

Abhijit Mahalle

amahalle@terpmail.umd.edu | GitHub: [abhijitmahalle](#) | LinkedIn: [Abhijit Mahalle](#) | Website: [abhijitmahalle](#) | College Park, MD
Domain skills: Robot Perception, Computer-Vision, ROS, Machine Learning, Robotics Software Development, Path-Planning, State-Estimation

EDUCATION

University of Maryland, College Park

Master of Engineering, Robotics

Courses: Perception, Path-Planning, Robotics Software Development, Machine Learning, Deep Learning, Aerial Robotics, Control Systems

May 2023

GPA: 4.0/4.0

University of Mumbai, India

Bachelor of Engineering, Mechanical Engg.

May 2018

GPA: 8.36/10.0

TECHNICAL SKILLS

Languages and Tools: C++, Python, MATLAB, ROS, Gazebo, RViz, Git, Docker

Libraries: OpenCV, PyTorch, TensorFlow, Keras, NumPy, sklearn, pandas, GTest, pytest

Deep Learning: CNN, RNN, LSTM, GAN, Transformers

Controls: MPC, LQG, LQR, PID

Planning: A*, RRT*, Real Time-RRT*, Dijkstra

WORK EXPERIENCE

Maryland Robotics Center | *Researcher*

May 2022 - Present

- Created a ground-truth dataset for **VIO** and **SLAM** models by fusing data-streams from **event** cameras, classical camera, and **IMUs** for ego-motion, depth-estimation, scene-segmentation, and optical-flow.
- Leveraged **Vicon** motion capture system to get the ground-truth pose and simulated the recorded events on **Mujoco** to generate the masked sequences.
- Developed a pipeline that **calibrates** event and monocular camera simultaneously and in **real-time** by converting event-stream to gray-scale images using E2VID deep-learning network.

Worley | *Machine Learning Engineer*

Aug 2020 - July 2021

- Developed a **LSTM**-based machine learning model to predict the temperature of distillation columns for inference control and reduced the energy consumption by **40%**.

Piping Design Engineer

Sept 2018 - July 2020

- Designed piping systems using CAD tools for fluid transfer within the process plants considering chemical process requirements.
-

PROJECTS

Structure from Motion - Reconstructed a **3D scene** and simultaneously obtained **camera poses** from a given set of images using their feature point correspondence (**epipolar geometry**, **triangulation** and **non-linear optimization**). [GitHub](#)

Robot Path-Planning - Implemented **BFS**, **DFS**, **Dijkstra**, **A***, and **Real Time-RRT*** on **holonomic** and **non-holonomic** robots. [GitHub](#)

Home organizing robot - Developed a **ROS** package with GitHub **Continuous Integration** and **GTest** for mobile manipulator for indoor search and object manipulation. Used **MoveBase** for autonomous navigation and **MoveIt** for manipulator control. [GitHub](#)

SfMLearner - Estimated **depth** and **egomotion** from image sequences using unsupervised learning approach and improved the performance by **7%** over baseline. [GitHub](#)

Hand-written digit recognition - Implemented **Linear** and **Kernel SVM**, **Logistic Regression**, and **CNN** along with **transfer learning** on **MNIST** dataset with an improvement in accuracy by over **42%**. [GitHub](#)

Panorama stitching - Stitched images to create a **panorama** using traditional (**Homography** estimation using feature points) and deep-learning (**HomographyNet**: supervised and unsupervised) methods. [GitHub](#)

Face swap - Swapped faces in videos using traditional (**Delaunay Triangulation** and **Thin Plate Spline**) and deep-learning (**Position Map Regression Network**) methods. [GitHub](#)

April Tag detection and tracking - Detected and tracked April Tag by background removal using **FFT** and corner detection. Superimposed a custom image and a virtual cube on it using **projection**, **calibration**, and **homography** matrices. [GitHub](#)

Lane detection and turn prediction - Detected straight and curved lanes mimicking the lane departure warning system using homography, **hough lines**, polynomial **curve fitting** using **sliding window** and predicted turn from the radius of curvature. [GitHub](#)

Human Position Estimator - Developed a software by **Agile Iterative Process** that detects and tracks humans using a pre-trained **HOG** descriptor and **SVM** detector of OpenCV. [GitHub](#)

Sensor fusion using ES-EKF - Fused sensor data from **IMU**, **LIDAR**, **GNSS** using **Error State Extended Kalman Filter** to estimate pose of an autonomous vehicle. [GitHub](#)

Gantry Crane Control - Designed **LQR** and **LQG** control for a gantry crane, analyzed the **controllability** and **observability**, and implemented **Kalman** filter to account for Gaussian noise in sensor measurements. [GitHub](#)