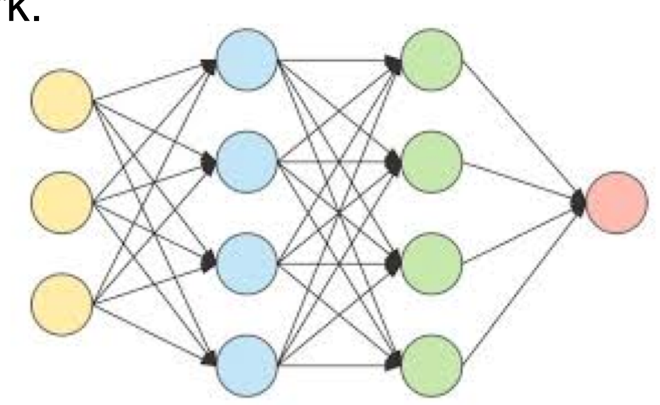
Design of Safe Self-Driving Vehicles

Rachel Loh, rachelloh03@gmail.com
Arcadia High School, Class of 2021
USC Viterbi Department of Electrical and Computer Engineering, SHINE 2020

Introduction

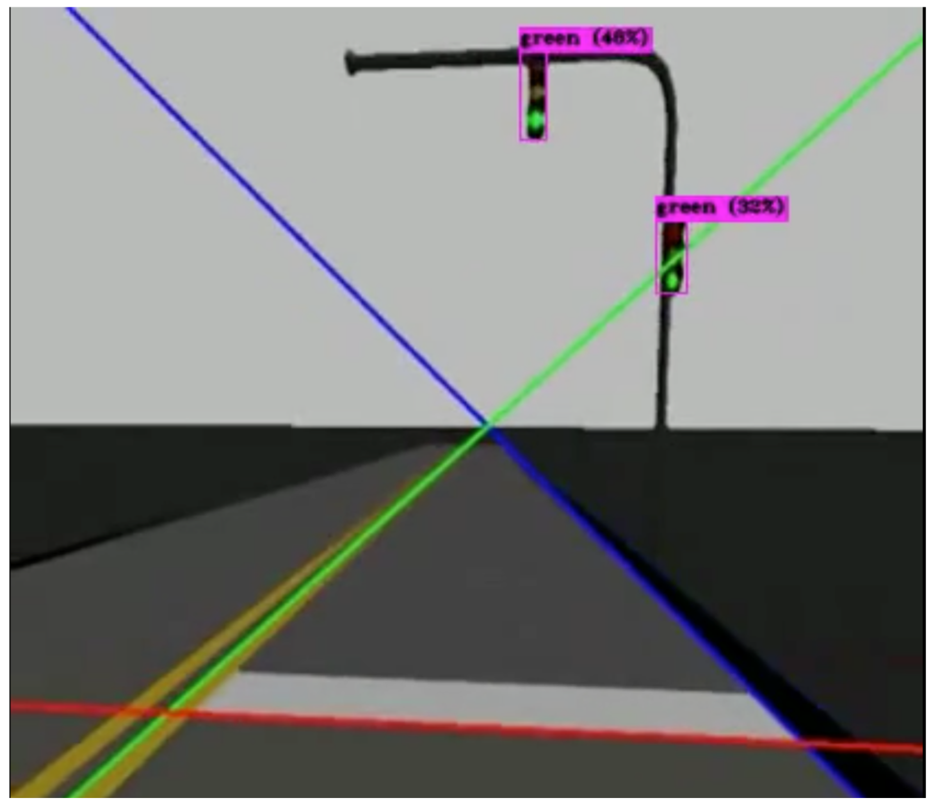
Existing methodologies do not support efficient and accurate design and verification of cyber-physical systems (CPSs) with machine learning components due to the complexity and heterogeneity of modern CPSs. To solve these challenges, we first propose a framework which enables creation of CPSs with machine learning components. Then, we propose mathematical methods to verify the CPSs created from the framework.



Neural network diagram. PC: KDnuggets

Objective & Impact of Professor's Research

The objective of DesCyPhy lab's research is to design and verify modern CPSs (including autonomous vehicles) such that they are safe and reliable. This is done by creating and training neural networks using data collected from multiple simulation environments, designing safe control strategies and algorithms to actuate the vehicles, and using mathematical methods to verify the safety of the overall system.



Control algorithm detection of lane and neural network detection of green traffic light in simulation world. PC: Rachel Loh

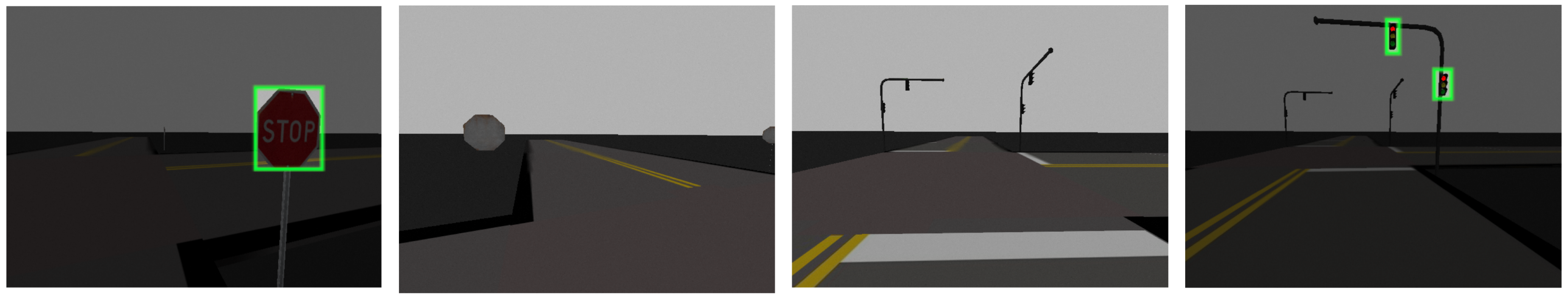
Research Process

Conversion of Simulation World Files From ROS1 to ROS2:

- ROS2 provides improved network communication, supports isolated builds only, and enables more complex logic in launch files

Data Curation:

- 1,200 images from 6 simulation worlds
- Use rqt_image_view and keyboard teleop to collect data
- Use labelImg to label data
- Use Roboflow.ai to convert image annotations to YOLO Darknet
- Split data into 70% train, 20% validate, 10% test



Training data samples from Roboflow.ai after labeling and conversion. PC: Rachel Loh

Customization, Training, and Testing of Neural Network for TurtleBot3's Perception:

- Use a pre-existing YOLOv4-tiny neural network architecture in Google Colab
- Train the neural network to recognize stop signs and traffic lights
- Adjust batch size and learning rate to achieve highest accuracy
- Test the neural network using the best weights

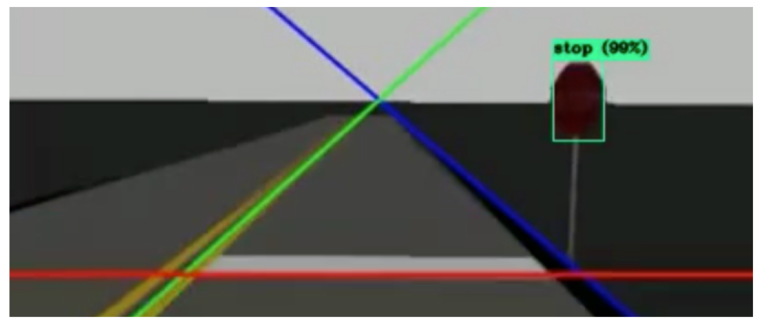
Neural network detection after training. Collected from Google Colab. PC: Rachel Loh

Attempt (#)	1	2	3	4	5
Batches (#)	64	64	64	64	128
Learning Rate	0.00261	0.00200	0.00139	0.00232	0.00261
Best Accuracy (%)	68.57	68.52	68.21	69.38	70.78

Fig. 1. Batch size, learning rate, and best accuracy for each training attempt. Created in Google Slides.

Analyzing, Modifying, and Running Control Algorithms:

- Python from GitHub
- Lane detection
- Lane controller linear code



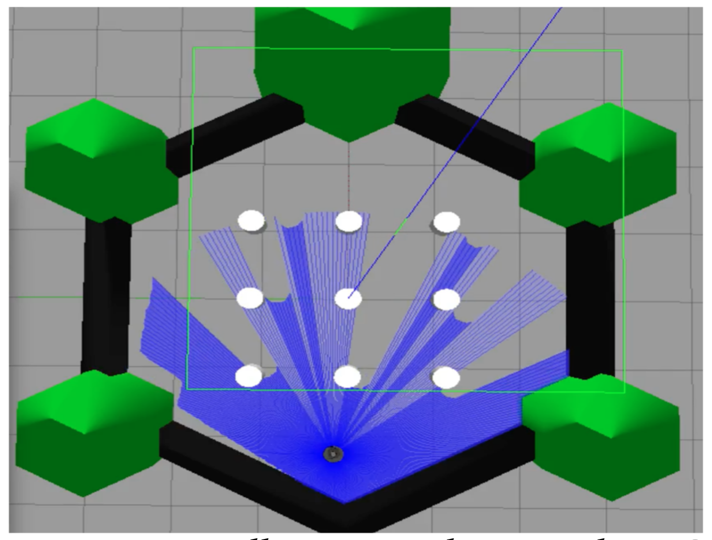
Control algorithm detection of lane and neural network detection of stop sign in simulation world. PC: Rachel Loh

Next Step for the Lab:

- Verification of the Turtlebot3 control system with a neural network-based perception component using mathematical methods

How This Relates to My STEM Coursework

By taking AP Computer Science A, I learned the computational thinking and fundamental concepts needed to understand the TurtleBot3 code and machine learning components. SHINE showed me how computer science can positively impact society, a perspective I can share with Coding Club.



TurtleBot3 using collision avoidance code in Gazebo TurtleBot3 World. PC: Rachel Loh

Citations

[1] Ivanov, R., et al. (2019). Case Study: Verifying the Safety of an Autonomous Racing Car with a Neural Network Controller. *23rd ACM International Conference on Hybrid Systems: Computation and Control.*

[2] Jia, R., et al. (2018). Design Automation for Smart Building Systems. *Proceedings of the IEEE.*

[3] Nuzzo, P. (2019). From Electronic Design Automation to Cyber-Physical System Design Automation. *Proceedings of the 2019 International Symposium on Physical Design.*

Acknowledgements

Thank you Professor Nuzzo for welcoming me into your lab. Thank you to my lab mentor, Chanwook Oh, my Center Mentor, Erica De Guzman, my lab mate, Branden Leong, Dr. Mills, Cathalina Juarez, and the SHINE team.