

# Jeel Chatrola

🐙 Github: jeelchatrola | 🌐 LinkedIn: jeel-chatrola | ✉ Email: jeelchatrola046@gmail.com | 📞 +1 (774) 519-9378

## EDUCATION

---

### Worcester Polytechnic Institute

Worcester, MA

Master of Science - Robotics Engineering (GPA: 4.0/4.0)

## SKILLS SUMMARY

---

<b>Programming Languages</b>	Python, C++, C, MATLAB
<b>Frameworks</b>	PyTorch, TensorFlow, Scikit-learn, OpenCV, Open3D, PCL, MLflow, ROS/ROS2, MoveIt, OMPL, Ray
<b>Tools</b>	Bash/Zsh, Git, Linux, Docker, Bazel, Nix, Simulators (Gazebo, Pybullet, Isaac Sim, MuJoCo)
<b>Hardware Platforms</b>	Embedded Systems (Teensy, ESP32, Raspberry Pi, Nvidia Jetson), Stereo Camera, LIDAR, IMU, Turtlebot3, UR10
<b>Other Skills</b>	Version Control, Debugging and Profiling, Software Development, Functional Programming, Software Design Patterns, Documentation

## WORK EXPERIENCE

---

### Torc Robotics

Ann Arbor, MI

Robotics Software Engineer - Simulation

*Aug 2025 - Present*

- Contributing to developing a scalable simulation stack with **Ray**, boosting simulation throughput by over **30%** for closed-loop testing of autonomous driving systems.
- Developing a framework to support **Reinforcement Learning** workflows, enabling validation of new algorithms across **2,000+** diverse simulation scenarios and accelerating policy iteration cycles by **25%**.
- Collaborating with 3 cross-functional teams (Perception, Planning, and Controls) to gather requirements, leading to a **20%** reduction in validation time for new features.

### Magna Services of America - Advance Manufacturing Innovation

Troy, MI

Robotics Software Engineer

*Aug 2024 - Jul 2025*

- Developed an automated synthetic data generation pipeline using **Isaac Sim** and CAD models for training robust perception models for automotive components, generating a dataset with over **500M+ grasp labels**.
- Implemented and training a **PointNet++** based grasp detection network on synthetic point cloud data, achieving an estimated **80-85%** grasp success rate in simulation, improving upon baseline geometric methods.
- Designed the perception and grasping modules as reusable **ROS2** components with standardized interfaces for deployment on varied hardware platforms.
- Benchmarked model inference performance on target hardware (**NVIDIA Jetson**) to meet real-time processing requirements for industrial automation tasks.

### Robert Bosch Centre for Cyber-Physical Systems, IISc Bangalore

Bangalore, India

Robotics Research Associate

*Jan 2022 - Jun 2023*

- Developed a comprehensive autonomous robotics framework in ROS2, integrating a real-time perception pipeline using **YOLOv7** and **DeepLabv3+** for object detection and semantic segmentation, optimized for **Jetson TX2** with 15FPS throughput.
- Implemented a multi-sensor perception stack integrating low-cost **LIDAR** and **stereo cameras** for ground robot navigation, utilizing **Kalman Filters** for sensor fusion to improve localization reliability by **10%** in dynamic environments.
- Managed a team of **20 interns** on research initiatives, establishing standardized documentation and testing procedures that improved project completion rates by 25% while ensuring long-term maintainability of deployed systems.
- Created comprehensive documentation and testing procedures for robotics software, ensuring maintainability and reliability of deployed systems.

## PROJECTS

---

### 3D Scene Reconstruction: NeRF vs SfM [↗](#)

*Jun 2024 - Jul 2024*

- Implemented and benchmarked Neural Radiance Fields (NeRF) against traditional Structure from Motion (SfM) for 3D reconstruction, achieving 12% better novel view synthesis with NeRF while SfM required 60% less computation time for comparable spatial accuracy (Python, OpenCV, COLMAP)

### Automatic Camera Calibration [↗](#)

*May 2024 - Jun 2024*

- Implemented Zhang's Camera Calibration method for intrinsic and extrinsic parameter estimation and lens distortion correction, achieving sub-pixel re-projection error on calibration datasets
- Utilized OpenCV to process checkerboard images and optimize camera parameters, enhancing accuracy of downstream vision tasks (Python, OpenCV)

### Panorama Stitching [↗](#)

*May 2024 - Jun 2024*

- Implemented advanced algorithms such as SIFT and SURF for feature extraction, brute force matching for feature alignment, and homography for image warping to combine multiple images into one (Python, OpenCV)

### Vision based Obstacle avoidance for quad-rotors [↗](#)

*Apr 2024 - May 2024*

- Created a Deep-Reinforcement Learning based framework to perform end-to-end motion planning for quad-rotor using motion primitives (Python, PyTorch, OpenCV)

### Mask R-CNN for Pose Estimation in Dynamic Environments [↗](#)

*Nov 2023 - Dec 2023*

- Fine-tuned a Mask R-CNN model with a ResNet-50 backbone using TensorFlow on the COCO dataset for accurate 2D human keypoint detection.
- Developed a real-time feedback application using OpenCV to visualize pose estimation results, assisting users in maintaining correct exercise form with processing latency under 100ms. (Python, TensorFlow, Scikit-learn, OpenCV)

### Path Planning Algorithms [↗](#)

*Sep 2023 - Nov 2023*

- Implemented 8+ path-finding algorithms from scratch, including BFS, DFS, Dijkstra's, PRM, RRT, RRT, and A\* (Python, C++, OMPL)

### Research Grade Modular Mobile Robot Platform [↗](#)

*Jan 2023 - Jun 2023*

- Developed a ROS-based hardware and software stack for a modular mobile robot, enabling algorithm development for perception and planning
- Implemented a closed-loop real-time controller in Embedded C with encoder feedback, achieving sub-10ms latency for precise speed and position control (ROS, Embedded C, Python)

### Autonomous Navigation for Ground Robots using Symbolic Controllers [↗](#)

*Jul 2022 - Dec 2022*

- Engineered a safety-focused autonomy framework for mobile robots, incorporating a symbolic motion planner with formal guarantees, improving navigation efficiency by 35% compared to baseline RRT\*
- Validated the framework in ROS/Gazebo simulations and on a Turtlebot3 platform, demonstrating robust performance in cluttered environments (ROS, Python, C++)

### Multi-Robot Safe Planning using Control-Barrier Functions [↗](#)

*May 2022 - Jul 2022*

- Implemented Control Barrier Function (CBFs) layer to ensure 100% safety compliance for testing of diverse swarm/multi-agent algorithms with motion capture, enhancing experimentation and reliability (ROS, Python)

### Vision Based Package Sorting using a UR5 Robotic Manipulator [↗](#)

*Dec 2020 - Feb 2021*

- Developed a ROS-based vision system for package sorting using a UR5 robot arm in a Gazebo simulated warehouse
- Implemented object detection using YOLOv5 for package identification. Integrated motion planning using MoveIt for efficient sorting tasks (ROS, Python, C++, OpenCV, MoveIt, Gazebo)

## PUBLICATIONS

---

- **Multi-layer Motion Planning with Kinodynamic and Spatio-Temporal Constraints** [↗](#)  
ACM Hybrid System Control Conference (HSCC - 2025)
- **Autonomous Exploration using Ground Robots with Safety Guarantees** [↗](#)  
IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS - 2023)
- **Scalable Controller for Multi-Agent Systems Using Barrier Functions and Symbolic Control** [↗](#)  
IEEE Conference on Decision and Control (CDC - 2023)
- **Funnel-based Reachability Control of Unknown Nonlinear System using Gaussian Process** [↗](#)  
IEEE Indian Control Conference (ICC - 2022)