Sahasrajit Anantharamakrishnan

anantharamakrishn.sa@northeastern.edu +1 (857) 206-2833 linkedin.com/in/sahas-ananth sahas-ananth.github.io

EDUCATION

Northeastern University, Boston, MA — **May 2024**
Master of Science in Robotics Engineering — **GPA: 3.939/4.000**
Courses: Legged Robotics, Graph Theory, Deep Learning, Autonomous Field Robotics, Mobile Robotics, Computer Vision, Reinforcement Learning & Sequential Decision Making

Anna University, Chennai, India — **May 2022**
Bachelor of Engineering in Electrical and Electronics Engineering — **GPA: 8.66/10.00**

WORK EXPERIENCE

Robotics and Intelligent Vehicles Research Laboratory (RIVeR), Boston, MA — **May 2024 - Present**
Robotics Research Assistant, ***Project:*** *Stochastic Model Predictive Control for bipedal loco-manipulation*

- Introduced probabilistic models into traditional MPC to create Stochastic MPC (SMPC), to improve adaptability and robustness against uneven terrain and unexpected loads.
- Spearheaded the adaptation of the SMPC framework from quadrupedal to bipedal robots to demonstrate its generalizability, using simulation platforms such as PyBullet and Gazebo
- Refined the dynamics model and cost function to satisfy the constraints of the bipedal robot to guarantee stability

Autonomy & Intelligence Laboratory, Boston, MA — **January 2023 - May 2024**
Robotics Research Assistant, ***Project:*** *High-Speed Off-Road Autonomy Robot* — Lab Link

- Developed an innovative 2.5D terrain model accommodating uncertainties in both the shape and properties of challenging off-road environment
- Created and Optimized a custom MPPI algorithm using JAX python, slashing average run time from 1000 ms to 1 ms
- Crafted a custom cost function for MPPI controls, prioritizing speed in unstructured environments while considering the robot's kino-dynamics, terrain traversability, and safety constraints
- Fine-tuned STEGO, a self-supervised semantic segmentation head for DINOv1 vision transformer, on RUGD, RELLIS, and a custom dataset to achieve clear class clusters for RGB image semantic segmentation
- Employed sensor fusion techniques to combine 3D-LiDAR data with semantically segmented RGB images, resulting in a Semantic Point Cloud, essential for downstream perception, control, and motion planning tasks
- Utilized Fusion 360 to engineer and assemble a customized compute and sensor suite payload, designed to meet the distinct needs of AgileX's scout and Clearpath's Warthog robotic platforms, to enable high-speed offroad autonomy capability

Rigbetal Labs LLP, Pune, India — **August 2021 - November 2021**
Robotics Engineer Intern

- Formulated a novel algorithm, Road Anomaly Detection System (RADS), in C++ to detect road anomalies (Potholes, Speed Bumps, etc.) using normal estimation
- Reduced cost by 90%, by generating a 3D Point cloud from a series of moving 2D Laser scans
- Simulated a multi-agent (robot) mapping environment in Gazebo ROS to create a cohesive 2D map
  - Deployed the same in a cloud environment using AWS Robomaker to enable remote multi-user control of an agents

Capgemini Technologies Services, Bangalore, India — **July 2020 - December 2020**
Robotics (Medical Devices) Intern — Project Link

- Fabricated a ROS-based autonomous ground vehicle in Fusion 360 to sterilize and sanitize offices from SARS-COV2 virus with Ultraviolet (UV-C) irradiation
- Directed communication and task distribution between the team and clients, enhancing team efficiency and client relations.

SKILLS

Languages / Libraries Python, PyTorch, JAX, C++, CUDA, C, MATLAB, OpenCV, Tensorflow, PCL
Software and Tools ROS, Ubuntu Linux, Git, CMake, Docker, Gazebo, Nvidia Issac Sim, PyBullet, MQTT, Simulink Fusion 360, Blender, LaTeX

PROJECTS

Implementing Batch Informed Trees (BIT*) Motion planning Algorithm — **March 2023 - April 2023**
Paper: Batch Informed Trees (BIT): Informed asymptotically optimal anytime search* — Project Link

- Reduced the run-time of the algorithm in python using hash-maps, parallelization, and caching
- Engineered intuitive visualization techniques to better analyze the BIT* algorithm
- Tested the algorithm against baselines results such as RRT, RRT*, FMT*, and RRT Connect