

# JiaWei Lee

☎ +886 912 911 672 | @ qw960211@gmail.com | GitHub

Software engineer with 2 years experience in Turing-Drive, dedicated to researching path planning and GNSS positioning, the main programming languages is C++, C, Python

## Education

---

### National Yunlin University of Science and Technology

Department of Electrical Engineering

- Mobile robot, Robot Operating System(ROS)

Yunlin, Taiwan

Sep. 2017 – Jun 2019

### National Formosa University

Department of Electrical Engineering

Yunlin, Taiwan

Sep. 2015 – Jun 2017

## Professional Experience

---

### Turing Drive Inc(C++/C, Python, ROS, Shell, Linux, Path Planning, GNSS)

Software Engineer

Taipei, Taiwan

Nov 2019 – Present

- Optimized the path planning algorithm and the integrated obstacle information in the Autoware system, execution speed is more than 60% faster, total mileage exceeds 5,000 km, and over 4,000 people have been onboard
- Designed and implemented NMEA parser and NTRIP Caster available to the projects positioning system
- Designed and implemented data log to path planning system, used for algorithm debug, improve on-site work efficiency by about 50%
- Developed a through 2D LiDAR detection person leg and 3D LiDAR camera fusion

## Selected Projects & Awards

---

### Quantitative Trading(Python)

May 2022- Present

- Developing the trading strategy
- Get current Cryptocurrency price using the Binance API

### Smart trash can(C++/C, ROS, Navigation Stack, SLAM, Opencv, PCL)

Sep 2017 – May 2019

- Designed and Integrated SLAM module, navigation module and sensor module through the ROS, robot served aperiod of time at the NYUST EC404 laboratory
- Developed 2D LiDAR leg detection to using the Adaboost, the detection accuracy on the empty environment is more than 90%
- Researched and Developed 2D LiDAR SLAM algorithm and Navigation Stack

## Skills

---

**Languages:** C/C++, Python, Shell scrip

**Frameworks:** ROS, ROS2

**Developer Tools:** Git, Docker, VS Code, Vim, Cmake,

**Libraries:** Eigen, PCL, Opencv, Pandas, NumPy, Matplotlib