5069 Berwyn Road
College Park MD 20740
https://github.com/nantha007

NANTHA KUMAR SUNDER

(240) 579-8788
nantha07@terpmail.umd.edu

EDUCATION

College Park, MD **University of Maryland** **Aug. 2018 - Present**

- **Major:** Robotics, M.Eng. (GPA: 3.55/4)
- **Minor:** Computer Science
- **Coursework:** Planning, Perception, Control systems, Robot Learning, Machine learning, Computer Vision, Robot Modelling, Software development for robotics.

Coimbatore, India **Amrita University** **Aug. 2011 – May 2015**

- **Major:** Electrical and Electronics Engineering, B.Tech. (GPA: 7.7/10)
- **Minor:** Embedded Systems
- **Coursework:** Control Systems, Embedded Systems, Neural Networks, Electronics, Programming C++.

WORK EXPERIENCE

Software Engineer **KPIT Technologies** **Nov. 2015 – Apr. 2018**

- Composed system requirement for Autonomous Emergency Braking based on Automotive Safety Integrity Level.
- Designed a path-planning algorithm for autonomous parallel car parking.
- Developed a MATLAB program to verify and validate the data generated from the autonomous vehicle model in Simulink.
- Leveraged Knowledge in C++, MATLAB, Simulink, CarMaker.

PROJECTS

Frontier Exploration using ROS and Gazebo

- Used turtlebot to autonomously explore and construct the unknown map using ROS and SLAM.
- Implemented the Breadth-First Search (BFS) to reduce the time taken to completely explore the environment.
- **Utilized:** C++, ROS, Gazebo, SLAM, Google Test framework, Doxygen, Agile Methodology, BFS.

Augmented Reality using OpenCV and Python

- Developed a python program to detect and decode QR code.
- Used OpenCV to project an image and a 3D cube on top of the detected QR code.
- **Utilized:** Python, OpenCV, Homography, Harris corner detector.

Traffic sign recognition using SVM

- Implemented MSER algorithm to detect the traffic signs.
- Improved the classification of the traffic signs by 95% using negative hard mining and Support Vector Machines.
- **Utilized:** Python, OpenCV, HOG, SVM, MSER.

Monocular Depth Prediction on NYU Depth data using PyTorch

- Developed a Convolutional Neural Network based on ResNet-50 to encode the image.
- Used zero-padded Upconvolutional layers and reduced the loss by 60% in the predicted depth image.
- **Utilized:** Python, PyTorch, OpenCV, Convolutional Neural Network, ResNet-50.

Obstacle detection using pcl and LIDAR

- Implemented a C++ program to process LIDAR data from a car in Urban scenarios and detect the obstacle obstacles.
- Used RANSAC to get point clouds of obstacles like cars from the LIDAR data.
- Designed KD-Tree algorithm to separate each obstacle by drawing bounding box.
- **Utilized:** C++, PCL, LIDAR, RANSAC, KD-Tree.

SKILLS

- **Software:** (*proficient*): Python, C++, MATLAB (familiar): Git, C.
- **Tools/Libraries:** Robot Operating System (ROS), TensorFlow, PyTorch, Numpy, Keras, Simulink, OpenCV, pcl.