

# OLIVER HUANG

☎ 812-241-9901 ✉ [siyuanhuang97421@gmail.com](mailto:siyuanhuang97421@gmail.com) [in linkedin.com/in/username](https://www.linkedin.com/in/username) [github.com/username](https://github.com/username)

## Education

### Worcester Polytechnic Institute

*Master of Science in Robotics Engineering*

Aug 2022 – May 2024

Worcester, MA

### Rose-Hulman Institute of Technology

*Bachelor of Science in Mechanical Engineering*

Aug 2015 – Feb 2019

Terre Haute, IN

## Experience

### MathWorks

*Software Engineer Intern*

May 2023 – August 2023

Natick, MA

- Crafted a **C++** test infrastructure leveraging **Google Test** fixtures to automate the management of **Redis** server lifecycles within a Continuous Integration/Continuous Deployment (CI/CD) pipeline.
- Engineered a **multi-process handshake** protocol to safeguard against Redis server process drooling in the event of an unit test crash.
- Established **cross-platform** testing capabilities spanning a range of operating systems.

### DEKA Research & Development Corp.

*Control Systems Engineer–Autonomous Driving*

Mar 2019 – Mar 2022

Manchester, NH

- Designed and implemented a **Model Predictive Controller (MPC)** in **C++** for trajectory planning and obstacles avoidance. (Tested over 1000 miles on real world sidewalks)
- Adapted global and local **optimization algorithms** to find optimal solutions towards global minima.
- Introduced a parallel computing algorithm with **CUDA** to calculate an Euclidean distance map for collision checking. (Achieved 60% calculation speed increase for  $1024^2$  occupancy grid.)
- Spearheaded the design and execution of an **automated data collection** pipeline for on-demand data acquisition with **Robot Operating System (ROS)**.
- Led the initiative to design and implement a multi-controller framework, enabling on-demand rapid controller switching through the utilization of **multi-threading** and shared memory techniques.
- Customized **RVIZ** for engineer-centric debugging visualization and utilized **Foxglove** for customer-facing demonstration visualization.

## Projects

### Monocular camera traffic scene regeneration | *Python, PyTorch, JSON, Blender*

Mar–Apr 2023

- Applied Cross Layer Refinement Network (CLRNet) and Bezier curve fitting techniques to execute lane detection.
- Adopted YOLO-V7 for traffic scene objects detection. (car, truck, bicycle, motorcycle, person, traffic signs...)
- Utilized Deep Learning in conjunction with geometric constraints to estimate the 3D pose of the detected objects.
- Estimated pedestrian 3D poses by leveraging a 3D Human Pose and Shape Regression algorithm. (PyMAF)
- Effectively handled dependencies among multiple Git repositories using Docker.
- Automated traffic scene reconstruction rendering through the use of JSON intermediate representation and Blender Python scripting.

### Drone quaternion attitude estimation | *6-DoF IMU, Madgwick Filter, UKF*

Sep 2023

- Implemented quaternion-based attitude estimation for a 6-DoF IMU, employing both the Madgwick Filter and Unscented Kalman Filter (UKF) algorithms.
- Conducted benchmarking of attitude estimation results from both filters against ground truth data obtained from Vicon motion capture system.

### Monocular camera calibration | *OpenCV, Python*

Feb 2023

- Applied Zhang's method to estimate monocular camera focal length, distortion coefficients, and principle point.
- Utilized OpenCV to perform corners detection for checker board calibration pattern.

### 3D scene sparse reconstruction with structure from motion

Feb–Mar 2023

- Extracted and matched monocular camera image key features using SIFT descriptors and RANSAC algorithm.
- Computed the fundamental and essential matrices based on epipolar geometry.
- Determined the camera pose from the essential matrix and ensured Cheirality Condition compliance through linear and non-linear triangulation.
- Aligned multiple images (camera poses) utilizing both linear and non-linear Perspective-n-Point (PnP) algorithms.
- Performed scene reconstruction using bundle adjustment, aided by the visibility matrix.

### POV (persistence of vision) display | *ESP32, Embedded C*

Nov–dec 2022

- Engineered the mechanical and electrical assemblies of a POV display featuring rotating LED strips.
- Developed firmware for an ESP32 micro-controller to monitor the rotational speed of an LED strip using a hall effect sensor with hardware interrupts.
- Employed timer interrupts to precisely control the LED refresh rate, enabling the persistence of vision effect.

**Photorealistic scene reconstruction with NeRF** | *PyTorch*

**Mar 2023**

- Implemented the original neural radiance field (NeRF) paper using the PyTorch framework.
- Reconstructed self-collected scenes using NVIDIA's Instant-NGP NeRF pipeline.

**Multi-agent path finding** | *ROS, PyGame*

**Mar-May 2023**

- Created a custom PyGame simulator compatible with ROS (actionlib) to facilitate multi-agent pathfinding.
- Computed optimal collision-free paths for scenarios involving more than 50 agents on a 128x128 grid map using both the Conflict-Based Search and Symmetry Breaking Conflict-Based Search algorithms.