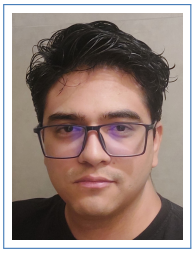
Gabriel F P Araújo

Autonomous System Engineer

Sønderborg, DK
+45 71 43 91 17
+55 (61) 982 308 980
gabriel.fp.araujo@gmail.com
gastd
Gastd
gastd
5561982308980



Nothing in life is to be feared, it is only to be understood
Marie Curie

Skills

Expert skills

Hardware	Robotic Sensors (Perception, Localization), Sensor Integration, Hardware Debugging
Algorithms	Autonomous System Testing, Kalman Filters, Vehicle Control, Obstacle Avoidance, Path Planning
Systems	Development Models, Field Experiments, Control Systems, Robot Simulation

Development

Languages	C, C++, Python, JavaScript	Frameworks	ROS1, ROS2, Autoware
DevOps	Git, Unit Testing, CI/CD	Tools	CMake, GDB, Jenkins
Libraries	ROS Navigation, BehaviorTree.CPP, PyTrees, PCL, OpenCV, Eigen, FreeRTOS		
Applications	MATLAB, SolidWorks, Fusion 360, Blender, Eagle, Inkscape, Altium		

Education

2022 – 2024 **MSc in Informatics**, *University of Brasília*, Brasília, Brazil
Coursework: Digital Signal Processing, Artificial Intelligence, Software Engineering

2012 – 2021 **BSc in Mechatronics Engineering**, *University of Brasília*, Brasília, Brazil
Coursework: Control Theory, Robot Control, Introduction to CS for Robotics, Real-time Processing

Experience

April 2023 – Present **Autonomous System Engineer**, *Danfoss Power Solutions*, Nordborg, DK, Full-time

Outline Working as a System Engineer for a world-class mobile hydraulic and electrification product and solution provider. Based on its headquarters in Denmark. Responsible for writing requirements and testing the autonomous software blocks. Designing system architecture. Also, responsible for R&D activities. Robotic maintenance.

Key Achievements
- Requirements analysis
- Testing
- R&D

April 2022 – March 2023 **Robotics Engineer (remote)**, *Smoob BV*, Leeuwarden, NL, Full-time

Outline Working as the leading robotics engineer for a self-driving car start-up in the Netherlands. Responsible for leading the development of a software stack for an Autonomous Vehicle. I was in charge of planning and prototyping possible solutions for the final product. Besides that, I also handled communication with stakeholders and requirement analysis.

Key Achievements
- Development and integration of Perception and Planning modules using Autoware and ROS.
- Developing proprietary Simultaneous Localization and Mapping (SLAM) algorithm.
- Selecting and designing algorithms and tools for self-driving cars.

Jan. 2022 – March 2022 **Robotics Consultant (remote)**, *Smoob BV*, Leeuwarden, NL, Contract

Outline Worked as a Consultant for an early-stage start-up based in the Netherlands. Responsible for setting up demonstrations for investors and solving technical issues.

Key Achievements
- Remotely checks and sensor set up for a self-driving car prototype.
- Set up Autoware.AI stack to run demonstrations for stakeholders.
- Solving errors and installation issues regarding ROS and other libraries.
- Advisement about technical solutions.

Dec. 2021 – March 2022 **Jr Robotics Software Engineer**, *Automni*, São Paulo, BR, Full-time

Outline Worked on a leading Brazilian start-up in the logistics business. I deployed robots in a big distribution center for a multinational customer in Brazil. Promoted from intern to full-time employee. Execution of tests and validation of robots and troubleshooting on the customer's site.

Key Achievements
- Development and integration of object detection module using 2D Laser and PCL.
- Robot calibration, configuration, testing, and validation on-site.
- Log issues and faults during test and validation steps and documentation of testing results.
- Suggestion of corrections and bug fixes to both hardware and software.
- Hardware maintenance of the deployed robots.

May 2020 – Dec. 2021 **Robotics Developer (*remote*)**, *Automni*, São Paulo, BR, Internship

Outline Worked as a robotics intern at a leading Brazilian start-up in logistics. Responsible for maintaining and documenting the code base and helping the engineering team add new project features.

Key Achievements
- Development of the order management system for multi-robots.
- Addition of new behaviors into the robot's Behavior Tree using BehaviorTree.CPP library.
- Integration of the client's system to send orders to the robots.
- Development of User Interface in ReactJS for interacting with the robots using Roslibjs.
- Feature addition and bug fixing in the company's multi-robot route planner.
- Integration of 2D Laser security areas into ROS using OpenCV.
- Documentation and testing.

April 2019 – June 2019 **Embedded Developer**, *LandSense Soluções Tecnológicas*, Brasília, BR, Internship

Outline Worked on developing a new product in a Brazilian start-up as an undergraduate student. Responsible for prototyping the new product and its validation.

Key Achievements
- Design and implementation of a Bluetooth mesh protocol using Bluetooth® 4.0.
- Field testing for the mesh communication.
- Embedded software development using the Nordic nRF52832 microcontroller and RTOS.
- Main technology: C/C++.

July 2015 – April 2019 **Undergraduate Researcher**, *LARA (Automation and Robotics Laboratory)*, Brasília, BR

Outline Worked in a robotics research laboratory at the University of Brasília as an undergraduate student. Responsible for maintaining the robots and helping the researchers to do experiments.

Key Achievements

- Implement an indoor localization system using Cameras, Augmented Reality (AR) markers, OpenCV, and EKF.
- Firmware development for an Inertial Navigation System (INS) using STM32F103C8 development board.
- Development of an environment explorer using SMACH ROS and ROS Navigation stack.
- Implementation of ROS drivers for GPS and IMU sensors.
- Maintenance and documentation of ROS packages and robot hardware.
- Configuring and setting up robots for experiments.
- Creation of a chatbot system for controlling a mobile robot using speech recognition and state machine.
- Also engaged in other research projects in robotics, specifically on multi-robot control and navigation.

Projects

Jan. 2021 – Oct. 2021 **MissionControl: An architecture for mission coordination of heterogeneous robots**
The work is divided into three parts: (i) Implementation of a simulation environment for multi-robot systems using *Blender*, *Fusion360*, *ROS Navigation* and Morse simulator in *Python* and *ROS1*. (ii) Implementation of a robotics action sequencer using *Behavior Trees* in *Python* and *ROS2*. (iii) Design and implement a test orchestration architecture based on *Docker* to run several experiments autonomously. Paper published in a major journal, online available at 10.1016/j.jss.2022.111363.

March 2019 – June 2019 **Autonomous Navigation System for a Car-like Robot**
Development of a navigation architecture for a car-like robot with four steps: (i) *Probabilistic roadmap (PRM)* for path planning, (ii) a naive discretization as trajectory generation algorithm, (iii) non-linear automatic controller based on *Lyapunov theory* for path following, and (iv) *PID* for motor control. Navigation system implemented in *C++14* in *ROS* and was deployed in a simulated robot in *Gazebo* simulator for testing.

May 2017 – Aug. 2017 **Expansion of the GNSS-SDR software to GLONASS system**, *Google Summer of Code 2017 participant with GNSS-SDR*
Implementation of Acquisition and Tracking blocks in a GPS receiver for the GLONASS satellites. The Acquisition aims to find all the satellites visible to the antenna and discover which Doppler frequency they are. The Tracking tracks the satellites' signal so the navigation data can be retrieved from them. Addition of unit and system tests of the developed code. Further details in the report, online available at *click here*. Project implemented in *C++14*. Inside GNSS magazine published an article about the work, available at *click here*.

July 2014 – June 2015 **Need4SS: Autonomous Driver using Evolutionary Algorithm**
Development of an autonomous driver for the TORCS – an open-source 3D car racing game – to compete in the Simulated Car Racing Championship (SCRC), a former GECCO Competition. The implementation used *Finite-State Machine* in *C++11*. Later, we evolved the state machine parameters using a *Genetic Algorithm* to get the best driver possible with our approach. The driver got fifth place in the SCRC 2015. We published and presented the development and results as a conference paper, online available at 10.1109/SBGames.2015.19.