Vinay Naidu

Autonomy/GNC Engineer — AUV Navigation (Sensor Fusion • Geodetic/DR • Planning • Reliability)

vinaynaidu13@vt.edu | 240-728-8230 | LinkedIn | Portfolio

Education

- **Virginia Tech** — Blacksburg, VA
  M.S. in Computer Engineering (Thesis: Underwater Robotics & Autonomy) — *Aug 2023 – Dec 2025*
- **Vellore Institute of Technology** — India
  B.Tech in Electronics and Communication Engineering — *Aug 2018 – May 2022*

Experience

- **Virginia Tech — Center for Marine Autonomy and Robotics (CMAR)** — Blacksburg, VA
  Graduate Research Assistant — Autonomy/GNC, Mapping & Multi-Agent Tracking — *Aug 2023 – Present*
  - Engineered ambiguity-aware Bayesian estimation for underwater sensing (EKF + multi-hypothesis / set-based tracking) using hydrophone, sonar, and IMU streams; reduced drift **4**× (2.0m → 0.5m) over 500m trajectories in field-style runs.
  - Implemented distributed information fusion for multistatic AUV networks (contact-sharing + track-sharing) designed for low-bandwidth, failure-prone comms; integrated track-to-track association and sequential disambiguation to suppress ghost tracks under port–starboard ambiguity.
  - Built reliability tooling for field deployment: timing watchdogs, sensor heartbeat checks, measurement gating, and failure-mode tracing (dropouts, bias, delayed packets) to keep autonomy stable under real ocean conditions.
  - Developed data-product workflow from raw logs → QC metrics → operator visualizations (drift plots, track confidence, replay tools), accelerating iteration across multiple open-water trials.
- **Aerospace Engineers Pvt. Ltd.** — Hosur, India
  Robotics Software Developer — Navigation & Control (GNC) — *Mar 2022 – May 2023*
  - Developed end-to-end GNC architecture: coordinate frames (geodetic + local tangent frames), timing model, sensor interfaces, estimation → guidance → control loops, and fail-safe behaviors for constrained underwater operations.
  - Implemented global planning (graph search/costmaps) and reactive local avoidance; reduced planner compute ∼40% via tighter map updates and pruning while maintaining safe navigation at operational speed.
  - Built repeatable sim-to-real validation: containerizable SITL/HITL-style regression scenarios, log replay, and parameter-sweep harnesses to speed controller tuning and prevent regressions.
- **IntRobotics Pvt. Ltd.** — Hyderabad, India
  Perception Engineer — Multi-Sensor Fusion — *Dec 2021 – Mar 2022*
  - Built LiDAR-camera fusion for occupancy-grid mapping: calibration + extrinsic alignment + synchronized processing; achieved **90%** obstacle detection precision at 20 Hz in on-vehicle tests.
  - Created automated evaluation + visualization tools (error metrics, drift plots, scenario playback) to quantify perception changes and accelerate debugging on test vehicles.

Selected Projects

- **Neural A* for ROS2 Nav2 (Planner Plugin + PyTorch Guidance):** Implemented a learned-guidance A* variant and benchmarked against standard A*; achieved ∼11.6× fewer node expansions with a small path-length gap in indoor sim (start/goal held constant).
- **Distributed Information Fusion for Multi-AUV Tracking (C++/Python):** Implemented contact-sharing + track-sharing fusion with association and ambiguity management for bandwidth-limited comms; designed for real-time deployment and log-replay validation.
- **Factor-Graph Trajectory Optimization (GTSAM/C++17):** Built a pose-graph optimizer with Levenberg–Marquardt refinement over navigation/landmark constraints to reduce drift and improve global consistency.

Technical Skills

Languages: C++ (14/17/20, STL, templates, multithreading), Python, Bash, CMake
Robotics: ROS/ROS2, Nav2, TF/TF2, rviz, rosbag2, Gazebo, URDF, lifecycle nodes
Navigation/Estimation: EKF/UKF, particle methods, multi-hypothesis tracking, factor graphs (GTSAM), Bayesian inference
Planning/Control: A*/D*/graph search, RRT/RRT*, trajectory optimization, PID, basic MPC familiarity
Perception: Point clouds, occupancy grids, calibration/extrinsics, real-time sensor pipelines, sonar/hydroacoustic processing
Systems/Infrastructure: Embedded Linux (Jetson-class), profiling/debugging (gdb), logging/telemetry, Docker (sim/replay), unit tests (pytest/gtest), CI (GitHub Actions), TCP/IP fundamentals