


SANJAR NORMURADOV

Munich, Germany | +49 151 592 52908 | snnormuradov@gmail.com | [LinkedIn](#) | [Portfolio](#) | [GitHub](#)

WORK EXPERIENCE

Robotics Engineer AI & Perception Team

Agile Robots SE 
July 2024 – Present

Graduate Research Assistant

UW & Amazon Science Hub 

Robotic Manipulation in Densely Packed Containers

Jan 2023 – June (1 y. 6 m.)

- Architecture Development: Investigating performance boosts in Unseen Object Instance Segmentation by integrating the Segment Anything Model.
- Synthetic Data Generation: Utilized Scanned Objects from Google Research and NViSII to enhance synthetic imagery datasets, simulating diverse backgrounds and object arrangements.
- Improved the average precision of the ViT-based STOW framework for Warehouse Picking Robots on the expanded 140K-image dataset: 0.424 to 0.646 overall, and 0.336 to 0.573 for stacked bins.
- Simulation: Enhanced simulation environment precision and robustness by revising URDF/Xacro/XML files and creating a single source for all pod models, thereby expanding the system's versatility.
- Motion Planning: Optimized pod positioning relative to the robot workstation by automating test-run procedures and refining control and MoveIt parameters, such as collision checking frequency and discretization.
- Reachability Test: Reduced UR16e reachability failures from 20/1600 to 1/1600 via modifications above.

Graduate Teaching Assistant

Electrical & Computer Engineering Department UW

"Software Engineering for Embedded Applications"

Dec 2023 – March (4 mo.)

- Revised course materials on C++ fundamentals such as data structures, Abstract Data Types, STL containers, and State Machines using Docker containers, Google Test unit testing environment, and Doxygen API documentation.

"The Self-Driving Car: Intro to AI for Mobile Robots"

Sep 2023 – Dec 2023 (4 mo.)

- Guided students in implementing Autonomous Vehicle stack on NVIDIA Jetson Nano-based MuSHR rally car.
- Prepared course materials on Probabilistic Robotics fundamentals such as Control (PID/MPC), State Estimation (PF/EKF), Path Planning (Dijkstra/PRM/RRT/RRT*/A*), Perception (Pinhole/Stereo/Depth Camera, CNNs).
- Developed a Raspberry Pi based mechatronic gate featuring an ultrasound distance sensor, 16x2 LCD for IP and distance display, a servo, and a buzzer to create a ROS service-managed obstacle for the course's final race.

PROJECT EXPERIENCE

Robotics Software Engineer

Udacity 

Nanodegree

Oct 2023 – Feb 2024 (4 mo.)

- Built a custom world/robot from scratch in a Gazebo, and created ROS nodes/services to program the robot.
- Applied EKF and Adaptive Monte Carlo Localization packages to localize a robot.
- Mapped an environment using the Occupancy Grid Mapping algorithm and RTAB-Map package.
- Simultaneously mapped and localized a robot relative to the map with Grid-based FastSLAM and GraphSLAM.
- Navigated a dynamic environment with obstacle avoidance using SLAM and A* path planning algorithm.

Autonomous Robotics

UW Graduate Studies

"Self-Driving Cars" / "AI – Robotics"

Sep 22 – Dec 22 / Mar 23 – Jun 23 (7 mo.)

Applied novel methods in Autonomous Robotics to varied robotic platforms (MuSHR car, 3-DoF robotic arm)

- Control: Implemented PID and MPC to manage the steering angle and speed of the rally car.
- Localization: Implemented PF/EKF (Odometry/Velocity Motion and Landmark/Beam-based Sensor Models)
- Path Planning: Applied RRT/RRT*/A*/LPA* to navigate in robotic arm C-space, and A* to the rally car.
- Reinforcement Learning: Explored Behavior Cloning, DAgger, and Policy Gradient techniques in MuJoCo.

EDUCATION

University of Washington

Seattle, WA

M.S. in Electrical and Computer Engineering, Robotics (GPA: 3.6)

Oct 2022 – June 2024

- Activities: UW + Amazon Science Hub; Robotic and State Estimation Lab; Sensor System Lab.
- Courses: Self-Driving Cars; AI-Robotics; Deep / Machine Learning; SWE for Embedded Applications.

SKILLS

CV | ML | DL | RL | Motion Planning | Localization | Control

Python | C++, Git | Jira | Docker, ROS | RViz | Gazebo | MoveIt | Pybullet | MuJoCo | PyTorch