Environmental Effects on Lidar Points for Robotics

Karissa Stisser
Final Paper
Source code: https://github.com/kstisser/LidarAnalysis
Intro to Mathematical Methods and Modeling
Masters of Science in Applied Mathematics
Northeastern University

***Abstract*—Lack of lidar points on an object can adversely affect algorithms in robotics when trying to extract information from a point cloud. In order to predict when there would be a reduction in points experiments were performed with an Intel L515 lidar to test the environmental effects of object size, object color, object distance, object temperature, room light, and room temperature and humidity (estimating fog) and their effect on number of points returned from an object. A model was constructed to support dynamic algorithmic or sensor changes based on environmental factors.**

***Index Terms*—lidar, point cloud, modeling, environmental factors, algorithms, robotics**

I. Introduction

In robotics light detection and ranging (lidar) is often used by sending electromagnetic light signals and measuring the time difference of the return signals to detect distances of objects throughout the environment [3]. Algorithms process these returns to ascertain information on the scene, such as plane detection or dynamic obstacle detection. However, a lack of returns on an object can result in inaccurate calculations due to algorithm failure. Environmental factors most likely cause light points to fail to reflect as expected. Therefore, experiments were conducted to find relationships in environmental effects on lidar points, and a model was constructed for responding dynamically to anticipated point return failures.

II. Experiment

To conduct the environmental experiments the factors that were varied were object size, object color, object temperature, object distance from the lidar, and room temperature and humidity. Two object types were used. One was a set of three white PLA 3D printed cubes of the varied sizes 40mm, 80mm, and 160mm in width. The other was a set of four (silver, black, blue, and rose gold) hand warmers of the same size with three temperature presets of 40°C, 45°C, and 50°C. Ambient temperature was also considered for all variations. ROS bag recordings were made of the light, temperature and humidity in the room as well as the lidar points detected on the object. Recordings were made with the variations of both cubes and hand warmers at each distance 2FT, 4FT, and 6FT from the lidar. Size was varied for the cubes at each distance, while color and temperature were varied for each distance for the hand warmers. These experiments were conducted in a bathroom, starting with ambient conditions, and then leaving a hot shower running to increase temperature and humidity, attempting to replicate conditions of fog. Experiments were done during the night, so the only source of light was an compact fluorescent light bulb, which was turned off and on every 30 seconds throughout each experiment. After all experiments were conducted, a few scenarios were not able to be used due to corrupted bag files.

A. Hardware Setup

- **Intel L515**- This was used for collecting lidar depth data
- **Laptop**- A laptop was used to handle recording lidar data in a ROS bagfile through realsense-viewer. It was also used for post processing and analysis.
- **Sense Hat**- This was used to detect temperature and humidity
- **Circuit Playground Express**- This was used for light readings of the room
- **Raspberry pi**- Both the Sense Hat and the Circuit Playground Express were plugged into the Raspberry pi. The pi handled reading the sensor data, publishing out sensor readings on a ROS topic, and recording them in a ROS bag file

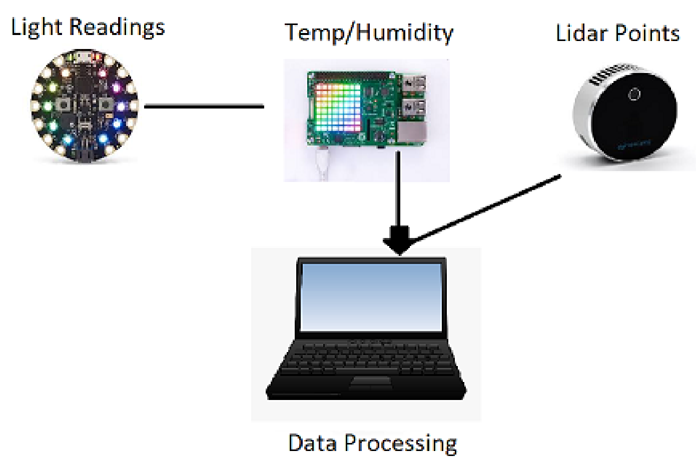

Fig. 1. Hardware data flow

B. Post Processing

The objective of post processing the data was to obtain all data from the experiment in one python DataFrame so it could be further analyzed. To do this respective bag files from the lidar and from the light/temperature/humidity system were first moved into the same file system. Then, C++ code was generated to utilize the librealsense serialization mechanism [2] to read out the depth data from the lidar recording, and this was output to a JSON file along with other relevant experiment information, such as object width, height, temperature, color, distance, and size. Because the lidar ran at 30 fps and light/temperature/humidity ran at 10 fps, every third reading was taken from the lidar. Then python was used to read in temperature, humidity, and light, and that was related to the previous JSON file of lidar points and experiment parameters.

C. Analysis

As the number of data points did not line up perfectly a few different variations of the data were taken to analyze. First, zeros were input when there were not enough sensor measurements to line up to the other sensor returns. Next,returns were cut at the length of the shortest sensor return feed. Finally, the results of the second method mentioned above were divided by the expected number of lidar returns at a single frame, which was computed using the equation for P, and compared to the actual number of lidar points detected.

Three different level of plots were generated, at the individual experiment level, at the Cube or Handwarmers accumulated data level, and at the entire data set combined level.

The different types of plots that were generated included a seaborn pairwise plot [1] of all data to see relationships, a matplotlib 2D plot comparing each factor we're evaluating to the number of lidar points or percentage of expected lidar points, as well as a few 3D plots to further investigate relations found.

As a preface to explaining findings, there was not a large amount of data, so further solidifying findings would require more experiments and more data, which was not within the scope of this class project.

1) Object Distance: It was found that fewer points than expected are captured at a closer distance, as shown with a linear relationship in Figure 2.

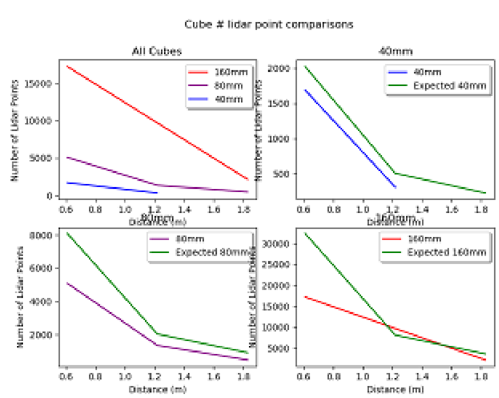

Fig. 2. Distance Size Comparison Plot

2) Object Size: When plotting the results in Figure 3 compared to expected results for size it was found that a larger size resulted in fewer than expected lidar points. The small amount of data leaves the type of relationship ambiguous, but it could be argued to be either linear or exponential, and the variations of equations below will explore this idea.

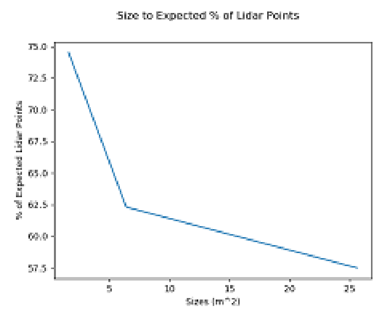

Fig. 3. Size Relationship

3) Object Temperature: Object temperature did not seem to have any noticeable affect, shown on the z axis, as there is little deviation on number of lidar points with increased temperature shown in Figure 4.

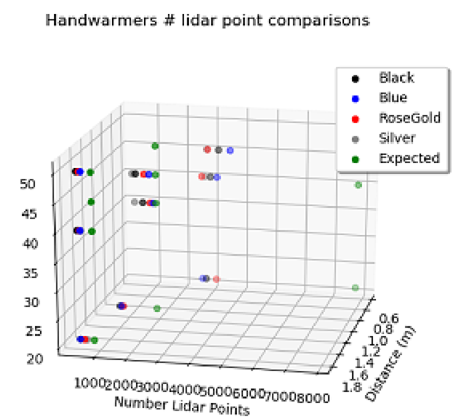

Fig. 4. Object Temperature and Color Relationship

4) Object Color: Object color did not seem to have any noticeable affect, as there was no clear pattern in the order, shown in Figure 4.

5) Room Light: Room light did not seem to have any noticeable affect, as the fitted slope is near 0 in Figure 5. One potential reason for this is the light that was measured was not in the spectrum of light that would impact the IR light being sent from the Intel Lidar at 860nm.

6) Room Temperature: Room Temperature was shown to have a linear trend with lower lidar points the greater the temperature rose, shown in Figure 6.

7) Room Humidity: Room Humidity was shown to have a linear trend with lower lidar points the greater the humidity rose, shown in Figure 7.

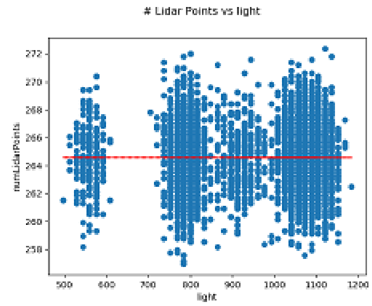

Fig. 5. Light Relationship

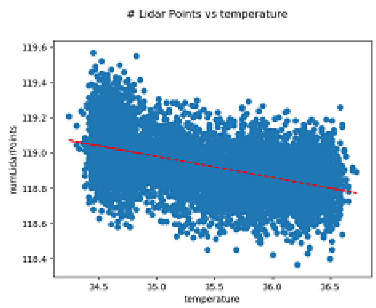

Fig. 6. Temperature Relationship

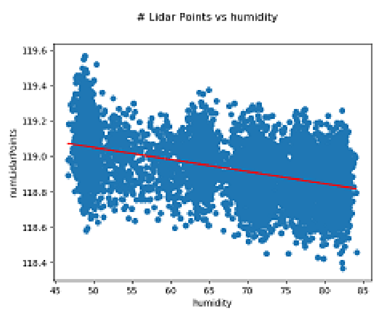

Fig. 7. Humidity Relationship

III. Model

A. Intel Lidar Specs

Note- these are the specs that were used for this experiment, but is not the only specs this lidar is capable of.

- Field of View: $70°x55°$
- Min range: 0.25 meters
- Max range: 9 meters
- Depth resolution: 640x480
- Frame rate: 30 FPS
- Wavelength: 860nm

Note- Figure 8 represents both a top down view of the lidar's perspective as well as a 90 degree rotated side perspective. The top down view (TDV) variables will denote a subscript W for the object's width, and the side view (SV) will denote a subscript H for the object's height. Figure 9 shows the view of a cube's depth data from the lidar's perspective.

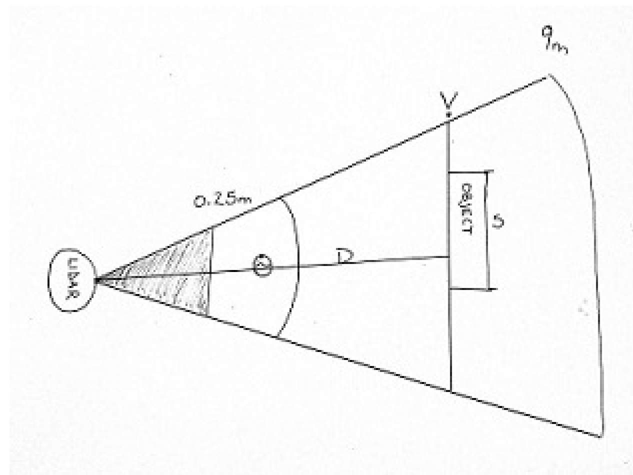

Fig. 8. Lidar Perspective Diagram

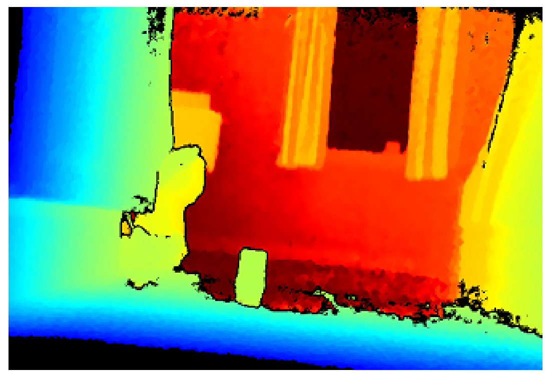

Fig. 9. Depth data

B. Variables

Table I shows the variables used in the equation models.

C. Equations

Due to the Specs of the Intel Lidar, the equations are considered within the lidar's range of 0.25m to 9m.

1) Base Equation:

$$V_W = 2 * tan(\theta_W/2) * D \tag{1}$$

$$V_H = 2 * tan(\theta_H/2) * D \tag{2}$$

$$X_W = V_W/R_c \tag{3}$$

$$X_H = V_H/R_r \tag{4}$$

$$N_W = S_W/X_W \tag{5}$$

$$N_H = S_H/X_H \tag{6}$$

$$P = N_W * N_H \tag{7}$$

TABLE I
VARIABLES- (GREEN ARE FIXED BY LIDAR SPECS)

Variables	
D	Distance from the lidar to the object
S_W	Object's Width
S_H	Object's Height
S	Object's Surface Area
V_W	Viewing range at the distance of the object, 90°to D (TDV).
V_H	Viewing range at the distance of the object, 90°to D (SV).
θ_W	Lidar's Field of View (TDV)
θ_H	Lidar's Field of View (SV)
R_c	Resolution of depth image (columns or TDV)
R_r	Resolution of depth image (rows or SV)
X_W	Meter per pixel ratio at depth D (TDV)
X_H	Meter per pixel ratio at depth D (SV)
N_W	Number of depth pixels expected for object's width
N_H	Number of depth pixels expected for object's height
P	Total depth pixels expected for object's surface area at depth D
P_D	P with diminished returns from environmental factors
T	Instantaneous temperature
H	Instantaneous humidity
a	Tuning variable
b	Tuning variable
c	Tuning variable

2) Base Equation with Intel Lidar Specs:

$$V_W = 2 * tan(70°/2) * D \tag{8}$$

$$V_H = 2 * tan(55°/2) * D \tag{9}$$

$$X_W = 1.4D/640 \tag{10}$$

$$X_H = 0.51D/480 \tag{11}$$

$$N_W = 500(S_W/D) \tag{12}$$

$$N_H = 941.2(S_H/D) \tag{13}$$

$$P = 470600(S/D^2) \tag{14}$$

3) Potential Equations with Influencing Factors:

1) Linear size relationship with combined relations

$$P_D = P - PcSTH/(aD) \tag{15}$$

2) Exponential size relationship with combined relations

$$P_D = P - e^{bS}PcTH/(aD) \tag{16}$$

3) Linear size relationship with separate relations

$$P_D = P - bSP - cTHP - P/(aD) \tag{17}$$

4) Exponential size relationship with separate relations

$$P_D = P - e^{bS}P - cTHP - P/(aD) \tag{18}$$

IV. FINDING THE BEST EQUATION

Equation 17 was found to be the best fit for the equation. To find this the curve fit function in the python optimization library was used to fit each equation to find the optimal parameters. The data was carefully separated so most data was used to fit, but the RoseGold handwarmer data was kept separate for evaluating results. This data chunk was chosen because several of the tests were the only source of certain data, such as combinations of a size and a distance, but because the color was not a factor, RoseGold largely represented the same data other handwarmers had collected. This model resulted in 5.9% difference compared to the actual number of points that were measured.

The optimal parameters found for the equation were:

$$a = 4.08$$
$$b = 41.8$$
$$c = 0.0003$$

So, the end equation becomes:

$$P_D = 470600S/D^2 * (1 - 41.8S - 0.003TH - 1/(4.08D)) \tag{19}$$

V. SIMULATIONS

As this sensor is meant to be used indoors to avoid conflict with sunlight, I will come up with a few real world application scenarios to predict how well this sensor will perform.

A. Scenario 1

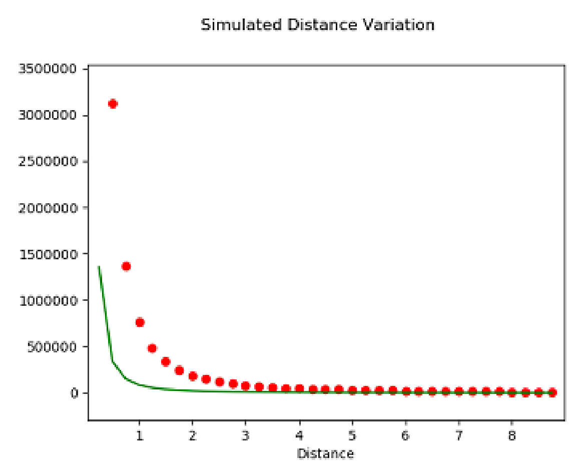

Fig. 10. Distance Simulation

This scenario will imagine this sensor is being used for an indoor vacuum cleaner. So, the distance to objects will vary over time. We want to see how well the vacuum would detect points to a toddler. We'll assume the average toddler is about 0.9 m tall and 0.2 m wide, making our Area = 0.18m. We'll assume a comfortable home remains around Temperature = 21°C, and Humidity = 35 with little change. We'll simulate the distance varying for the range of the sensor from 0.25m to 9m.

In Figure 10 the red depicts our simulation output, and the green depicts the expected value at that point. Just as the experiments showed we performed less accurately closer up and closer to the expected value further away. Overall, there was about an 8% reduction in points at an average of 62,607 points with a minimum of 9,688 points. At worst case scenario you would still have 8,912 points which should not be enough to negatively impact an algorithm looking to detect dynamic obstacles.

B. Scenario 2

We will create another scenario where this lidar is used as a delivery robot in a restaurant, automatically retrieving meats from the freezer as well as helping delivering food to the customers. This will anticipate a temperature change from about 0°C to 30°C with a constant humidity of 35, as well as looking at the depth variation of the sensor range from 0.25m to 9m. I will assume detection of the average human would give a size of about 1.6m by 0.3m, which gives a surface area of 0.48.

The results indicate there was a larger average error of about 16.97%. At the worst case on the minimum end there would be 2605 points, which would leave 2162 points, which is still enough points for detecting dynamic obstacles. However, on the upper end, as you can see in Figure 11 at the very close range the number of lidar points inflate, and increased temperature only worsens the situation, which might make accurately detecting people a challenge under 1 meter. It would be recommended to add an alternative sensor for close range detection based on these results, such as ultrasonics. Also, the operating temperatures of this sensor are 0-30°C, so further testing would be encouraged to ensure the lower end of the spectrum did not have unintended results that deviated from the model.

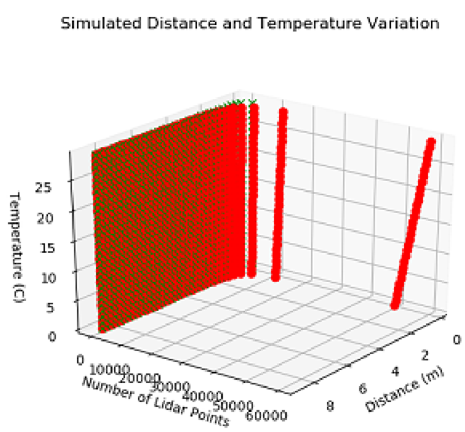

Fig. 11. Temp Distance Simulation

VI. Further Research

One interesting and unexpected result was that there was a difference in results between the handwarmers and the cubes. This was found when comparing color factors with red component in Figure 12 with the blue component in 13 with the green component in 14.

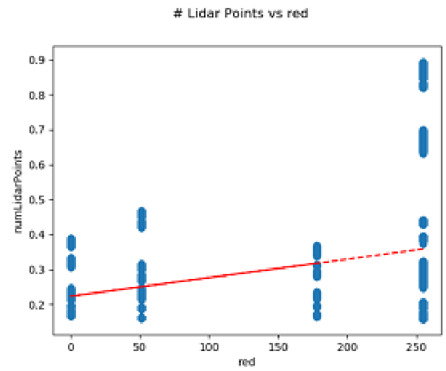

Fig. 12. Red Component

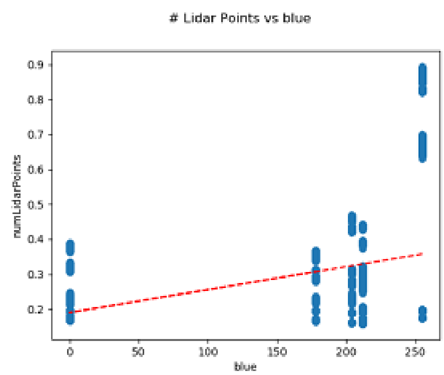

Fig. 13. Blue Component

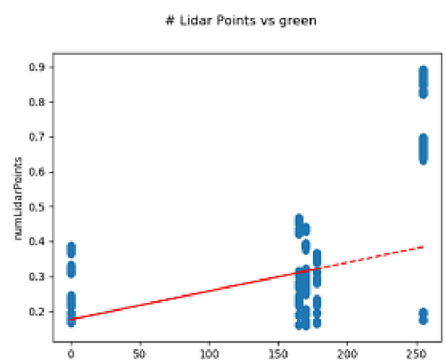

Fig. 14. Green Component

White cubes have a color component of (255,255,255), and was the only test object to have this color. The RoseGold had a 255 red component while all other color components were less than 255. As all values lower than 255 in all three plots had far fewer than the expected number of points than the 255 column, other than the red, which is evenly distributed, this shows that the handwarmers have a smaller percentage of expected results than the cubes. Potential reasons for this could be either material makeup which reflect IR points differently, or the contour shape of the handwarmers, in contrast to the flat shape of the cube, which could result in points being reflected or refracted in different directions from the lidar. To further investigate this, I reconstructed the original depth images that were taken from the handwarmers to see if there was any trend of increased detections in the center, which was flatter, vs the outer edges, which had more contour. Figure 15 indicates that there was no trend to be found. This would lead me to think that material makeup of the object being detection could be a fifth variable that could influence results, though further testing could be done by placing different surfaces on the front of a cube to be detected. This is a harder component to measure in robotics though. One method could be using machine learning to detect known things we wish to use this equation for, like ground surfaces, cars, or people, and add in an emissivity value for these known items based on a lookup table.

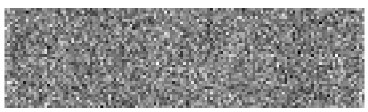

Fig. 15. HandwarmerDepthImage

VII. Conclusion

In conclusion, the factors that affected the number of lidar returns for the Intel Lidar are distance, size, temperature, and humidity. The linear and separated equation fit best. The largest challenge for accurate detection is at close range, which can be mitigated by adding alternative sensors for close range detection like ultrasonics. Light was not evaluated in the correct spectrum to be confident that it has no impact, and further testing could potentially allow near 860nm wavelength light be a factor. Also, these tests and experiments did not test the full capable spectrum of the sensor, so further testing would help solidify or improve the model found to ensure that extremes in temperature, humidity, distance and size did not produce an unexpected result. While the Intel Lidar was chosen for these experiments due to cost, the model will not necessarily directly translate to other Industry lidars easily because the Intel lidar looks strictly forward while the Velodyne lidars continually sweep in a 360 fashion to retrieve points, though the methods for testing, post processing, and evaluating could all be translated to evaluate another lidar.

References

[1] Michael Waskom, "Seaborn.pairplot," https://seaborn.pydata.org/generated/seaborn.pairplot.html 2012

[2] Gsaponaro, librealsense repo, https://github.com/IntelRealSense/librealsense 2020

[3] LOAM: Lidar Odometry and Mapping in Real-time. J Zhang, S Singh - Robotics: Science and Systems, 2014