridge University Press.
9. Carroll, J., *American National Standard for Safe Use of Lasers*. *Neuroscience Research*, 2007. **66**(1): p. 22-29.
10. Yenkanchi, S., *MULTI SENSOR DATA FUSION FOR AUTONOMOUS VEHICLES*, in *Department of Electrical & Computer Engineering*. 2016, University of Windsor: Electronic Theses and Dissertations. p. 68.
11. Kalman, R.E., *A New Approach to Linear Filtering and Prediction Problems*. 1960. **82D**: p. 35-45.
12. Obolensky, N., *Kalman filtering methods for moving vehicle tracking*. 2002, University of Florida.
13. Haykin, S., *Kalman Filtering and Neural Networks*. 2001: John Wiley & Sons, Inc. 170 - 174.
14. Snare, M.C., *DYNAMICS MODEL FOR PREDICTING MAXIMUM AND TYPICAL ACCELERATION RATES OF PASSENGER VEHICLES*. 2002.
15. Pak, J.M., et al., *Self-recovering extended Kalman filtering algorithm based on model-based diagnosis and resetting using an assisting FIR filter*. *Neurocomputing*, 2016. **173**: p. 645-658.
16. Mahalanobis, P.C., *On the generalised distance in statistics*. *Proceedings of the National Institute of Sciences*, 1936. **2**: p. 49-55.

17. Luo, Z., S. Habibi, and M.v. Mohrenschildt, *LiDAR Based Real Time Multiple Vehicle Detection and Tracking*. World Academy of Science, Engineering and Technology, International Journal of Computer, Electrical, Automation, Control and Information Engineering, 2016. **10**(6): p. 1125-1132.
18. Holý, B., *Registration of Lines in 2D LIDAR Scans via Functions of Angles*. IFAC-PapersOnLine, 2016. **49**(5): p. 109-114.
19. Brown, M., et al., *Safe driving envelopes for path tracking in autonomous vehicles*. Control Engineering Practice, 2017. **61**(Supplement C): p. 307-316.
20. Kou, C.C. and R.B. Machemehl, *Modeling vehicle acceleration deceleration behavior during merge maneu*. Canadian Journal of Civil Engineering, 1997. **24**(3): p. 350-358.
21. Moussa, G., E. Radwan, and K. Hussain, *Augmented Reality Vehicle system: Left-turn maneuver study*. Transportation Research Part C: Emerging Technologies, 2012. **21**(1): p. 1-16.
22. AKHILESH KUMAR MAURYA, P.S.B., *STUDY OF DECELERATION BEHAVIOUR OF DIFFERENT VEHICLE TYPES*. INTERNATIONAL JOURNAL FOR TRAFFIC AND TRANSPORTATION, 2012: p. 253-279.
23. V, J.H.L.I.A.J.H.L., *A HEAT TRANSFER TEXTBOOK*. 2011: DOVER PUBLICATIONS.
24. GOURI DHATT, G.T., *FINITE ELEMENT METHOD*. NUMERICAL METHODS SERIES. 2012: WILEY.
25. Camba, J.D., M. Contero, and P. Company, *Parametric CAD modeling: An analysis of strategies for design reusability*. Computer-Aided Design, 2016. **74**(Supplement C): p. 18-31.