Robotics Academy: migration of several exercises from ROS1 to ROS2 and refinement

Google Summer of Code

JdeRobot

GSoC ‘22 Proposal

Pratik Mishra

Table of Contents

---

1. About Me

1.1 Basic Information

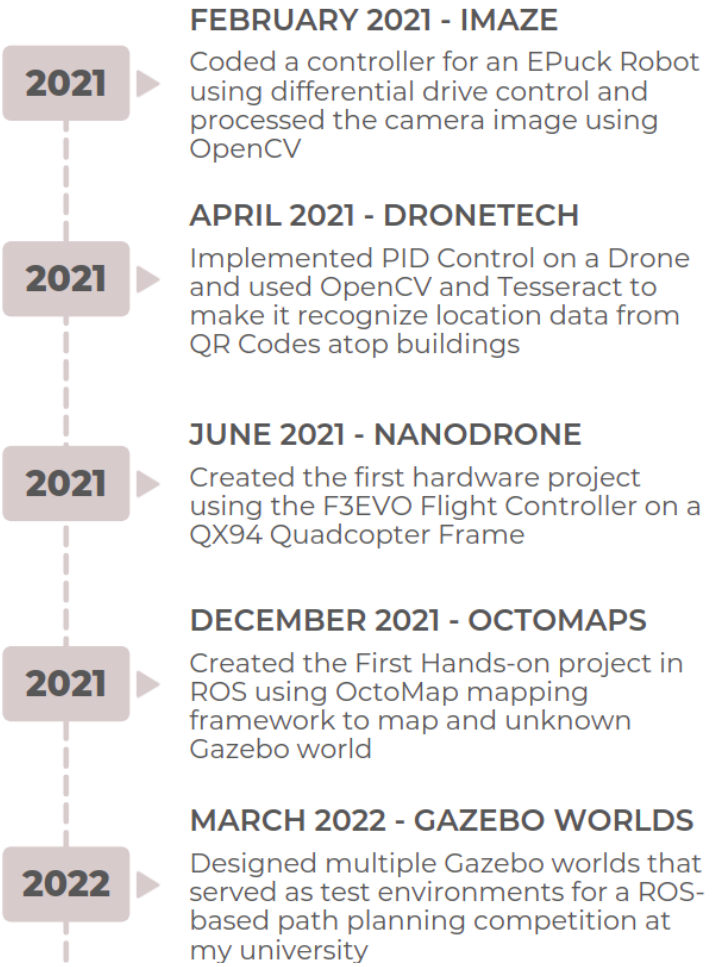
I, Pratik Mishra, am a sophomore in Electronics Engineering from the Indian Institute of Technology (BHU), Varanasi. My contact details, resume and field of study and more are as follows:

Name	Pratik Mishra
University	Indian Institute of Technology (BHU), Varanasi
Field of Study	Electronics Engineering
Degree	Bachelor of Technology
Expected	July 2024
Year	Sophomore
Timezone	Indian Standard Time (GMT +5:30)
Email	pratikmsr@outlook.com
Resume	Pratik Mishra Resume
GitHub	DolceParadise
LinkedIn	Pratik Mishra LinkedIn

Correspondence Address	C-301, Marigold -2, Beverly Park, Mira Road (E), Thane - 401107
Phone	+919969689907
Discord Username	DolceParadise#1491

1.2 Why Robotics?

I have been interested in Robotics and autonomous systems since my freshman year. My first encounter with Robotics was with iMaze, a Robotics event oriented for freshers at my university. I was fascinated by how close we are to automated vehicles and more such intelligent systems. As soon as I participated in this event, I was hooked on the field of Robotics. I have described my journey so far in the below graphic.



1.3 Why JDERobot?

Robotics, especially in my country, is relatively a niche field of study. There aren't a lot of courses offered that teach Robotics or ROS to students. The tutorials offered at ROS Wiki were bland. I couldn't realize the immense potential of ROS while reading and doing the tasks in these tutorials. For a more thorough experience, I relied on the ETH Zurich Course - Programming for Robotics for a complete learning experience in ROS. As someone who has been frustrated with multiple setups and failures in my quest to increase my knowledge in Robotics, I found JDERobot to be a brilliant tool for teaching Robotics. The Dockerized containers combined with web templates run very smoothly and offer cross-platform functionality, sparing the learner from the pain of multiple setups. The instructions offered are lucid and the exercises are relevant which gives the learner a gradual guide into Robotics and its applications. Being impressed by this application, I wish to contribute to RoboticsAcademy and JDERobot to be a part of a community that is doing good for the field of Robotics.

1.4 Completed Pre-Proposal Tasks

1.4.1 Robotics Academy Challenge

- Task: To Install JDE Robot and solve its exercise. A more detailed explanation of the task can be found here
- Expected: A video simulation launching the chosen Academy Exercise.
- I chose the Visual Follow Line Exercise. The exercise focused on detecting the line from the image input and making the

robot model follow the given line. My solution to this exercise can be found here and the video demonstration can be found here

1.4.2 C++ Challenge

- Task: To make an executable C++ application that serves the purpose explained in the link
- Expected: An application that should compile with the gcc compiler > 4.8
- The code and the instructions to run the executable application can be found here

1.4.3 Python Challenge

- Task: To simulate Brownian motion in a circular beginning at the center of the visualized screen.
- Expected: A python module with the necessary simulation steps
- The python module that runs the simulation is present here and the video demonstration can be found at this link

1.4.4 ROS2 Challenge

- Part1: Introduction to ROS2
    - 'Hello! ROS2 is fun' challenge wants us to create a ROS2 workspace containing publisher and subscriber nodes publishing the desired message. The code of the workspace and the simulation video can be found in code and video
    - The second part of this challenge focused on installing a robot of my choice and visualizing its laser data in Rviz2.

I choose TurtleBot3 for this task. The video demonstrating laser data visualization can be found here

- Part2: Navigating TurtleBot
  - This task focused on performing at least 3 waypoint navigations with the TurtleBot. The video demonstration for the waypoint navigation can be found here

---

2. Project Description

2.1 Synopsis

Currently most RoboticsAcademy exercises are based on ROS1 Noetic and Gazebo 11. There are also several prototypes of ROS2 Foxy based exercises that require refinement. The main goal of this project is to migrate several RADI-3 exercises to RADI-4, updating the models of the robots involved in those exercises to their homologous model in ROS2. This will require understanding the complete infrastructure and modifying exercises to use ROS2 communications. In addition, the support for several ROS tools (such as rqt_graph and Rviz) from the corresponding exercise webpages should be implemented (using VNC mainly). New exercises integrating the ROS2 Navigation stack are also welcome, which involve the use of functionalities such as collision avoidance, global path planning, and Multi-robot coordination.

ROS2 has put forward several improvements over ROS with changes in middleware and software architecture in many aspects. In this project, we would focus on developing new exercises with ROS2. For more information on ROS2 based exercises, have a look at GSoC 2021 project and corresponding academy exercises 1 and 2. In addition to porting exercises, contributors are also welcome to suggest improvements to the current RADI framework.

- Skills required/preferred: C++, Python programming skills, experience with ROS. Good to know: ROS2
- Difficulty rating: hard
- Expected results: Migrating the current web template exercises from (ROS1 based) RADI-3 to (ROS2 based) RADI-4
- Expected size: 350 hours
- Mentors: Siddharth Saha (sahasiddharth611@gmail.com) and Shreyas Gokhale (shreyas6gokhale@gmail.com).

2.2 Current Codebase

The Robotics Academy Docker Image (R.A.D.I 4.0) for ROS2 Foxy was built in last years' Google Summer of Code. The current branch of the RADI 4.0 for Foxy includes the following exercises:

- amazon_warehouse
- amazon_single
- follow_line

More exercises can be ported to Foxy and RADI 4.0 to utilise the ROS2 stack. I have chosen the following exercise and candidates for the migration to ROS2.

- Obstacle Avoidance

- Drone Cat and Mouse
- Follow TurtleBot
- Vacuum Cleaner
- Localized Vacuum Cleaner
- Optical Flow Teleop

The scope of additions of new exercises along with changes in the codebase for better implementation will be explored.

2.3 Timeline

I have proposed a rough timeline for the execution of my project in this section taking hints from the previous years' GSoC timeline.

2.3.1 20th April - 20th May: Pre-GSoC period

- Go through Docker documentation and understand its usage and implementation for R.A.D.I
- Set up and Execute the previous exercises, viz. Amazon Warehouse in ROS2 environment.
- Extensively read the Final Report and find valuable insights.

2.3.2 20th May - 12th June: Community Bonding Period

- Communicate with the mentors and discuss the findings of the Pre-GSoC period
- Understand the working of the Web Template architecture's backend

2.3.3 13th June - 4th July: Week 1-3

- Port Obstacle Avoidance and Optical Flow Teleop Exercises to RADI 4.
- Run the program and resolve any bugs.
- Create a RADI web template for each of these exercises.

2.3.6 4th July - 25th July: Week 4 - 6

- Port Drone Cat and Mouse and Follow Turtlebot Exercises to RADI 4.
- Run the program and resolve any bugs.
- Create a RADI web template for each of these exercises.

2.3.9 26th July - 15th August: Week 7-9

- Port Vacuum Cleaner and Localized Vacuum Cleaner Exercises to RADI 4.
- Run the program and resolve any bugs.
- Create a RADI web template for each of these exercises.

2.3.11 15th August - 5th November: Week 10-12

- Run and debug the exercises.
- Develop reference solutions for the same.

2.4 Do you understand this is a serious commitment, equivalent to a full-time paid summer internship or summer job?

I promise to remain committed during my job's orientation phase to any remaining tasks in my project. I recognize that this is a serious commitment, equivalent to a full-time paid summer internship or summer job. I have no other commitments during the summer.

2.5 Do you have any known time conflicts during the official coding period?

My summer vacations begin on 12th May and end on 20th July. During this period, I will be able to work for up to 60 hours each week. However, I'll be on a break for two days in July when my college reopens and post reopening, I will be able to give 40 hours each week. The drop in 20 hours will be due to regular college lectures and errands.

---

3. Studies

3.1 What is your School and degree?

I study Electronics Engineering at the Indian Institute of Technology (BHU), Varanasi.

3.2 Would your application contribute to your ongoing studies/degree? If so, how?

My summer vacations begin on 12th May and end on 20th July. During this period, I will be able to work for upto 60 hours each week. However, I'll be on a break for two days in July when my college reopens and post reopening, I will be able to give 40 hours each week. The drop in 20 hours will be due to regular college lectures and errands.

---

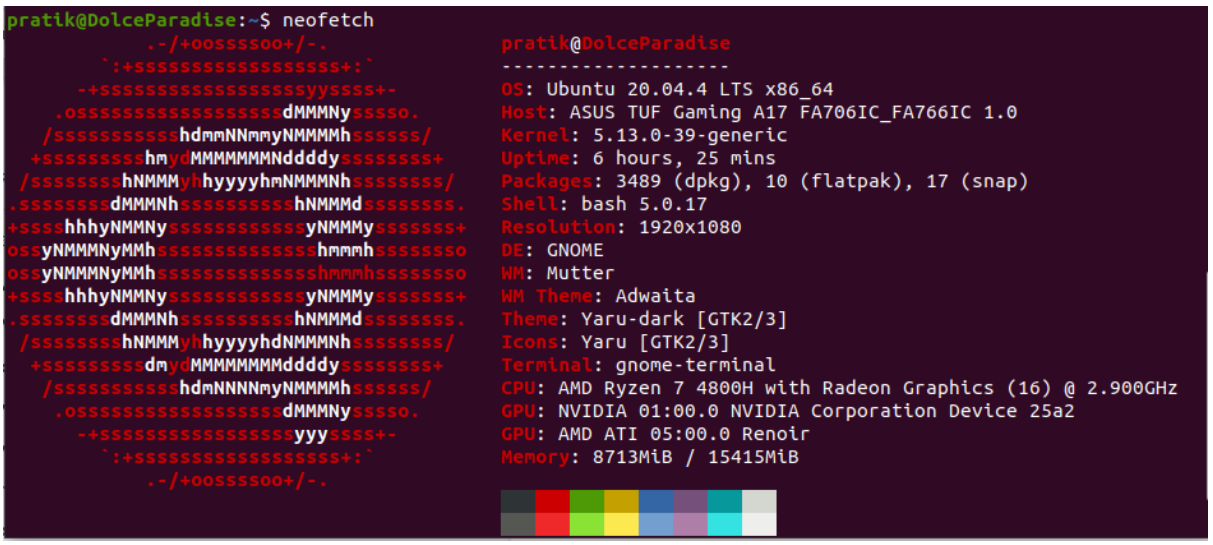
4. Programming Background

4.1 Computing Experience

4.1.1 Operating Systems and Hardware

The description of my daily working setup is as follows

- Hardware: ASUS TUF A17 with 8 core 16 thread Ryzen 7 4800H processor, 16 GB DDR4-3200MHz RAM, 4GB Nvidia RTX 3050 Graphics with CUDA Enabled
- OS: Ubuntu 20.04 Focal Fossa
- ROS: ROS1 Noetic and ROS2 Galactic, both working fine when sourced separately



4.1.2 Relevant Programming Languages and Tools

- C++: Very proficient
- Python: Very proficient
- Experience with ROS1: Worked on Projects in both ROS1 Melodic and Noetic; very proficient
- ROS2: Completed basic tutorials; mid-level proficiency
- Git: Very proficient

4.2 Robot Programming Experience

I previously worked with ROS using Octomaps which mapped any Gazebo environment using /camera/depth_points topic from the Hector Quadrotor drone as a part of the RoboReg, the Robotics Research Group at my university - Indian Institute of Technology (IIT - BHU). The entire documented project along with a working demo video can be found here.

---

5. GSoC Participation

5.1 Have you participated in GSoC before?

No. This is my first time applying to the Google Summer of Code program.

5.2 Have you submitted/will you submit another proposal for GSoC 2022 to a different org?

Yes, I have submitted another proposal to MoveIt in the project "Python bindings for MoveIt2" for GSoC 2022.

---

6. Post-GSoC

6.1 What will I gain from GSoC?

Since I am only a sophomore, I don't have any external commitment and have the utmost yearning for learning new things.

This opportunity will not only get me a sneak peek into the world of robotics but also introduce me to mentors and people who are proficient and accomplished in this field. I hope to share insights with them and gain from their experiences. This will further me into the world of open-source robotics.

6.2 What after GSoC?

I hope to keep using and contributing to JDERobot after the completion of GSoC. There are several exercises yet to be ported to RADI 4.0 that need to be. I'll keep continuing to update the JDERobot's Robotics Academy infrastructure.