

# UTKARSH GUPTA

Los Angeles, CA | (213) 696-8156 | utkarshg@usc.edu | linkedin.com/in/utkgupt

## EDUCATION

### University of Southern California

Master of Science in Computer Science

Relevant Coursework: Intro to Robotics, Deep Learning, Robotic Perception, Autonomous Systems

Aug 2024 - May 2026

(GPA: 3.9/4.0)

### Maharashtra Institute of Technology, Pune

Bachelor of Technology in Computer Science and Engineering

Dec 2020 - Aug 2024

(GPA: 3.8/4.0)

## SKILLS

### Languages and Platforms:

- Python, C++, ROS2, CUDA (C++), Linux

### Robotics Specific:

- Perception:** 3D Data Processing, Sensor Fusion (Camera + LiDAR), 3D Reconstruction (Structure from Motion), Digital Twins
- Motion Planning, Navigation, Localization:** Nav2, RRT, PRM, A\*, Pose Estimation, EKF, SLAM
- Simulation:** Rviz, Gazebo, NVIDIA Omniverse

### Machine Learning for Robotics:

- Tasks:** Vision-Language-Action (RT-X, Octo), Computer Vision (Segmentation, Classification), Reinforcement Learning (DQN)
- Libraries and Framework:** PyTorch, TensorFlow, Keras, OpenCV, SciPy, NumPy, scikit-learn

## RELEVANT EXPERIENCE

### Robotics Engineering Intern

May 2025 - Aug 2025

ERIC Robotics

- Built real-time Railtrack feature extraction and 3D safety box algorithms with LiDAR/cameras for safer dynamic AGV navigation
- Optimized AMR initial pose estimation using LiDAR scans, cutting localization time by 40% over manual and brute force inputs
- Deployed system wide monitoring/control scripts for lifecycle node for better system consistency and reduced downtime

### Robotics Research Assistant

Nov 2024 - Present

CPS Vida Lab, USC

- Conducting research on Multi-Agent Path Finding and Human-Robot Interaction for complex task breakdown and execution
- Multi-Agent Path Finding: Developing collaborative strategies between agents to find distributed goals using imitation learning
- Human-Robot Interaction: Leveraging language model policies to improve performance for low confidence actions in VLAs

### Machine Learning (Computer Vision) Intern

Jul 2023 - Dec 2023

Center for Development of Advanced Computing (CDAC)

- Improved quality of 9,000+ artworks by implementing denoising techniques for Gaussian, crack, and speckle distortions
- Increased PSNR by 10.2% on average through PSO optimization, resulting in better denoised image quality
- Maximized image quality and processing efficiency by PSO algorithm optimization and sharpening filter integration

## PROJECTS

### 3D Reconstruction of an Industrial Component for Digital Twin Integration | Digital Twin, 3D Reconstruction, Photogrammetry

- Built a reconstruction pipeline to generate 3D models, integrating them into Digital Twins for industrial planning and inspection
- Used COLMAP for sparse reconstruction and Patch-Match MVS for dense point cloud, exporting models as glTF for visualization
- Applied Poisson surface reconstruction to create complete meshes from partial image scans (180–250°)

### Vision-Based Emergency Collision Avoidance & Rerouting | OpenCV, PyTorch, YOLOv5, DeepLabV3, StereoSGBM, KITTI Dataset

- Developed a camera-only ADAS pipeline for real-time obstacle detection and emergency rerouting using KITTI stereo data
- Used StereoSGBM for depth estimation and fused with YOLOv5 detections for 3D obstacle localization and collision time
- Implemented braking decision logic and A\* on an occupancy grid fused from road segmentation and detected obstacles.

### Stealth Grid World Goal finding using Reinforcement Learning | Python, Gym, PyTorch, Matplotlib, Stable-Baselines3

- Built a multi-agent RL environment where agents locate goals under limited FOV while evading moving guards
- Developed a customizable grid framework and parallel training pipeline with randomization to improve policy robustness
- Increased robustness of learned policies under varying grid sizes, guard behaviors, and agent FOV constraints

### Motion Planning with 6-DOF Manipulator | Python, ROS, RViz

- Implemented Rapidly exploring Random Tree for motion planning of a 6-DOF robotic manipulator using Python and ROS
- Optimized trajectory generation by using goal-biased sampling and smoothing techniques for better execution time
- Analyzed trajectories using RViz, comparing results through video recordings for different parameter configurations