

## SUMMARY

Automation Engineer with a passion for self-driving cars and a background in computer vision and deep learning.

## SKILLS

- Programming: Python, C++, C, OpenCV, PyTorch, MATLAB & Simulink, Rospy
- Software: PyCharm, Visual Studio, Google Colab, ROS, Unity 3D
- OS: Ubuntu, Windows
- Languages: English (Fluent), Urdu (Native)

## EXPERIENCE

### Computer Vision & Deep Learning Intern

**T3LAB; Bologna, Italy** Oct 2019 - Mar 2020

- Reviewed the accuracy of pretrained 2D object detectors like Faster-RCNN, SSD, & YOLO on video data
- Developed a real-time (30 FPS) application for vehicle and person counting at busy road junctions
- Also did research on my Master Thesis

### AUDI Autonomous Driving Competition Participant

**AUDI; Ingolstadt, Germany** May 2018 - Oct 2018

- Among 16 teams selected from all over Europe
- Led a team of 4 other students in designing an autonomous car algorithm using deep reinforcement learning and computer vision techniques
- Designed a simulator (from scratch) using C++ and Unity where RL agent was trained and tested using Apprenticeship Learning
- Responsible for extraction of lane lines and aruco traffic markers that were fed as input to the RL agent
- Implemented state machine to execute a sequence of maneuvers at road junctions (turn left, turn right etc.)

## EDUCATION

### Master's in Automation Engineering

**University of Bologna; Bologna, Italy** Sep 2017 - Dec 2020

- **Coursework:** Computer Vision, Industrial Robotics, Optimization Models, Probability & Stochastic Processes, Mechatronics Systems, Mechanics, Real-Time Systems
- **Thesis:** Object trajectory prediction from an ego-centric perspective by combining 3D object detector (PointPillars) with depth map (MADNet) and real-time semantic segmentation neural nets

### Bachelor's in Electrical Engineering

**National University of Science & Technology; Islamabad, Pakistan** Sep 2012 - Jun 2016

- **Coursework:** C/ C++ Programming, Control Theory, Electrical & Digital Circuit Analysis, Instrumentation, Embedded Systems
- **Thesis:** Designed a path-following drone and performed analysis under different PID values in the Unity simulation environment where it followed a collection of waypoints in a sequential manner under the influence of varying external disturbances

## PROJECTS

### **Contributor to Open Source Robotics Library**

*Aug 2020 - Sep 2020*

Python Robotics is an open source library for learning about robotics (path planning, mapping, navigation). My contributions: Bug Planning, Flowfield Pathfinding, and Variants of A\*.

### **Long Range IoT Hackathon (Bologna, Italy)**

*May 2018 - May 2018*

Made an application that would use an Arduino to send waste data from ten locations to a central radio network and formulate the optimum garbage collection route using Dijkstra's Algorithm.

### **Visual Inspection of Blade Tools**

*Jan 2019 - Jan 2019*

Developed an application using C++ and OpenCV that combined Canny Edge Detector, Hough Transform, and Harris Corner Detector to detect pointed edges of a blade tool, compute their angle, and also locate imperfections using a custom filtering technique.

### **Autonomous TurtleBot mapping & navigation in ROS**

*Jan 2020 - Feb 2020*

Implemented mapping using a custom bug planning algorithm and navigation with PRM (Probabilistic Roadmap) on a Turtlebot in the ROS simulation environment. Potential Field Algorithm was used for following the planned trajectory.

### **Analysis of Joint-Space Trajectories of PUMA 560 manipulator**

*May 2020 - May 2020*

Studied performance of a PUMA 560 manipulator under PD Control with Gravity Compensation, Inverse Dynamics Control and Robust Sliding-Mode Control. Analysis was done in MATLAB using Peter Corke's Robotics Toolbox.

### **Walking Philosopher**

*Dec 2017 - Dec 2017*

Solved Tanenbaum's Walking Philosophers problem using an application developed in C with POSIX threads. The successful application had minimum starvation, no deadlock, and could work for any number of philosophers.

### **Efficient Charging and Discharging of Plug-in EVs**

*Dec 2018 - Dec 2018*

Applied the dual subgradient algorithm to provide an optimized schedule for the charging and discharging of Plug-in EVs when connected to the grid. The project was done in C++ using the Armadillo library for linear algebra operations and optimization and the MPI library for multithreading.

## BLOG

<https://medium.com/@sarim.mehdi.550>

I write about my academic and hobbyist projects

## AWARDS

### **Unibo Action 2**

**University of Bologna** 2017-05-01

Received a two year study grant for Master's course based on meritorious GRE scores