Navigation

Haokun Feng, Mingheng Wu, Hengjun Zhang, Xinyi Zhou,
Kate Chen, Matthew Zhang

Link

Navigation Goal

Senario:

Accept commands from the Inoribit to autonomously transport items grabbed by the manipulation team to a designated location, and wait for the receiver to take the items.

Goal 1:

Achieve accurate GIX mapping through lidar scanning basing on the robot movement.

Goal 2:

Enable the robot to perform smooth navigation and cargo transportation tasks.

Challenge

Hardware

Have never operated or programmed a create robot before, and there is relatively limited information available online.

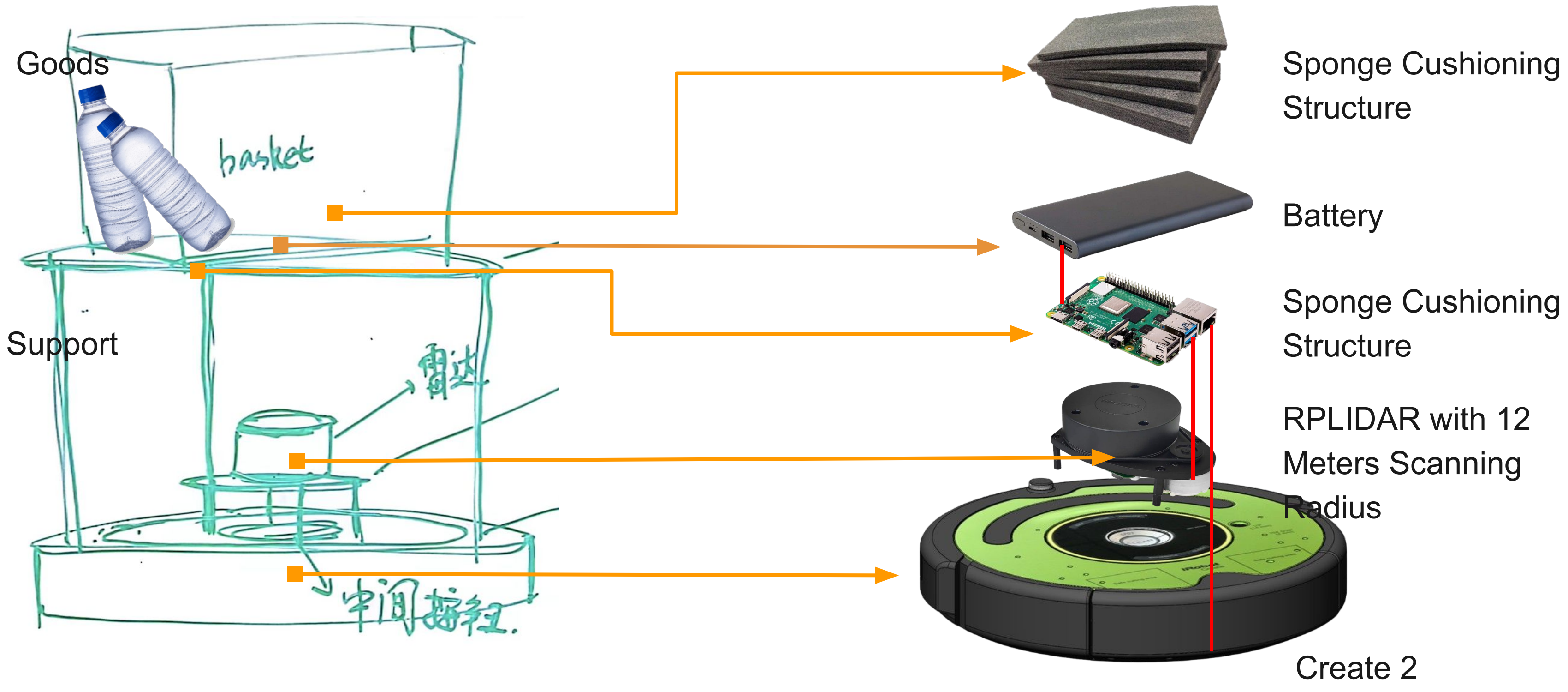
Precise Mapping

Precision control of lidar for creating mapping, also, The robot and lidar need to operate at low altitudes, navigating a relatively complex environment.

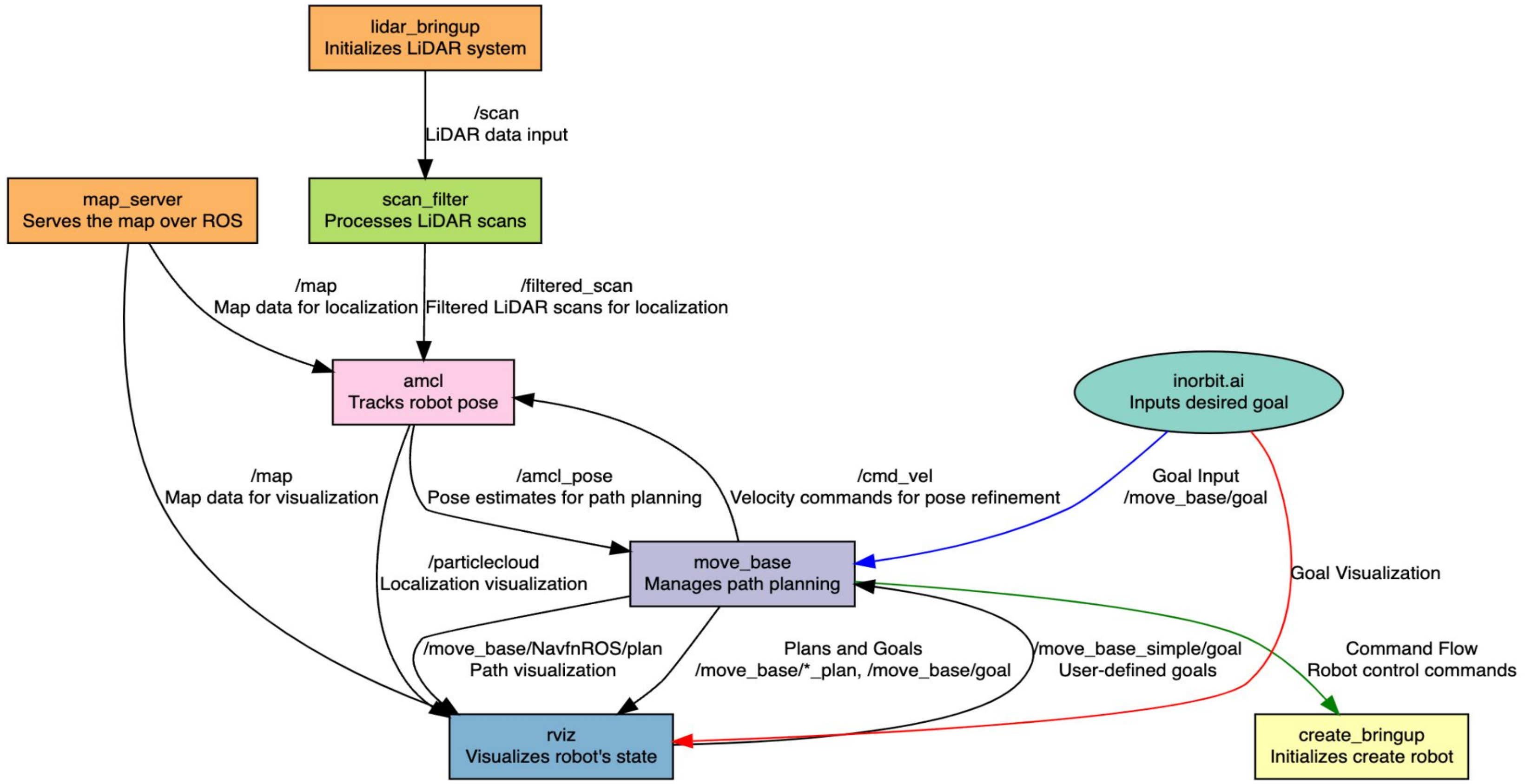
Parameters Tuning

Navigation parameters such as costmap Amcl and planner have a significant impact on the navigation performance of robots, but there is no clear guideline to determine which parameter is effective. Therefore, the impact of different parameters can only be measured repeatedly in experiments. Ultimately determine the parameter with the highest success rate.

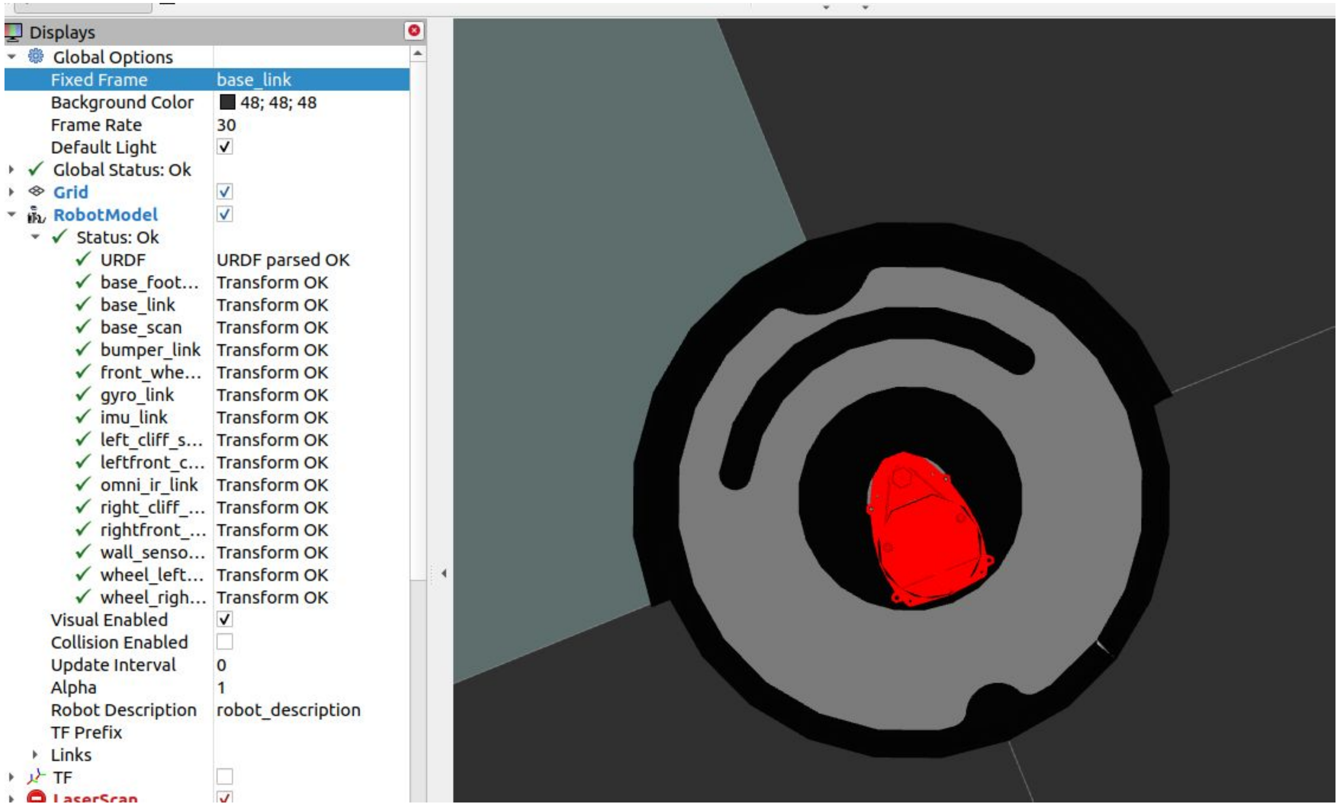
HW & Product Design


Goods
basket
Support
雷达
中间[illegible]
Sponge Cushioning Structure
Battery
Sponge Cushioning Structure
RPLIDAR with 12 Meters Scanning Radius
Create 2

SW Architecture

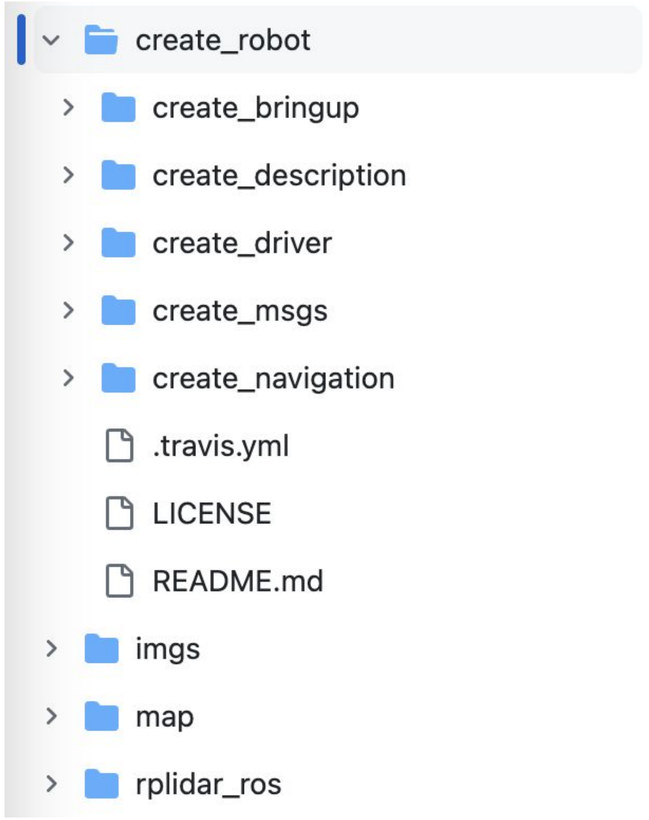


SW Architecture: Physical visualization

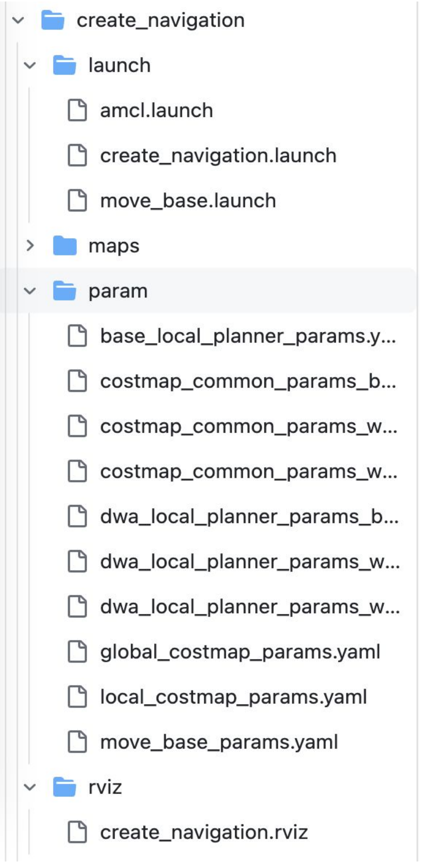


Displays	
Global Options	
Fixed Frame	base_link
Background Color	■ 48; 48; 48
Frame Rate	30
Default Light	✓
✓ Global Status: Ok	
Grid	✓
RobotModel	✓
✓ Status: Ok	
✓ URDF	URDF parsed OK
✓ base_foot...	Transform OK
✓ base_link	Transform OK
✓ base_scan	Transform OK
✓ bumper_link	Transform OK
✓ front_whe...	Transform OK
✓ gyro_link	Transform OK
✓ imu_link	Transform OK
✓ left_cliff_s...	Transform OK
✓ leftfront_c...	Transform OK
✓ omni_ir_link	Transform OK
✓ right_cliff_...	Transform OK
✓ rightfront_...	Transform OK
✓ wall_senso...	Transform OK
✓ wheel_left...	Transform OK
✓ wheel_righ...	Transform OK
Visual Enabled	✓
Collision Enabled	☐
Update Interval	0
Alpha	1
Robot Description	robot_description
TF Prefix	
Links	
TF	☐
LaserScan	✓

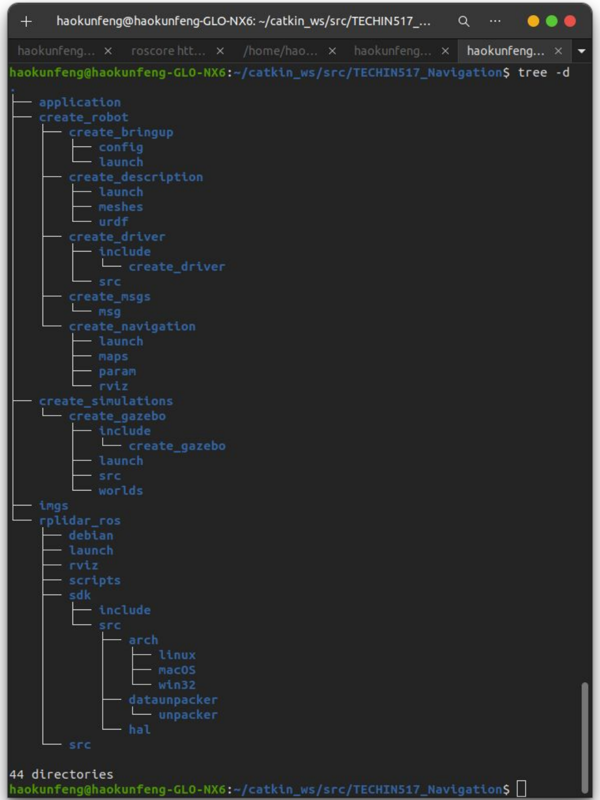
SW Architecture: Implement package for Create2 and Lidar for navigation



```
create_robot
  create_bringup
  create_description
  create_driver
  create_msgs
  create_navigation
  .travis.yml
  LICENSE
  README.md
imgs
map
rplidar_ros
```



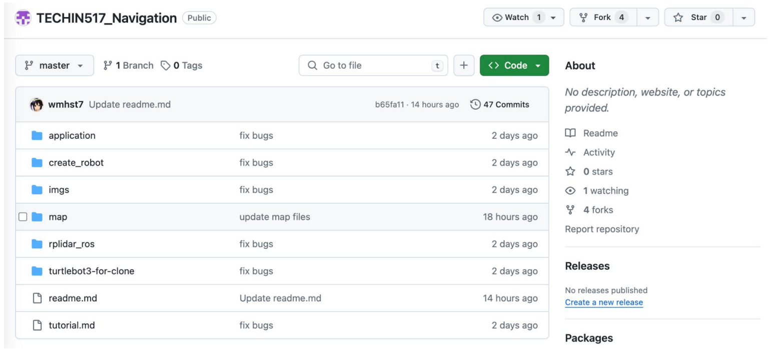
```
create_navigation
  launch
    amcl.launch
    create_navigation.launch
    move_base.launch
  maps
  param
    base_local_planner_params.y...
    costmap_common_params_b...
    costmap_common_params_w...
    costmap_common_params_w...
    dwa_local_planner_params_b...
    dwa_local_planner_params_w...
    dwa_local_planner_params_w...
    global_costmap_params.yaml
    local_costmap_params.yaml
    move_base_params.yaml
  rviz
    create_navigation.rviz
```



```
haokunfeng@haokunfeng-GLO-NX6:~/catkin_ws/src/TECHIN517_Navigation$ tree -d
.
├── application
├── create_robot
│   ├── create_bringup
│   │   ├── config
│   │   └── launch
│   ├── create_description
│   │   ├── launch
│   │   ├── meshes
│   │   └── urdf
│   ├── create_driver
│   │   ├── include
│   │   │   └── create_driver
│   │   └── src
│   ├── create_msgs
│   │   └── msg
│   └── create_navigation
│       ├── launch
│       ├── maps
│       ├── param
│       └── rviz
├── create_simulations
│   └── create_gazebo
│       ├── include
│       │   └── create_gazebo
│       ├── launch
│       ├── src
│       └── worlds
├── imgs
└── rplidar_ros
    ├── debian
    ├── launch
    ├── rviz
    ├── scripts
    ├── sdk
    │   ├── include
    │   └── src
    │       ├── arch
    │       │   ├── linux
    │       │   ├── macOS
    │       │   └── win32
    │       ├── dataunpacker
    │       │   └── unpacker
    │       └── hal
    └── src

44 directories
haokunfeng@haokunfeng-GLO-NX6:~/catkin_ws/src/TECHIN517_Navigation$
```

SW Architecture: Open source package for other users



TECHIN517_Navigation Public

Watch 1 · Fork 4 · Star 0

master · 1 Branch · 0 Tags · Go to file · Code

wmhst7 Update readme.md · b65fa11 · 14 hours ago · 47 Commits

application	fix bugs	2 days ago
create_robot	fix bugs	2 days ago
imgs	fix bugs	2 days ago
map	update map files	18 hours ago
rplidar_ros	fix bugs	2 days ago
turtlebot3-for-clone	fix bugs	2 days ago
readme.md	Update readme.md	14 hours ago
tutorial.md	fix bugs	2 days ago

About

No description, website, or topics provided.

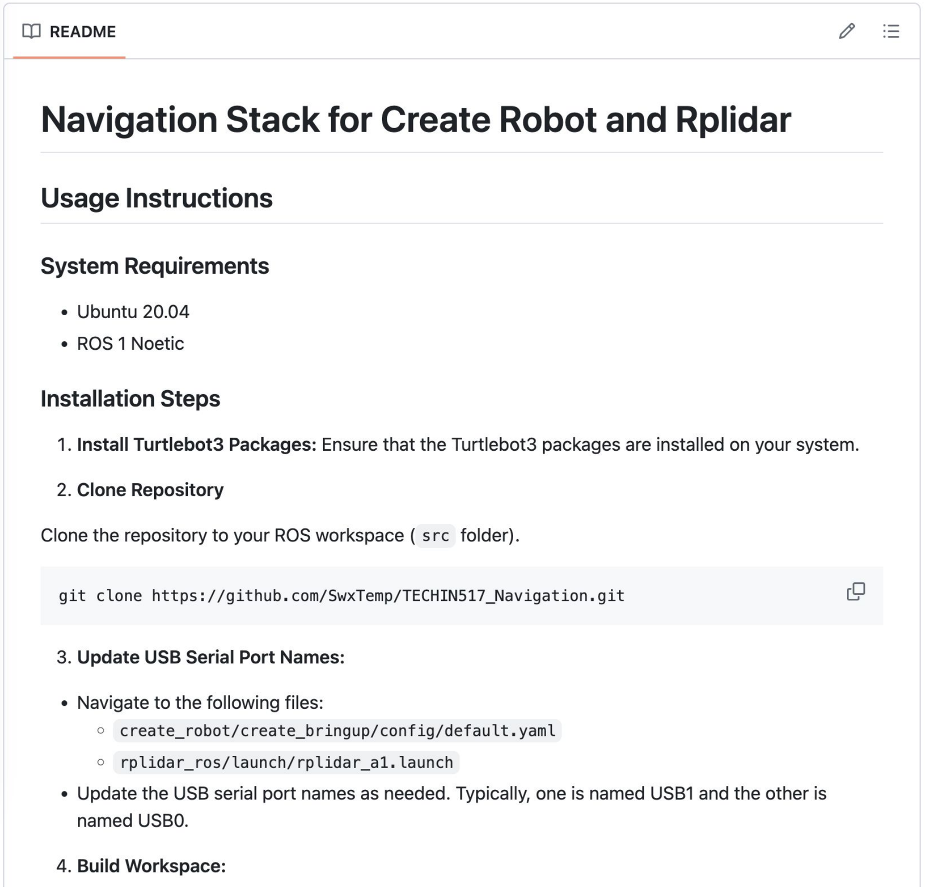
Readme
Activity
0 stars
1 watching
4 forks
Report repository

Releases

No releases published
Create a new release

Packages

https://github.com/SwxTemp/TECHIN517_Navigation/tree/gixlaptop



README

Navigation Stack for Create Robot and Rplidar

Usage Instructions

System Requirements

- Ubuntu 20.04
- ROS 1 Noetic

Installation Steps

1. **Install Turtlebot3 Packages:** Ensure that the Turtlebot3 packages are installed on your system.
2. **Clone Repository**

Clone the repository to your ROS workspace (`src` folder).

```
git clone https://github.com/SwxTemp/TECHIN517_Navigation.git
```

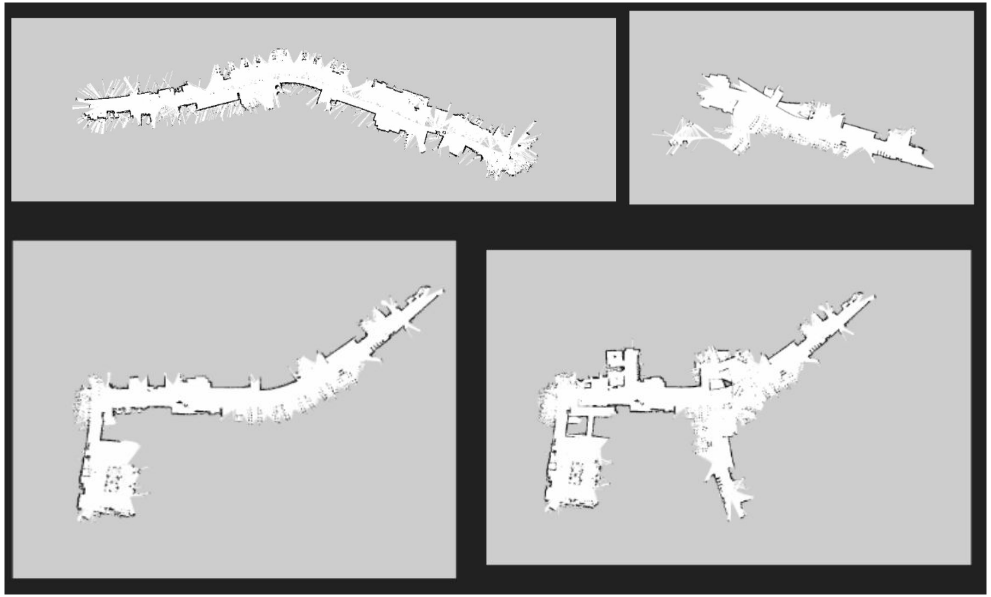
3. **Update USB Serial Port Names:**

- Navigate to the following files:
  - `create_robot/create_bringup/config/default.yaml`
  - `rplidar_ros/launch/rplidar_a1.launch`
- Update the USB serial port names as needed. Typically, one is named USB1 and the other is named USB0.

4. **Build Workspace:**

Map Stage I

Initial Mapping with Gmapping



Method:

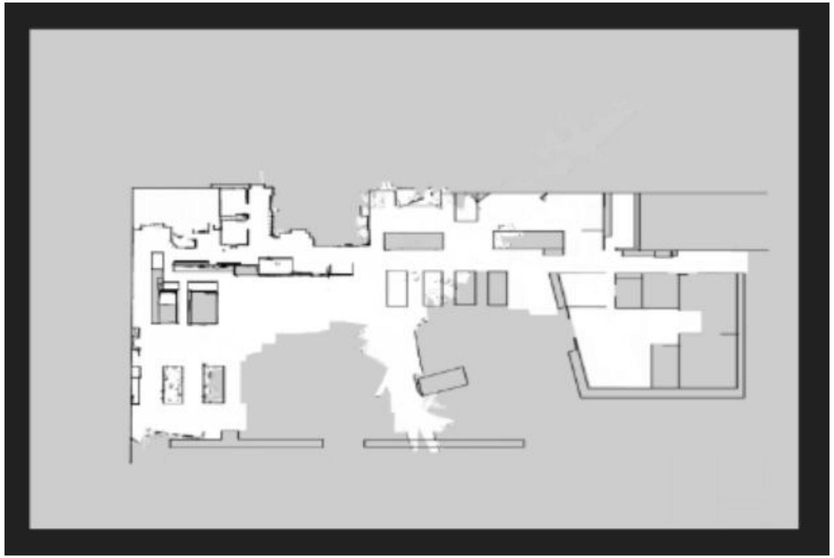
- Utilizes gmapping primarily relying on odometry data for Simultaneous Localization and Mapping (SLAM).
- Employs LIDAR scans for initial map creation.

Challenges:

- Inherent hardware inaccuracies lead to cumulative errors.
- Resulting maps may suffer from distortions due to error accumulation.

Map Stage II

Small-Scale Mapping and Manual Stitching



Method:

- Maps are drawn on a smaller scale using LIDAR, followed by manual stitching of these segments.

Outcomes:

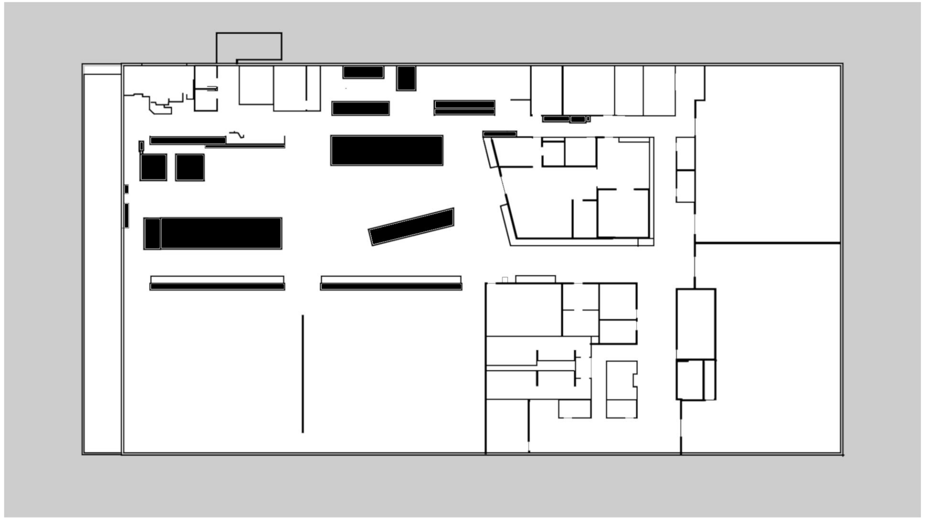
- Reduced overall distortion, bringing the map closer to real-world accuracy.

Challenges:

- Low efficiency in the mapping process.
- Proportional errors can occur during the stitching of map segments.

Map Stage III

Utilizing Construction Blueprints for Reference



Method:

- Maps are refined by comparing and integrating them with construction blueprints.

Outcomes:

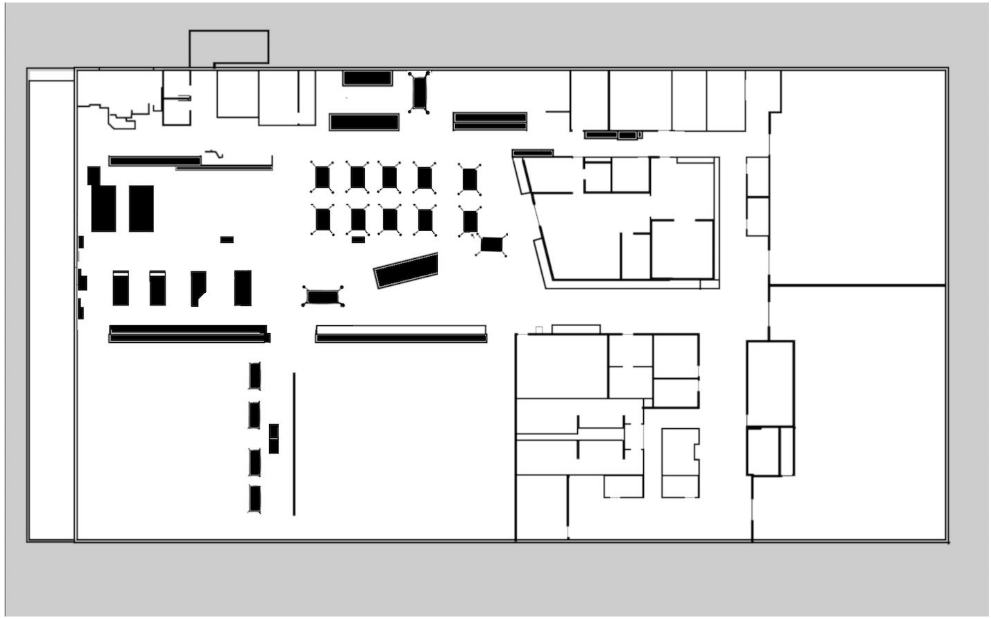
- Significantly improved overall accuracy, especially in the positioning and relationships between walls.

Challenges:

- Inaccuracies in depicting the relationship between movable objects and fixed structures.

Map Stage IV

Combining Methods for Enhanced Accuracy



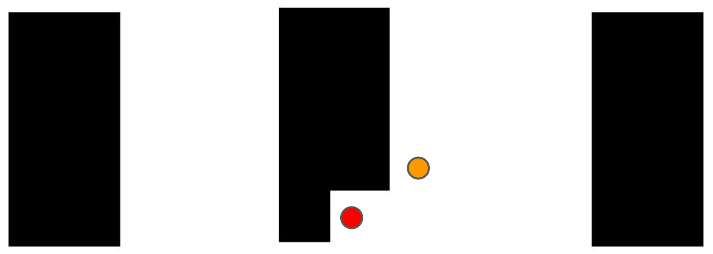
Method:

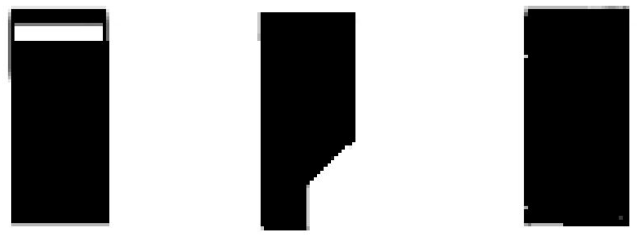
- Integrates LIDAR scanning with manual measurements to optimize the positioning and dimensions of movable objects like furniture.

Outcomes:

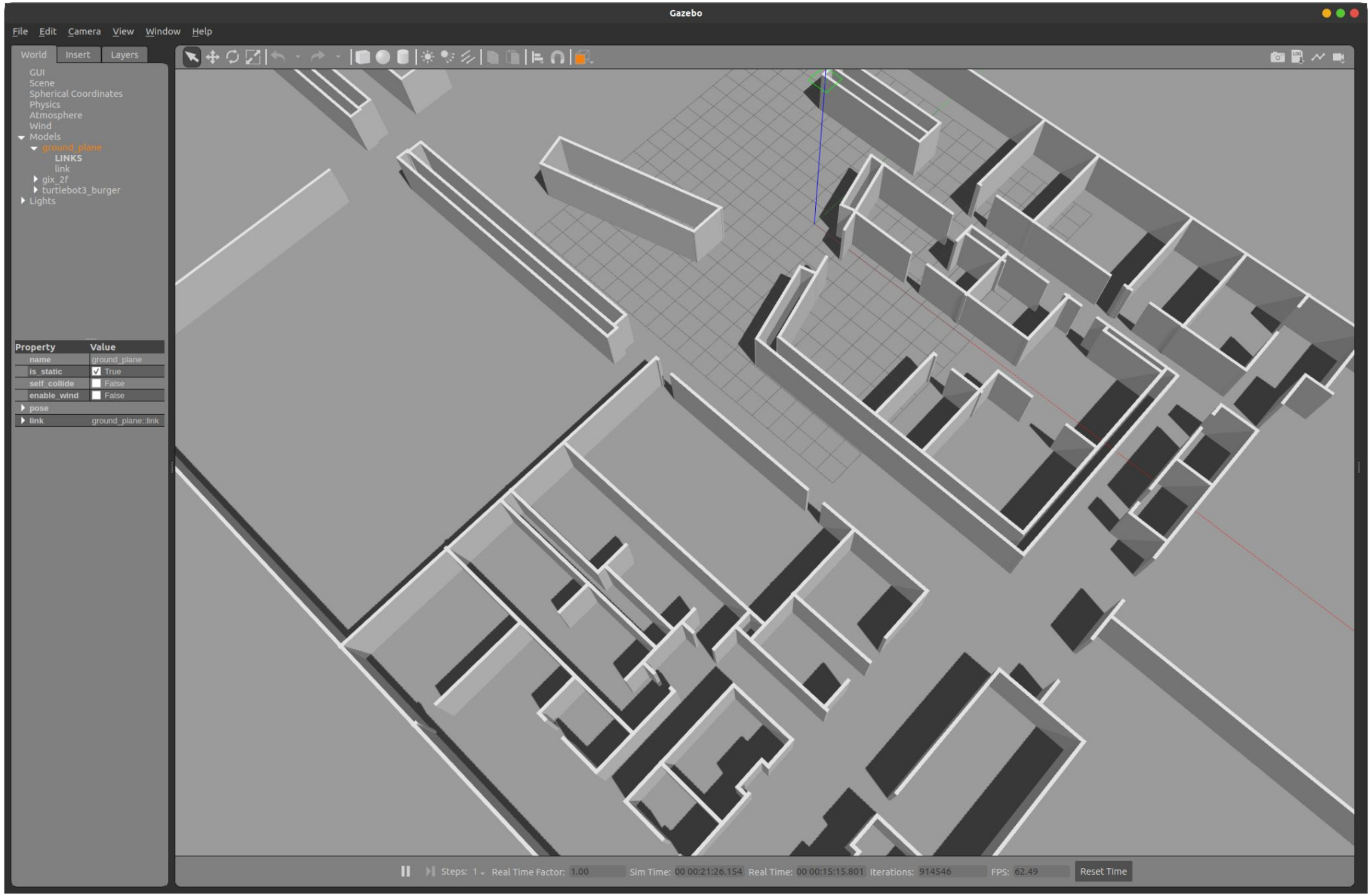
- Highly accurate maps that enhance navigation success and precision.

Change of Pickup Station


Old Version


New Version

High-Precision World Model with Simulation Package

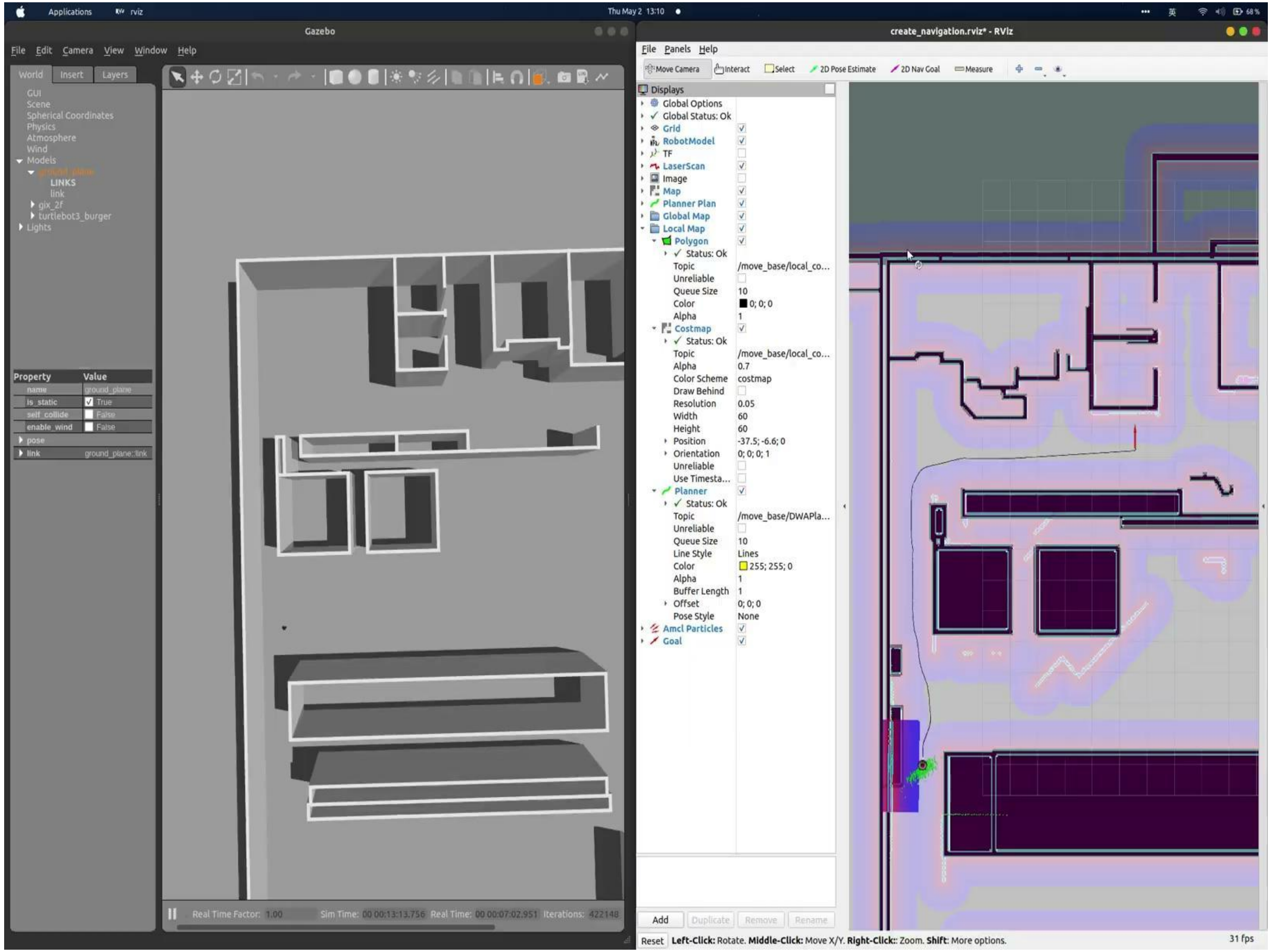


GUI
Scene
Spherical Coordinates
Physics
Atmosphere
Wind
Models
ground_plane
LINKS
link
gix_2f
turtlebot3_burger
Lights

Property	Value
name	ground_plane
is_static	✓ True
self_collide	False
enable_wind	False
pose	
link	ground_plane::link

Steps: 1 Real Time Factor: 1.00 Sim Time: 00 00:21:26.154 Real Time: 00 00:15:15.801 Iterations: 914546 FPS: 62.49

Simulation Demo



Property	Value
name	ground_plane
is_static	✓ True
self_collide	False
enable_wind	False
pose	
link	ground_plane::link

Real Time Factor: 1.00 Sim Time: 00 00:13:13.756 Real Time: 00 00:07:02.951 Iterations: 422148

Global Options	
Global Status: Ok	
Grid	✓
RobotModel	✓
TF	
LaserScan	✓
Image	
Map	✓
Planner Plan	✓
Global Map	✓
Local Map	✓
Polygon	✓
Status: Ok	
Topic	/move_base/local_co...
Unreliable	
Queue Size	10
Color	■ 0; 0; 0
Alpha	1
Costmap	✓
Status: Ok	
Topic	/move_base/local_co...
Alpha	0.7
Color Scheme	costmap
Draw Behind	
Resolution	0.05
Width	60
Height	60
Position	-37.5; -6.6; 0
Orientation	0; 0; 0; 1
Unreliable	
Use Timesta...	
Planner	✓
Status: Ok	
Topic	/move_base/DWAPla...
Unreliable	
Queue Size	10
Line Style	Lines
Color	🟨 255; 255; 0
Alpha	1
Buffer Length	1
Offset	0; 0; 0
Pose Style	None
Amcl Particles	✓
Goal	✓

Left-Click: Rotate. **Middle-Click:** Move X/Y. **Right-Click:**: Zoom. **Shift**: More options.

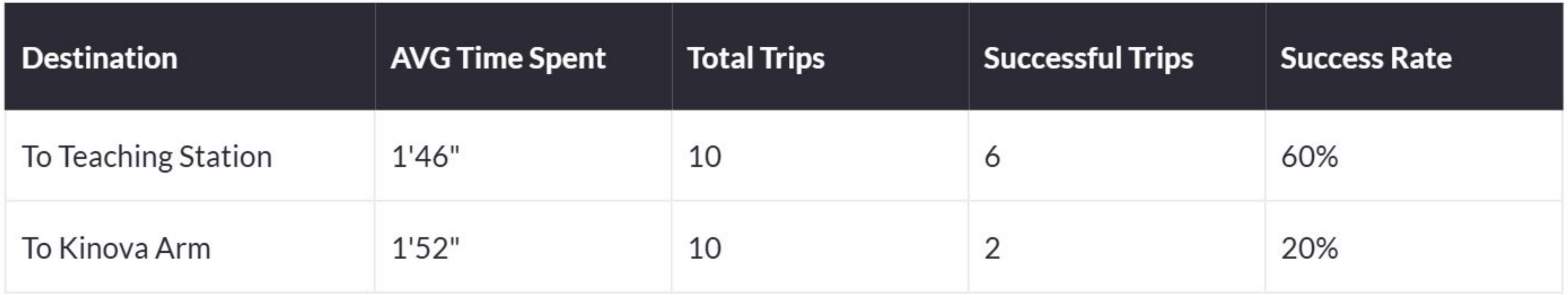
Physical Robot Experiments

Start Point: Kinova Arm (x=16.07, y=-32.44)
Destination 1: Classroom (x=25.57, y=-25.95)
Destination 2: Kinova Arm (x=16.07, y=-32.44)

Experimental Procedure: After calibrating the starting point of the robot on the map, send a fixed coordinate of destination 1 to make it self navigate and navigate to the set coordinate 1; If the target point 1 is successfully reached, send the fixed coordinates of target point 2 to navigate to it on its own. After arriving at the destination, the final error and success rate are determined by the displacement from the fixed coordinates.

Key indicators: success rate, time spent, accuracy of target location

Physical Robot Experiments



Destination	AVG Time Spent	Total Trips	Successful Trips	Success Rate
To Teaching Station	1'46"	10	6	60%
To Kinova Arm	1'52"	10	2	20%

Before Parameters Tuning

Inferred reasons for low success rate:

1. The map is not accurate enough, and there is an error in the relative position between the table and the wall;
2. Tables and miscellaneous items have errors in capturing spatial features by robots;
3. There is an issue with the navigation parameters of the robot itself.

Parameters Tuning

- **base_local_planner_params.yaml**
  - xy_goal_tolerance=0.1, yaw_goal_tolerance:0.2
- **costmap_common_params_burger.yaml**
  - inflation_radius=0.2, cost_scaling_factor=4.0
- **dwa_local_planner_params_burger.yaml**
  - xy_goal_tolerance=0.1, yaw_goal_tolerance:0.2

In the experiment, we attempted to modify many parameters, and after testing, the above parameters were the most useful ones

Parameters Tuning

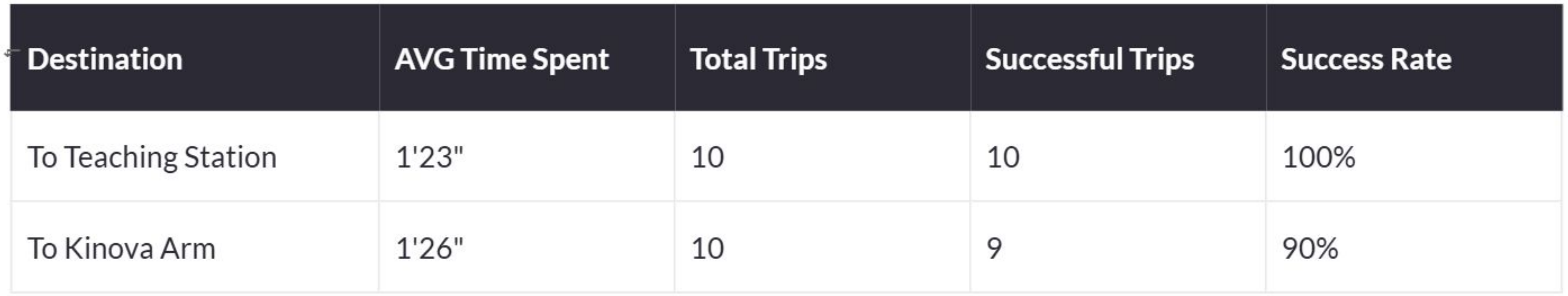
xy_goal_tolerance: This parameter defines the tolerance of the robot's navigation process towards the target position. Specifically, it determines the allowable error range for the robot to reach the target position during navigation tasks. The smaller the parameter, the more accurate the relative target position of the robot after reaching the destination

yaw_goal_tolerance: yaw_goal_tolerance defines the allowable range of orientation error (in radians) for a robot to reach its target position. The smaller the parameter, the more precise the orientation will be when reaching the target position. However, counterintuitively, if this parameter is smaller, the robot will take a long time to adjust after reaching its destination, so this parameter should be adjusted larger.

inflation_radius: infliction_radius is a key parameter used to set the radius of the obstacle expansion layer. The smaller this parameter, the closer the robot can approach the boundary of the obstacle.

cost_scaling_factor: cost_scaling_factor is used in robotic navigation to regulate the growth rate of Cost Value in Inflation Layer.It defines how the value of the expansion area is attenuated from the edge of the obstacle.The lower Cost_Scaling_factor will increase the value of the generation, and the robot may choose the path closer to the obstacle

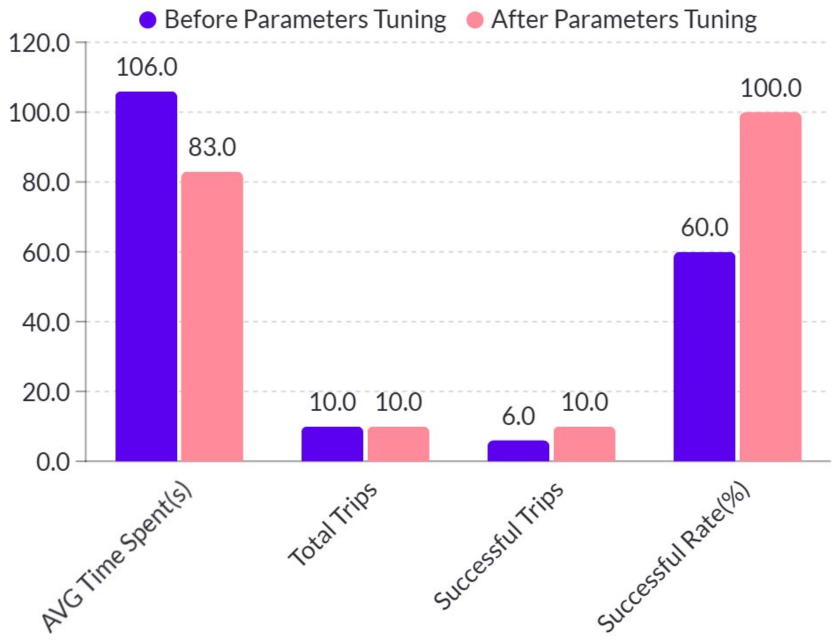
Physical Robot Experiments



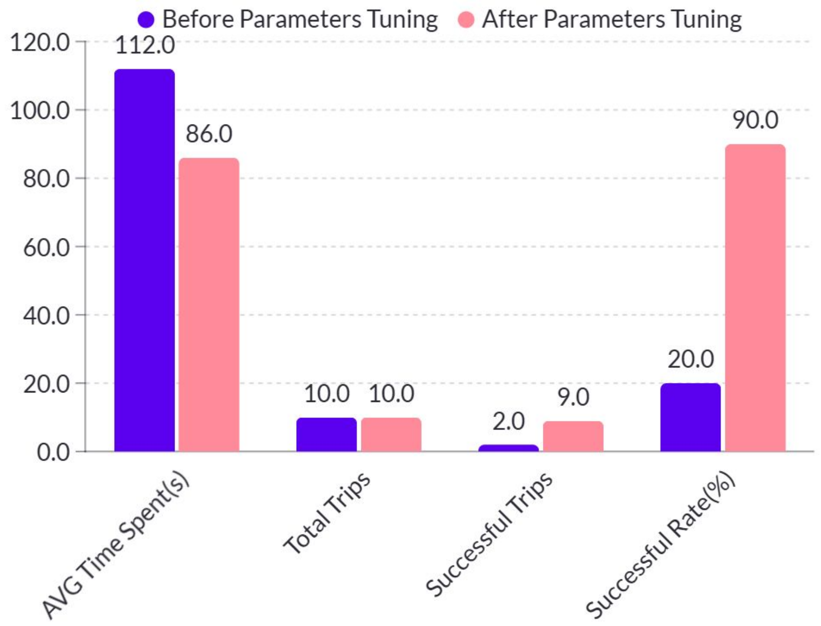
Destination	AVG Time Spent	Total Trips	Successful Trips	Success Rate
To Teaching Station	1'23"	10	10	100%
To Kinova Arm	1'26"	10	9	90%

After Parameters Tuning

Physical Robot Experiments

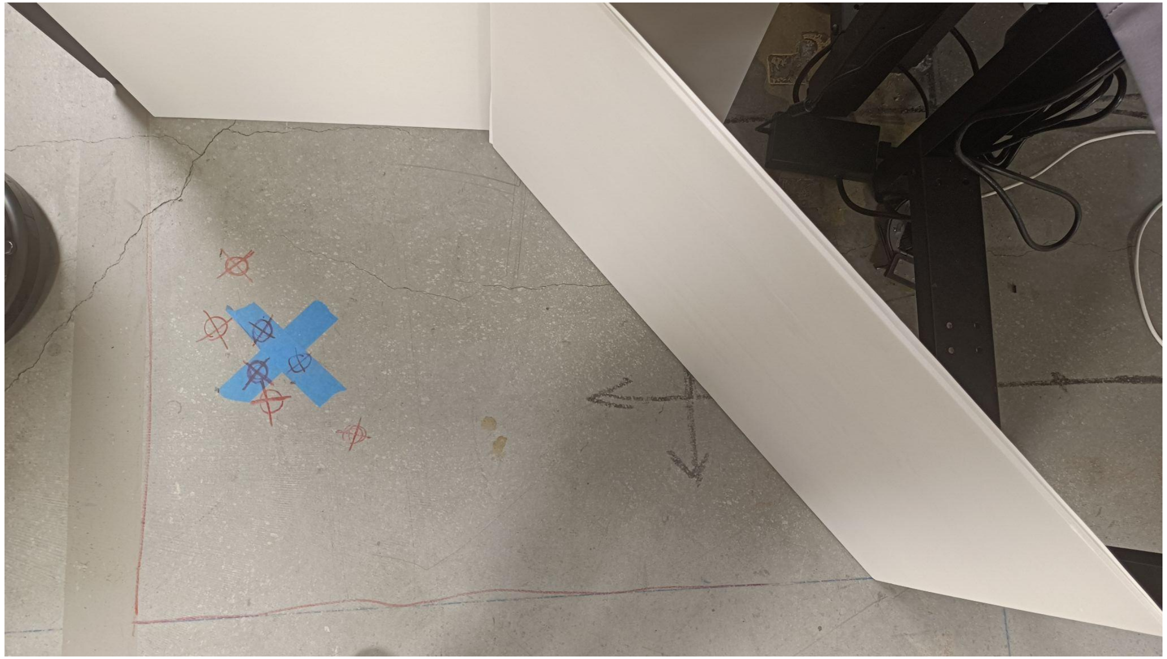


Comparison Experiment To Classroom Teaching Station



Comparison Experiment To Kinova Arm

Physical Robot Experiments



After parameter tuning and testing, the robot reaches the final target position with an error within the allowable range

Physical Robot Demo