

Simulate Self-Driving Taxi in Route Tracking

Mason Chen

Milpitas Christian School, Morrill Learning Center
San Jose, CA 95132

mason.chen.training@gmail.com

Abstract

Self-driving car performance is of great research interests these days. We would like to use a LEGO EV3 robot to simulate a Google self-driving taxi that is on the way to pick up five chief speakers to the conference center on time. Our major focus is on how to improve the route tracking accuracy and the cycle speed. Team has designed a very challenging field to test how the Robot could overcome the sharp turns at faster speed. Team measured the optics contrast to determine the measurement signal-noise ratio as the fundamental quality statistics. Team could optimize the Robot performance by measuring the optical contrast. Based on SPSS Histogram analysis, team has distinguished several Robot movement patterns associated with each distribution peak/mode. Team was able to conduct further root cause analysis accordingly to further improve Robot Architecture Mechanics, EV3 Software Programming, and control the environmental light variations. Each Histogram distribution mode has indicated a unique robot movement failure mode. Team can apply statistics on a very complicated Robot Mechanics, Physics, Optics, and Software Programming like an experienced Robotics Engineer through Team Building. Team has optimized these critical parameters in order to make this Google Taxi service safer and faster.

Keywords

Self-Driving, Robotics, EV3, Statistics, Google

1. Project Introduction and Define Phase

Self-driving car performance is of great research interests these days ^[1,2]. In this project, team would like to use a LEGO robot with EV3 program ^[3] to simulate a self-driving taxi that is on the way to pick up five attendees to the conference center. Project major focus is on how to improve the route tracking accuracy and the cycle speed.

1.1 Problem Statement

Assume a self-driving taxi is going to pick up five attendees at five different locations. What is the best route and how long does it take the taxi to transport all five people to the destination? Most important, how is the self-driving car's driving performance on the road? Can team use a Lego robot to simulate the motion of the car and provide reasonable assessment?

1.2 Observations and Literature Research

A research approach ^[4] enables real-time probabilistic object tracking. Now, computational time is allocated dynamically according to the shape of the track's posterior distribution. Another research ^[5] has demonstrated the tracking-based semi-supervised learning, which was an offline algorithm. This is fine in some contexts, but ideally a user could provide new hand-labeled training examples online, as the system runs, without retraining from scratch. Through discussion, team has identified the analogous aspects between a self-driving taxi and the LEGO robot as shown in Table 1.

Table 1 Research Comparison

Self-Driving Taxi	EV3 LEGO Robot
Travelling distance	Track size
Number of passengers on board	Weight of Robot
Sequence of pick-up	Track pattern
Road condition	Reflected intensity read by the color sensor

The following would be the benefits of the self-driving taxi:

- It will be cheaper because it would cost more money to hire a driver. With a Google self-driving car, you don't need a real driver.
- There will be fewer cars in parking lots and roads. Therefore, there will be less traffic.
- It is more efficient because the self-driving car will take the fastest and the most efficient route.
- It would be safer because drivers might get into car accidents.

Team has also researched the following information regarding the self-driving: Google's goal is to increase car utilization from 5-10% to 75% or more by facilitating sharing. That means fewer cars on the road. Who needs to own a car when you can just order a shared one and it'll drive up minutes later, ready to take you wherever you want?

"This has the potential to dramatically reduce the number of cars on the street, 80% of which have people driving alone in them, and also a household's cost of transportation, which is 18% of their income – around \$9,000 a year – for an asset that they use only 5% of the time," said Robin Chase, the founder and CEO of Buzzcar [6].

In 2030, self-driving cars are expected to create \$87 billion worth of opportunities for automakers and technology developers. With fewer cars around, parking lots and spaces that cover roughly a one-third of the land area of many U.S. cities can be repurposed. That could mean temporary downward pressure on real estate values as supply increases. It could also mean greener urban areas, as well as revitalized suburbs, as longer commutes become more palatable.

One study [7] also found that a fleet of 9,000 driverless taxis could serve all of Manhattan at about 40 cents per mile (compared to about \$4-6 per mile now). There are licenses for over 13,000 taxis in the Big Apple now. Wow, this Google Taxi project could study or simulate such an impact on our life in the future. Team was really more motivated and excited to conduct this STEM project after read these above information.

1.3 Project Scope

Team has identified several track patterns to simulate the travelling routes of the taxi depending on the sequence of pick-up in Figure 1.

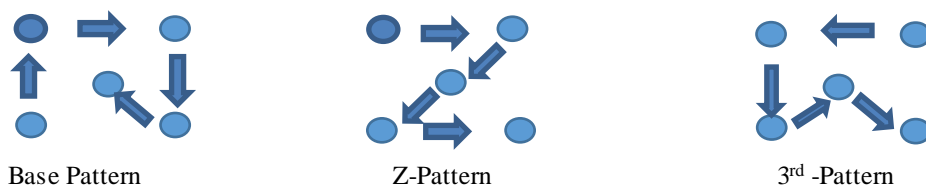


Figure 1. Pick up Patterns

Team has conducted a brainstorming on picking which track pattern. Team has considered two factors: how many sharp turns and how many long straight lines?

- 45 degree turn is much harder than 90 degree turn.
- Long straight lines are preferred for Robot to speed up after the sharp turn.

Base Pattern has one 45 degree turn and three long Straight lines. Z Pattern has two 45 degree turns and three long Straight lines. 3rd-Pattern has one 45 degree turn and two long Straight lines.

Based on the above information, team has decided to choose the Base Pattern as the Track Design in the beginning (shown in Figure 2), later, team will use the optimized system design on the Z-pattern and 3rd Pattern. A typical experimental setup is shown in the figure. We use an EV3 LEGO robot and several poster boards of different colors (white, yellow, orange, and green).

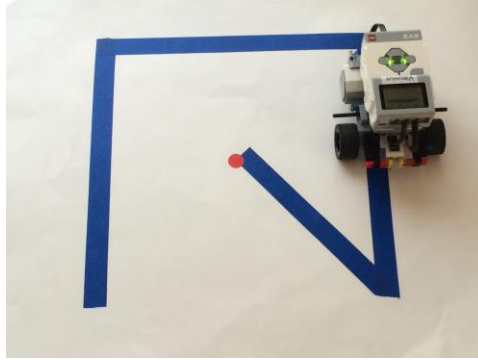


Figure 2. Base Pattern of Track Design

Project goal is to allow the robot to travel along the outer edge of the track and improve the cycle time. Before, team made the robot move on the inside of the track, but the problem was at the 45 degree turn. The robot could not make this inner turn because it was so sharp, so team changed the EV3 programming so the robot would move on the outside of the track. As seen in the Figure, the outer track has a less sharp angle than the inner track at this 45 degree turn.

1.4 Project Challenges

Based on the initial trials and investigation, team has identified the following project challenges in the simulation experiment:

- The robot turns too fast at the 45-degree angle location.
- The robot overshoots and goes out of the field.
- The robot turns by 90 degrees and deviates away from the corner.
- The robot travels almost vertically across the outer edge after turning at the 45-degree corner.
- The environmental light is not uniform. The variation is making significant impact on the robot on-track accuracy.
- The robot does not stop automatically at the end of the route.
- The robot does not move as expected.
- The robot travels too slowly.

The team conducted a root cause analysis for these challenges as shown in Table 2 below.

Table 2. Root Cause Analysis

Problem	Root Cause Analysis
Could not follow the track	<ul style="list-style-type: none"> a. Sensor location too high, Reflection intensity too weak, contrast too small b. Sensor too low, incoming light too weak, contrast too small c. Sensor in center, blocking light, robot components blocking light d. Ambient light intensity not uniform
Could not make sharp turn faster (speed too fast)	<ul style="list-style-type: none"> a. Single speed so you make turn at high speed, can't slow down. b. Robot not aware of turn c. Dimension – sharp turn – dimension too small compare to robot dimensions

1.5 Project Scope

SIPOC is a very powerful project management tool to define the project scope. SIPOC [7] is defined as: S - Supplier, I - Input, P - Process, O - Output, and C – Customer (Table 3). SIPOC starts with the Customer and ends with the Supplier.

Table 3 SIPOC Analysis

Suppliers	Input	Process	Output	Customer
Lego Mindstorms EV3	Field	Hardware: Build, Architecture	Cycle Time	Big City-Citizens Work in Traffic Hours
SPSS Data Analysis	Color Sensor Height	Software	Accuracy => Location Time	A lot of drivers lose their jobs
Lego Hardware	Number of Wheels	Collect Data, SPSS Analysis	Safety	Fewer cars on highway and parking lot
	Battery Life	Root Cause Analysis	Repeatability	Less Traffic
	Background Color	Improve	Repeat Results	You can go to work faster and save money
	Shadow	Optimize	Power => Less => Hardware Height	
	Brightness	Complete or Rebuild	Speed	
	Gravity		Reliability	

- Customers are the people who care and should define our product (customer driven).
- Outputs are what the customers really want: these are the true values customers are willing to pay for.
- Processes are the major steps which we can take to continuously improve our product in order to deliver the desired outputs.
- Inputs are the variables that we can optimize to improve our outputs through the processes.
- Suppliers are the vendors who will provide inputs to build and improve our product.

2. Measure Phase

2.1 Field Design

Team used several poster boards of different colors as the background template. Team mainly investigated the problem with the white board first since it will provide us the highest reflected color intensity. Team used blue tape to simulate the field the taxi travels on the way to pick up the five passengers. The field is designed to have a size of about the double length of the robot to make this project very challenging. The relative 2X field dimension has made Robot very difficult to move at fast speed when making 45 degree turns.

2.2 Hardware Design

The robot has two motors to control the two front wheels individually. A color sensor located in the front provides the reflected intensity and ambient light intensity to the input slot No.3. The program outputs to EV3 Brick (like Controller) slot #B and #C to instruct the wheel motions. The two front wheels have a diameter of about 1.75 inches. A metal ball with the diameter of 3/4 inches is installed in the back of the robot for better turning support. Team has spent significant time to optimize the Robotics Hardware Architecture in order to improve the turning efficiency.

2.3 Software Design-

LEGO MINDSTORM EV3 Software

The program is to keep the robot to track in the vicinity of the outer edge of the field pattern.

Basic algorithm (shown in Figure 3):

Measure the reflected intensity in the vicinity of the outer edge of the track along the path to find multiple thresholds X_1 and X_2 chosen as follows:

X_1 : the mean reflected intensity + 10

X_2 : the mean reflected intensity - 10

- Team has decided to use ± 10 intensity range to set up the threshold window.
 - If the window is too narrow, Robot will make more frequent but smaller swings.
 - If the window is too wide, Robot will make less but much bigger swings.

If the Robot Sensor is completely on the White background, the reflected color intensity will be higher since the white background can return more lights to the Sensor. If the Robot Sensor is completely inside the Blue Tape, the reflected color intensity will be lower since the Blue Tape will adsorb most incoming lights and return less to the Sensor. Team choose the White Background and Blue Tape track to maximize the contrast in order to provide better differentiation for Robot sensor to detect the track edge by applying the following algorithm to adjust the Robot movement along the track edge. As shown in below Figure 3, team has developed the EV3 Algorithm to control the Robotics movement.

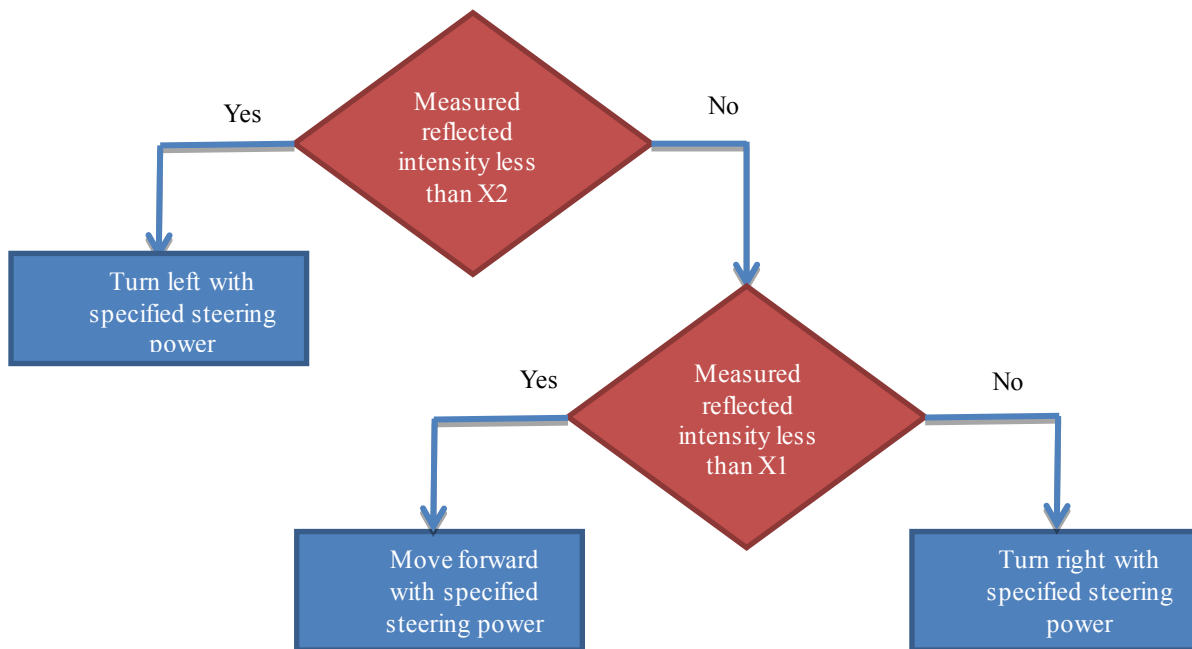


Figure 3, Software Algorithm Flow Chart

IBM SPSS Statistics Software

Team used the statistical tool in the SPSS software to analyze the measurement data and gain insight and ideas for improvement. Team downloaded the EV3 Raw data into SPSS Software in order to look at data pattern which may not be visible by human eyes.

3.4 Initial Observation

In initial attempt, the color sensor is far in front of the wheels and relatively high away from the field. Figure 4 is a scatter plot that shows the effect of environmental ambient light during the robot's motion path. The beginning of the chart shows the straight path of the robot when there is enough ambient light. The unevenness or skewness on the latter part of the chart is when the robot made the 90 degree turn and there is less ambient light. The location and placement of the robot's path was on the side of the room where there was less ambient light. It affected

the ambient light intensity and also the color intensity. When the robot made the first 90 degree turn, it blocked the light that shone on the path and its shadow was cast on the track as it moved. This resulted in skewness of the data.

As an improvement, it was considered moving to the center of the room. If the robot is in the center of the room, light will shine on all sides of the robot, resulting in a lot of ambient light. When the robot takes the turns, there probably won't be a problem of less ambient light, which could get rid of the skewness in the data.

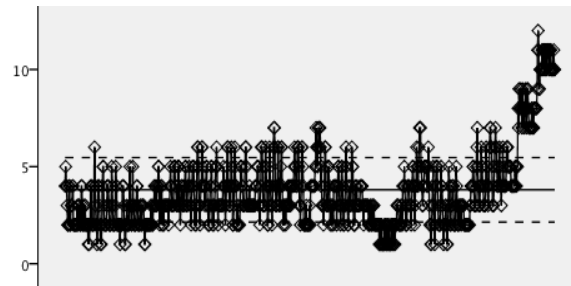


Figure 4. Control chart of ambient light

3.5 Baseline Capability Data

Team has collected the reflected intensity data along the path and obtained a mean of 32.41 with a standard deviation of 4.21. Several peaks were observed in Figure 5 Histogram.

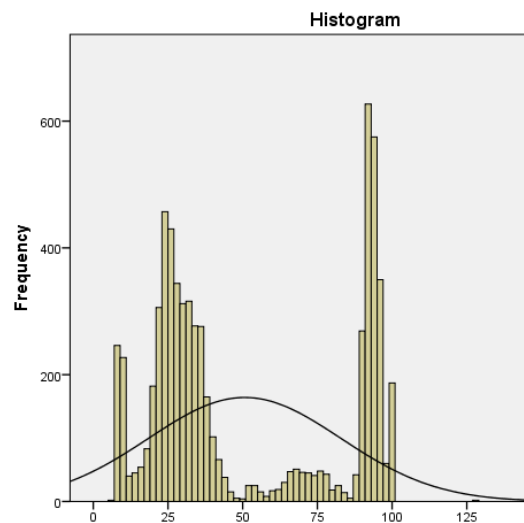


Figure 5. Reflected Intensity Histogram

Since the distribution is clearly the multi-modal distribution. Based on the intensity calibration record, team could attribute each mode to the following region:

- 1st Mode at around 10: when Robot Sensor is completely inside the Blue Tape
- 2nd Mode at around 25: when Robot Sensor is right at the Blue Tape Edge
- 3rd Mode at around 70: when Robot Sensor is completely in the white background
- 4th Mode at around 85: when Robot Sensor is completely in the white background where is in the brighter environment
- Two different modes at 70/85 are due to the environmental effect, and which has indicated that Robot may make a bigger deviation turn at 45-degrees (4th mode) and a smaller turn at 90-degrees (3rd mode). These two higher modes have provided a very reliable metric to measure how Robot made turns at these critical corners.

- Only the 2nd Mode Distribution (in range 10-45) is desirable. The project objective is to eliminate the other modes to improve the Robot on-track performance.

Through observation, we have found several problems to tackle with:

1. The robot does not follow the track (Modes at 10, 70, 85)
2. The robot does not turn immediately at the corners. It gets back to the track after traveling an arc distance. This results in multi-mode distribution. The bigger portion at Modes 70/85 will indicate a bigger turn at corners when making 45 degree or 90 degree turns.
3. The robot swings along the way indicated by a wide distribution associated with the 1st Mode around 25.
4. The robot moves very slowly forward during the swing.
5. The robot does not stop at the end of the track since no stopping guidance provided.
6. The robot does not turn at the inner angle such as in the “Z”-pattern. The inner angle will allow less turning space for the Robot to make turns than outer angle. Z-pattern could not avoid the inner angle problem.

It's amazing that team could utilize a very simple SPSS Histogram analysis and conduct a very thorough systematic problem root cause analysis by data-driven approach. Team did observe major Robot failures frequently, but without downloading the raw data and analyzed by SPSS analysis, team was hopeless and clueless until seeing this Histogram Analysis. Histogram analysis has provided a solid and objective evidence of two higher modes of off-track movement as well as swing behavior. Team has been progressing much faster since using the SPSS to analyze each EV3 Robot Raw Data. This is a wonderful learning experience for our younger kids to really learn what the data-driven means in solving the real science projects.

Based on the baseline analysis, team has decided to use the **Histogram Mode Analysis** as a major performance metric in addition to the total cycle time. There are many alternative ways to set the performance metric. Though, team wants to make this project simple and, based on the project challenges, it is making sense that making a much smaller turn is the most critical design challenge for both safety concern and the total cycle time requirement. This Taxi Service will be provided mainly in a BUSY campus like Google or Big City like New York City. The safety and on time delivery are the two most important performance requirements.

Team can imagine how important of making a smaller turns at both 45-degrees and 90-degrees situations in a real situation:

- How about making U-turns in a big city downtown area at rush hours?
- How about driving near the Lake or River and make a Right Turn onto the Bridge?
- How about driving on the hills along the cliff side?
- If the self-driving taxi could not control their turn at these turns, the consequence can be very risky.

Team could make a solid agreement that the SPSS Histogram Mode Analysis will be the major performance metric.

4 Analysis and Improvement

4.1 Sensor Position

The sensor position plays an important role in the measurements. Initially, the sensor was protruding in front of the front wheels with about 2 inches of distance. It was also about 1-inch high from the board. This could have introduced weak reflected intensity and resulted in small contrast. This contrast is the signal level, and the other variations (such as environmental light uniformity, Swing, Vibration, Efficiency, Mechanics..) will contribute to the noise. A higher Signal-Noise Ratio will make Robot follow the Blue Track Edge more accurately. If the Sensor position is too high, the returned light will be weaker due to the diffusion effect, resulting in a much weaker contrast or signal-noise ratio. A weaker contrast (signal) will make Robot more difficult to tell where is the Blue Edge since the noise level may confuse the Robot's vision on where is the Blue Tape Edge. Also, the Sensor Location is critical. The sensor should be placed closer to the Body Mass Center. If the sensor is placed to front, the rotation dimension (sensor-body center) will be too big to make a sharp turn (like turning a long train). Through this STEM project, team could learn both Mechanics and Optics in addition to EV3 Programming and SPSS Statistics. Team has repositioned the sensor so that it is at the front edge of the wheels and about ¼-inch from the board. This has significantly improved the robot performance, especially on the turning as shown in Figure 6.

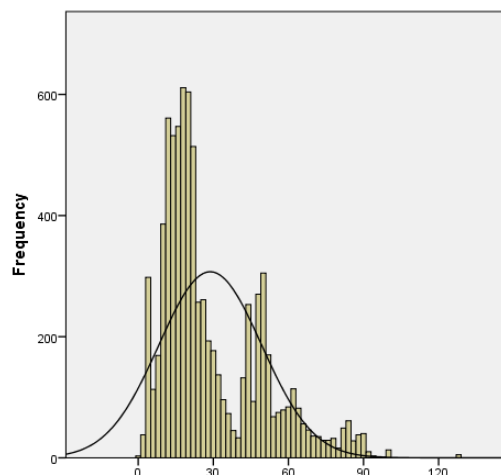


Figure 6. Reflected Intensity Histogram

In the Figure 6, the extraneous modes (1st Mode and the 4th Mode) disappeared.

The first mode (around 10) disappeared has indicated that Robot can make a more efficient swing adjustment to prevent too much penetration into the Blue Tape area completely. The improved Sensor design has allowed Robot more sensitive to detect the Blue Tape edge and make a quicker adjustment back to the Blue Tape edge. The fourth mode (around 85) also disappeared has indicated that Robot won't detour too much away from the edge when turning at 45-degree angle. The Robot can make a much smaller turn like at 90-degree angle.

The robot tracks the edge better at the sharp turns. However, this is achieved at the cost of very low speed with the undesirable total cycle time. The curve shows normal distribution and improvement from the baseline Histogram data. There is still some skewness seen for potential improvement. Team was very excited to see this improvement based on a very simple SPSS Histogram Analysis. It's really amazing that SPSS data analysis can provide objective evidence in details that our eyes could not catch during the robot movement.

4.2 Hardware Adjustment

In order to further reduce the total cycle time, after another brainstorming session, team has implemented the following hardware adjustments:

- Change to one whole battery pack (instead of 6 batteries) to reduce the robot weight by 70 grams (10%). Lighter Robot can consume less energy and makes Robot moves faster.
- Lower the robot brick to be closer to the motor by changing the connecting LEGO pieces. The lower Mass body center can help Robot move better at faster speed at the corners to overcome the centrifugal forces.
- Change the back wheel to Ball wheel design to help in hard turns, which can provide more freedom to make sharp turns.

4.3 Turning Improvement

In experimental observation, it typically takes about 10 seconds for the robot to get back to the track at each turns. Through experiment, we found that the turns should be taken care of on top of the regular left and right adjustments. Team has further modified the algorithm as follows:

Measure the reflected intensity in the vicinity of the outer edge of the track along the path to find the optimum thresholds X_1 , X_2 and X_3 :

X_1 : the mean reflected intensity + 10

X_2 : the mean reflected intensity - 10

X_3 : the mean reflected intensity + 20

This new added algorithm is to force Robot back to the track if deviating too much from the edge. The "+20" amount level is based on the previous Histogram diagram shown in Figure 6: the 2nd mode is around 20 higher than the distribution edge of the 1st mode. After referring to the histogram distribution modes, team thought these

threshold levels were reasonable and which could provide team a very accurate information and feedback control on the Robot swing efficiency in order to follow the Blue Tape edge on the corners.

4.4 Speed Improvement by Setting Independent Motor Control

Team found that, when making direction adjustment, controlling the two motors independently could help improve the speed while maintaining the same road tracking performance. In the programming implementation, alternate the motion of two motors based on the amount of reflected intensity detected by the color sensor, i.e., driving one motor with certain power while keeping the other still. Drive both motors with the largest power when the intensity becomes very high, indicating an angle has reached. The objective is to increase the speed but still be able to make the turns.

As shown in the Figure 7, at the most optimal setting, the intensity mean has maintained at 30 close to the calibrated ideal Blue edge intensity at 32. At the same time, the total traveling time has been reduced by 10 seconds. The cons here are that relatively large swings in the robot motion, which results in a large standard deviation of 20. It is observed that the swinging is acceptable in the sense that it does not introduce a second mode in the distribution. The large standard deviation may not be a bad news since the second mode has been almost eliminated. The second mode has been shrunk to becoming the right skew mode. This is an improvement from the previous runs, which has a large amount of skewness or 3rd mode. This change could have been result of bringing the sensor closer to the robot and lowering the sensor. Team was even more excited to see how the Histogram distribution pattern evolving: from four major modes, down to 2 major modes, now only one major mode. Ideal case, we should have only one major mode approximately in Normal Distribution.

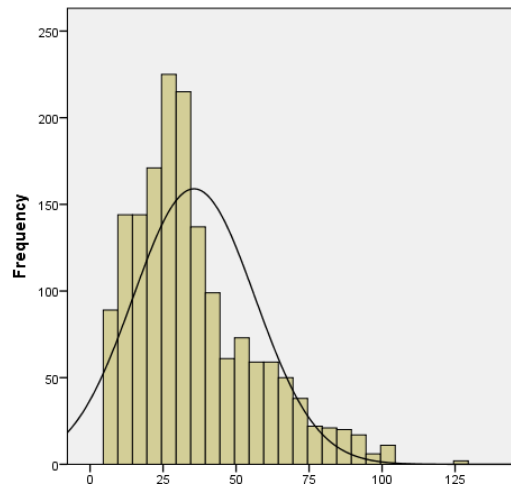


Figure 7. Reflected Intensity Histogram

4.5 Dealing with the Inner Angle of Z-pattern

The algorithm team has used so far still failing at the second turn in the “Z”-pattern, which essentially hits the inner edge scenario. Thus, team still has to deal with the sharp turn at the inner edge with the following algorithm improvement:

Measure the reflected intensity in the vicinity of the outer edge of the track along the path to find multiple thresholds X_1 , X_2 and X_3 :

X_1 : the mean reflected intensity + 10

X_2 : the mean reflected intensity - 10

X_3 : the mean reflected intensity + 20

X_4 : the mean reflected intensity - 20

- When the measured reflected intensity is less than X_2 but greater than X_4 , let the robot turn left with specified steering power.

- When the reflected intensity is less than X_4 , let the robot make the largest turn to the left with specified steering power.
- When the reflected intensity is greater than X_2 but less than X_1 , let the robot move straight forward with specified steering power.
- When the reflected intensity is greater than X_1 , let the robot turn right with specified steering power.
- When the reflected intensity is greater than X_3 , let the robot make the largest turn to the right with specified steering power.
- Without adding two more threshold levels, the Robot could not survive the inner angle at sharp turns.

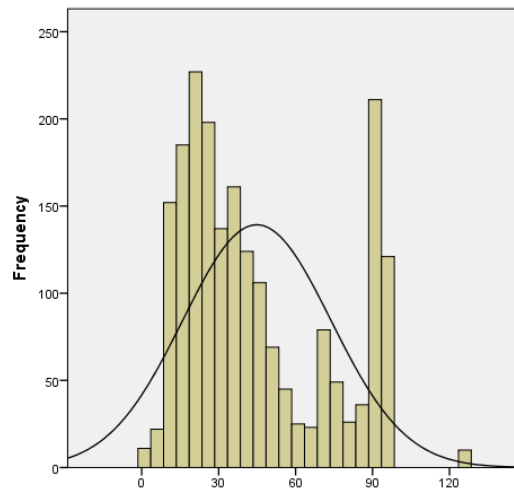


Figure 8. Reflected Intensity Histogram

As shown in Figure 8, although the robot reacts to the sharp turn at the inner angle, the large swings introduce the extraneous mode in the high intensity vicinity. There is lot of skewness, which should be fixed. There is an outlier around 120. The higher 2nd mode has indicated the difficulty to control the Robot turn at inner angle. It's still a challenging to team on how to overcome this Inner Angle unless Robot speed can be further reduced or/and Robot dimension can be shrunk.

After completed the Z-pattern, team also tried their Robot and EV3 program on the third pattern as shown in Figure 1 Pick Up Patterns. The performance on the whiteboard with the 3rd pattern is fairly good with a mean of 35.17. The standard deviation is still big because we can't avoid the swings to maintain the speed. There is an outlier around 125 and we should try prevent getting outliers in the data. There is little skewness, which is an improvement from the last runs, but we are aiming to get normal distribution and have no outliers.

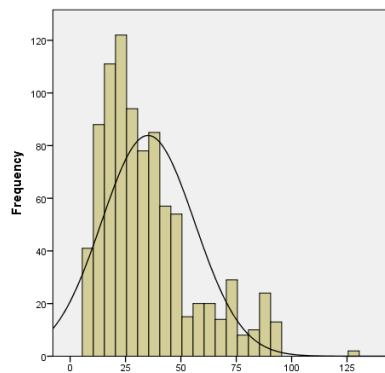


Figure 9. Reflected Intensity Histogram

In summary:

- Base Pattern has one 45 degree turn and three Straight lines and all along outer angle.

- Z Pattern has two 45 degree turns and three Straight lines and along one inner and one outer angle
- 3rd-Pattern has one 45 degree turn and two Straight lines and all along outer angle.

Z pattern has the worst performance due to two possible failure modes: (1) two 45 degree turns, (2) Inner Angle. The main failure mode should be due to the Inner Angle, which makes our Robot almost impossible to overcome this Inner Angle challenge due to the Robot/Field Track dimension constraints.

4.6 Develop Ending Stop Algorithm

Initially, the robot did not stop at the end point but strolls away. It has to be manually stopped in the end. Previously, team used an orange sticker. The robot did not sense it, so it kept moving. Team has fixed this issue by adding a red and bigger sticker at the end of the track as a stop indicator. When the color sensor detects the red sticker color, it will stop immediately. Red stick color was selected to enhance the contrast between Blue Tape Color, White Ground Color and Red Stop Color.

4.7 Comparison with different color template boards

Team also run the program on different color boards to simulate environmental color or light variations. It is observed that the same implementation works on orange and yellow boards since both provide good contrast. While on the green board, the intensity thresholds have to be adjusted as the green color has close intensity with the blue color track. With the adjustment, all boards have comparable performance as shown in Figure 10. This experience has shown us a good lesson to measure the contrast and to determine the signal-noise ratio as the fundamental statistics. This information is useful to test Design Robustness of tolerating the environmental deviations.

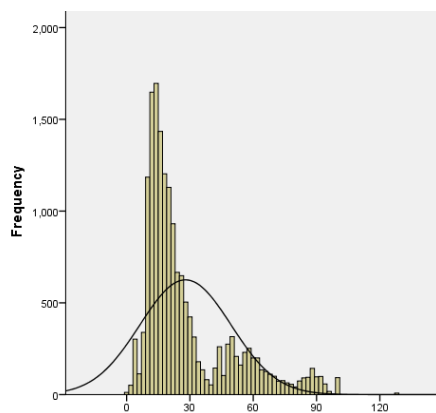


Figure 10. Reflected Intensity Histogram

5 Conclusion

The reason for skewness in the data is because the robot keeps swerving. The color sensor first detects white, then blue, and then white again. Because white has a very large reflected light intensity compared to blue, the intensity varies a lot, which is shown as skewness in the data charts. This experience has shown us a good lesson to measure the contrast and to determine the signal-noise ratio as the fundamental statistics. Team could predict the Robot performance by measuring the contrast first. Also, team has learned how to utilize SPSS data analysis. Most project root cause analysis could be explained by a very simple Histogram Mode Analysis. Each mode has indicated each unique robot movement pattern. Team can rely on this simple Histogram Analysis to attack a very complicated Robot Mechanics and Software Programming like an experienced Robotics Engineer. Team was so thankful, through IEOM STEM Project, team can integrate Robotics Mechanics, Software Programming, SPSS Statistics, Project Management, and Team Building in one amazing project.

Acknowledgements

Team was so lucky that we have mentor Dr. Charles Chen who could help us complete this IEOM STEM Project. Our Statistics mentor Dr. Charles Chen has helped us how to use AP Statistics and Minitab software to

analyze our data. We would also like to thank our parents Chia Lin, Huihui for their endless love to support our IEOM Tesla STEM Project.

References

- [1] Google Car Self-Driving <https://www.google.com/selfdrivingcar/>
- [2] Google Car Project https://en.wikipedia.org/wiki/Google_self-driving_car
- [3] Lego Mindstorms EV3 <https://www.lego.com/en-us/mindstorms/learn-to-program> [3] David Held, Jesse Levinson, Sebastian Thrun, Silvio Savarese. Robotics: Science and Systems (RSS), 2014.
- [4] A. Teichman and S., Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2013.
- [5] "2013 World Green Car." :: World Car Awards, [www.wcoty.com/web/eligible_vehicles.asp? year=2013&cat=4](http://www.wcoty.com/web/eligible_vehicles.asp?year=2013&cat=4)
- [6] By Katie Collins January 11, 2016 9:59 AM PST . "Tesla Cars Can Now Self-Park at Your Command." CNET, 11 Jan. 2016, www.cnet.com/news/tesla-cars-can-now-self-park-at-your-command/.
- [7] Simon, Kerri. "SIPOC Diagram". Ridgefield, Connecticut: iSixSigma. Retrieved 2012-07-03.

Biography

Mason Chen is a certified IASSC (International Associate of Six Sigma Certificate) Lean Six Sigma Black Belt. He has also certified IBM SPSS Statistics Certificate, IBM Modeler Data Analysis and Data Mining Certificates. He also won the 1st Place Award this year on the Mental Math and Abacus Math contests in the North California Region, USA. Mason Chen is familiar with Lego Robotics/EV3, Six Sigma DMAIC, DMADOV, Lean Production, Minitab, SPSS Statistics, SPSS Modeler CRISP Data Mining, AP Statistics, JAVA Programming, and ASQ (American Society for Quality) Quality Engineering. He is the founder of Mason Chen Consulting which organization helps develop young kids on Big Data Analytics and STEM Projects. Recently, he is also learning "Computational Biology" which integrates Biology, Chemistry, Physics, Mathematics, Statistics, and Computer Science fields.