Pedestrian Detection using Vehicle-to-Vehicle Communication

William ‘Scott’ Hanna, wshanna@wm.edu
Github: https://github.com/Scotthanna16/ECProject

I. Research Question and Significance

The goal of this research project is to lay the foundations for a system that could prevent vehicle-pedestrian collisions. The aforementioned system would achieve this be detecting a pedestrian and alerting other vehicle that may not be able to sense the pedestrian.

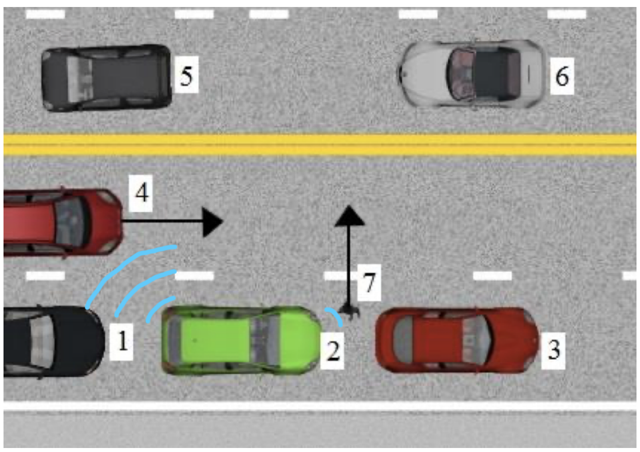

Fig. 1. System example

For example, let’s assume that vehicle number 4 can’t sense the pedestrian (7). If vehicle 4 is moving at a high speed, it may not have time to stop before hitting the pedestrian. However, if vehicle 2 senses the pedestrian and alerts vehicle 4 of the pedestrian, vehicle 4 may have time to stop before hitting the pedestrian.

This goal will be achieved by addressing this research questions: What is the fastest and most reliable way of sensing a pedestrian that might get hit by a vehicle?

II. Background

In 2022 over 7500 pedestrians in the United State were struck and killed by cars [1]. Furthermore, as seen in figure 2, pedestrian deaths are increasing at a rate 3 times higher than all other pedestrian traffic deaths. As cell phones become more and more popular, people will continue to take their eyes off the road more. This will cause inattentiveness of drivers to surge even more. Autonomous vehicles will help solve this problem,

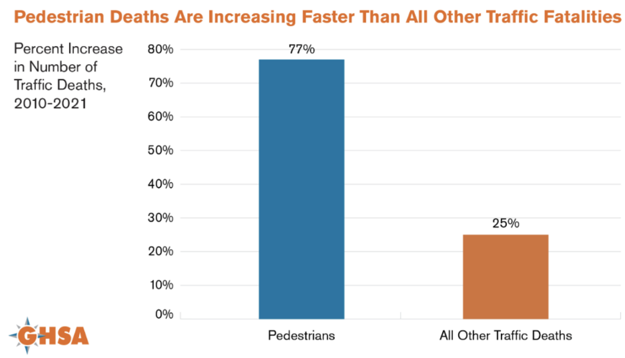

Fig. 2. Pedestrians Traffic Deaths rate

as these vehicles are constantly scanning, updating, and making decisions to ensure the safety of drivers and pedestrians, even when the drivers operating the vehicle are not focused. However, autonomous vehicles by themselves suffer from a problem that people also have, they can only make decisions based on what they can sense. As of January 15, 2023 there were 156 accidents involving fully autonomous vehicles in the United States [2]. Autonomous vehicles will continue to improve, but they will always have limitations. If there is an object blocking the view or sensing capabilities of an autonomous vehicle and a pedestrian walks out from behind that object the vehicle may not have time to stop.

The purpose of vehicle-to-vehicle communication is to create a network of vehicles that are constantly transmitting data in order to provide a safer and more efficient driving experience. This network would be a peer based system, in which the network would run through the vehicles, not a centralized system. This is crucial for speed and security. The basic goal of this network is for vehicles to transmit details about their speed, braking, turning, etc. which could prevent accidents while also making driving more efficient (e.g. preventing traffic jams) [3].

Pedestrian detection has been at the heart of autonomous vehicle research for years and it is obvious that pedestrian detection is a very important issue.

Currently, most pedestrian detection systems stem from two types of sensors [4].

1) Camera's can be used to detect pedestrians. In fact Tesla's pedestrian detection system uses cameras [5].
2) The other option to detect pedestrians is LiDAR(Light Detection and Ranging). Waymo and Ford are notable users of LiDAR detection.

Both options have their merits and will be further explored in the next section of this paper.

III. Existing Methods

A. Methods for Vehicle-to-Vehicle communication

Currently, vehicle-to-vehicle communication systems are made of devices which utilize dedicated short-range radio communication [6]. These systems operate in a similar way to Wifi. As the name suggests they are short range, being able to communicate data up to roughly 300 meters away. These systems are also capable of transmitting data at over 5 GHz [3].

B. Methods for Pedestrian Detection

As stated above, there are two popular methods for Pedestrian detection.

The first option is detection using cameras. Camera based pedestrian detection is already being used by many car brands [4]. These camera's take live pictures which are then analyzed by a deep convolutional neural network in order to obtain the necessary information. Camera's have multiple benefits compared to LiDAR. A camera system is considerably cheaper than a LiDAR system. Camera systems are also able to interpret weather conditions, which can be useful information. Finally, because these camera's 'see' they can gather other useful information. The camera system has three main flaws. The first is that a quick change in the amount of light (e.g. coming out of a tunnel) can disturb the camera's sensing capabilities. The second flaw, which is more important to the scope of this project, is that camera systems struggle to sense exact depth. Which when detecting if another car is going to hit someone could be very important. Finally, camera based systems struggle during the night time or when it is very cloudy outside. This is due to the lack of light available for the camera [4].

The second option is detection using LiDAR. The main benefit of LiDAR is that it is extremely accurate. LiDAR is extremely good at sensing pedestrians and identifying on small things. An example of this is LiDAR systems being able to pick up on cyclist gestures enabling the car to know when the cyclists will turn [4]. Another benefit is how accurately it can detect the distance and direction of a pedestrian. Finally, LiDAR works well anytime of day as it does not depend on computer vision like the camera based systems do. However, LiDAR cannot be used effectively in bad weather conditions, as these conditions can interfere with the light reflection. This problem is amplified as accidents are already more likely during bad weather conditions [4].

Something that should be noted is that these two systems actually complement each other quite well. LiDAR will work anytime, day or night, which is a weakness for the camera based system. In contrast, camera based systems work well in many different types of weather, which LiDAR can struggle in.

The problem this research project aims to address is a very unique situation, and both options have their merits and issues regarding this problem.

IV. Research Gaps

There are a number of problems that need to be addressed with current vehicle-to-vehicle communication systems. Transmission time and sensor error are among the most important of these issues. In "Pedestrian Protection Using the Integration of V2V and the Pedestrian Automatic Emergency Braking System" [7], the authors discuss two sensor types, camera and radar. LiDAR is another system that should be tested, as the direction and distance information provided by LiDAR could be very useful. The authors also discuss what a potential warning message would consist of. Their proposed message contains lots of information about the host vehicle, pedestrians, and other vehicles. Transmitting all of that information may not be necessary. If the host vehicle could confidentially compute if another vehicle was going to hit a pedestrian, the host could transmit a simple emergency stop message. A future project may answer the question: What data should be communicated to another vehicle that is about to hit a pedestrian?

My experiment will aim to build on the work in "Pedestrian Protection Using the Integration of V2V and the Pedestrian Automatic Emergency Braking System" [7], by testing LiDAR and comparing its accuracy to the camera based system.

V. Experiment objective

The objective of this experiment is to determine whether LiDAR or camera-based sensing is better for a theoretical V2V-pedestrian detection system. Both the LiDAR and camera-based sensors will be tested in a variety of different environments to determine their accuracy with respect to depth in day to day usage. This means that variables like weather and daylight will be varied during the experiment. Edge cases, like high glare scenarios, will not be the focus of this experiment.

VI. Method Description

The software I will be using in the experiment is CARLA. CARLA is an autonomous vehicle simulator that comes with built in tools which enable the testing of different types of sensors in different environments. CARLA also allows you to place pedestrians in testing scenarios. CARLA supports LiDAR and camera sensor simulation, and allows the user to access the raw data captured by those sensors. Each of these factors will be extremely useful for this project. CARLA also allows the user to access the ground truth, in other words the user can test the accuracy of the sensors.

VII. Hardware Description

CARLA has the following hardware requirement. On a windows system CARLA requires an x64 system, 165 GB Disk space, a 6 GB GPU (8 preferred), two TCP ports and internet [8]. The gaming laptop I used was able to meet all of these requirements with an 8 GB GPU.

VIII. Data Sets

The data sets for this project are going to be the different scenarios set up in CARLA. Setting up multiple different scenarios and then testing both the LiDAR and camera sensors, while varying the weather for each scenario should provide ample data about the accuracy of the sensors.

IX. Original Experiment design

In CARLA I will test two different types of sensors, LiDAR based and camera-based. Each sensor will be sent through the same scenarios so that their results are comparable. The result of each test is the difference between the distance given by the sensors. The variables in this experiment are the range of scenarios for each test. I will start with 5 basic scenarios. These basic scenarios will vary in two ways, the distance of the objects the car is sensing and the angle of the objects in relation to the car. The five basic scenarios are:

1) Sensing another car and pedestrian directly ahead 10 meters away.
2) Sensing another car and pedestrian directly ahead 30 meters away.
3) Sensing another car and pedestrian directly ahead 40 meters away.
4) Sensing another car and pedestrian perpendicular to the vehicle 10 meters away.
5) Sensing another car and pedestrian perpendicular to the vehicle 20 meters away.

I have picked these basic scenarios because I believe they are the most realistic. All of these base scenarios involve sensing both another car and pedestrian, and assume there is nothing blocking the sensing vehicle. They also assume there is something blocking the other car from seeing the pedestrian. I have made these assumptions because that is when a V2V-pedestrian detection system would be most useful.

Each of the five basic scenarios above will be varied in multiple ways. Camera-based sensing is known to struggle at night when there is little light available, while LiDAR based sensing struggles in certain weather conditions [4]. Each of the five base scenarios will go through the following variations:

1) Clear skies during the day (noon)
2) Cloudy skies during the day (noon)
3) Clear skies during the night (midnight)
4) Cloudy skies and light fog during the day (noon)
5) Cloudy skies and heavy fog during the day (noon)
6) Cloudy skies and light fog during the night (midnight)
7) Cloudy skies and heavy fog during the night (midnight)

I have chosen these weather conditions because they are very common, and should highlight where each sensor is weak. Edge cases, like snow or high wind scenarios, are important, and could be the subject of a future project.

In total each basic scenario has seven variations, and each of those variations will be tested by both sensors. Every variation of each scenario will be tested five times with both types of sensors. Testing each variation five times will ensure that the effect of outliers is minimal.

X. Experimental Procedure

The steps for each test are as follows:

1) Set up the scenario as I have described
2) Retrieve the sensor data when the true distance is at the required mark
3) compute the sensing depth error for both the pedestrian and other vehicle

The error will be the difference between the true distance of the objects, and how far away the sensor computes them to be.

XI. Sample Selection

At the end of the experiment there will be 350 data points collected. As I stated above I am running each variation five times to ensure no one data point is overvalued. I do not believe deleting outliers is the correct choice because some of the variations are designed to put the sensors in challenging situations where the sensor depth is off by a large margin.

XII. Experiment Changes

Unfortunately some technology issues forced me to change my experiment. The full version of CARLA requires the use of an Unreal Engine fork. I was unable to clone the version of Unreal Engine that CARLA requires due to persistent file corruption issues. I was able to download a smaller, less powerful, version of the simulator, but this severely limited what I was able to test. The major change in my experiment is that this smaller version of CARLA does not support LiDAR.

The camera testing was also handicapped by this version of CARLA. In the version of CARLA I used, I could not add rain to the simulation. So I changed from rain to fog in my experimental set up. The larger problem I encountered is with the results, which is discussed in the next two sections.

There were two more changes made to the original experiment. The first is instead of testing at distances of 10, 30, and 50 I decided to test at distances of 10, 20, and 30. I made this change because I thought those distances were more representative of how a system like this would be used in real life. Finally, I realized that changing the direction of the camera in CARLA was not going to affect the sensor data. This means that whether the pedestrian was directly ahead of the car or perpendicular to it, the accuracy was not going to affected. Due to this fact, I decided not to test when the pedestrian was perpendicular to the vehicle, as it would just be a waste of time.

XIII. Results

Below is the data for the experimental results. I placed objects at a distance of 10, 20, and 30 meters, the data is those respective columns is the sensed depth.

Camera Test Data

Scenario	10m	20m	30m
Clear Noon	7.84	19.61	31.37
Clear Midnight	7.84	19.61	31.37
Cloudy Noon	7.84	19.61	31.37
Cloudy Midnight High Fog	7.84	19.61	31.37
Cloudy Midnight Light Fog	7.84	19.61	31.37
Cloudy Noon High Fog	7.84	19.61	31.37
Cloudy Noon Light Fog	7.84	19.61	31.37

XIV. Analysis of Results

There are two issues with the results. The first is that they are the same for every scenario. The second is that they are not accurate. These two problems stem from the same issue. The simulator expresses depth using the RGB values in the various pixels, and those values are integers. The RGB values must also be the equal to one another so that the results is a neutral color. The formula to convert the RGB values to distance, in meters, is [9]:

$$Distance = \frac{1000(R + 256G + 256^2B)}{256^3 - 1}$$

As stated above, RGB values must be integers so this formula will only give certain results, even if the true sensed distance is slightly different.

Lets work through an example to demonstrate this problem. First assume we have an object that is 8 meters away. Plugging 8 into our equation we get:

$$8 = \frac{1000(R + 256G + 256^2B)}{256^3 - 1}$$

$$8 * (256^3 - 1) = 1000(R + 256G + 256^2B)$$

Note that $R = G = B$ because the depth camera's pixels are all neutral colors. If we let $R, G, B = C$ then we get:

$$8 * (256^3 - 1) = 1000(C + 256C + 256^2C)$$

$$8 * (256^3 - 1) = 65,793,000 * C$$

$$C = \frac{134,217,720}{65,793,000} = 2.04$$

As stated above C must be an integer so 2.04 rounds to 2.

Fig. 3. Simulator Clear Noon

Fig. 4. Simulator Cloudy Noon High Fog

Now lets compare compare this to an object that is 8.5 meters away.

$$8.5 = \frac{1000(C + 256C + 256^2 C)}{256^3 - 1}$$

$$8.5 * (256^3 - 1) = 1000(C + 256C + 256^2 C)$$

$$8.5 * (256^3 - 1) = 65,793,000 * C$$

$$C = \frac{142,606,327.5}{65,793,000} = 2.168$$

Again, C must be an integer so 2.17 also rounds to 2. Plugging 2 into the original equation we get:

$$7.84 = \frac{1000(2 + 256 * 2 + 256^2 * 2)}{256^3 - 1}$$

To further demonstrate and clarify the problem, I've included four figures. Figures 1 and 2 are from the simulator, while figures 3 and 4 are from the depth camera. Figure 3 corresponds to figure 1 and figure 4 corresponds to figure 2.

Looking at figures 3 and 4, it is obvious that the simulator has loaded the requested weather patterns. Ideally, the sensed depth in figure 3 would be close to perfect as the weather is clear and there is plenty of light available. The sensed depth in figure 4 is expected to be less accurate as the fog should interfere with the camera. Now let's look at the depth camera images.

Fig. 5. Clear Noon Depth Camera

Fig. 6. Cloudy Noon High Fog Depth Camera

The depth camera images are the same picture (figures 3 and 4), even though the environments are different. Hence, the same distance is calculated.

XV. Contributions

Unfortunately, there are no real contributions from this project. The data is simply not useful because it is not precise enough. If I had been able to test LiDAR as I originally intended, then a meaningful contribution could have been made. However, I have multiple ideas for future research projects.

XVI. Comparisons and Related Work

The base for a V2V-pedestrian detection system was cre- ated in "Pedestrian Protection Using the Integration of V2V and the Pedestrian Automatic Emergency Braking System" [7]. As stated above In the paper the authors discuss two sensor types, camera and radar, and run some basic simulations. Their testing is somewhat limited but their results are promising. If the results from my tests are accurate, which I do not believe, then the camera system is not going to work. The calculated depth would not be precise enough to correctly inform another vehicle it was going to hit a pedestrian.

As stated above I was not able to test LiDAR, but I believe to build a fully reliable system both LiDAR and cameras need to be used. LiDAR systems work well at night and are extremely accurate, but they struggle in adverse weather conditions. Camera systems work well in weather, but struggle in the night, and aren't good at calculating depth [4]. These two systems complement each other very well, and together could handle a wide variety of scenarios.

XVII. Future Research

Although I was not able to successfully complete my original experiment idea, it still should be tested. As stated above, depth sensing is one of LiDAR's strengths, which is very important for this system and worth researching.

LiDAR is not perfect and can struggle in conditions like rain and fog [4]. These are areas where camera systems work very well. Testing a system that uses both LiDAR and camera based sensing is another promising area in this field.

Another research project could be training a Convolutional Neural Network to recognize when a collision is or isn't going to take place given the location of the pedestrian and the other vehicle. An accurate model could allow for the transmission of the emergency stop message described in "Pedestrian protection using the integration of V2V and the pedestrian automatic emergency braking system" [7].

XVIII. Notes about Code

The code for this project is written in Python. Python version 3.7 is required, any other version of Python will not work. These scripts will only run on the CARLA nightly build. I followed three tutorials to get everything set up.

1) https://www.youtube.com/watch?v=jIK9sanumuU
2) https://www.youtube.com/watch?v=zZ8s_qrKYGE&t=289s
3) https://www.youtube.com/watch?v=om8klsBj4rc

The first two videos are just setting up the simulator. The third video is about setting up the sensors. Note, these tutorials are for the CARLA nightly build which was not the version of the software I originally tried to download.

References

[1] "Pedestrian traffic fatalities by State: 2022 preliminary data," GHSA, https://www.ghsa.org/resources/Pedestrians23#: :text =The%20report%20projects%20that%20drivers,of%2020%20 deaths%20every%20day.

[2] M. Salter, "How Many People Have Actually Been Killed By Self-Driving Cars?," SlashGear, Feb. 16, 2023.https://www.slashgear.com/1202594/ how-many-people-have-actually-been-killed-by-self-driving cars/#: :text=Ac cording%20to%20the%20NHTSA%2C%20carmakers

[3] "Understanding Vehicle-to-Vehicle (V2V) Communication | The Automotive Trend Guid | Avnet Abacus." https://my.avnet.com/abacus/solutio ns/markets/automotive-and-transportation/automotive/communications-and-connectivity/v2v-communication/#

[4] S. Iftikhar, Z. Zhang, M. Asim, A. Muthanna, A. Koucheryavy, and A. a. A. El-Latif, "Deep Learning-Based Pedestrian Detection in Autonomous Vehicles: Substantial Issues and challenges," Electronics, vol. 11, no. 21, p. 3551, Oct. 2022, doi: 10.3390/electronics11213551.

[5] "Safety," Tesla,https://www.tesla.com/en_my/safety#: :text=Eig ht%20cameras%20and%20powerful%20vision,like%20pedestr ians%2C%20bicyclists%20and%20vehicles.

[6] "NHTSA ISSUES ADVANCE NOTICE OF PROPOSED RULEMAKING AND RESEARCH REPORT ON GROUND-BREAKING CRASH AVOIDANCE TECHNOLOGY: 'VEHICLE-TO-VEHICLE COMMUNICATIONS: READINESS OF V2V TECHNOLOGY FOR APPLICATION.'" Available: https://www.nhtsa.gov/sites/nhtsa.gov/files/documents/v2v_fa ct_sheet_101414_v2a.pdf

[7] B. Tang, S. Chien, Z. Huang, and Y. Chen, "Pedestrian protection using the integration of V2V and the pedestrian automatic emergency braking system," 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Jan. 2016. doi:10.1109/itsc.2016.7795913

[8] "Windows build - CARLA Simulator." https://carla.readthedocs.io/en/latest/build_windows/

[9] "Sensors reference - CARLA Simulator," carla.readthedocs.io. https://carla.readthedocs.io/en/latest/ref_sensors/depth-camera (accessed Apr. 19, 2024).