EE 5900: Introduction to Robotics

Project 5

Working with the Jackal

Group 1: laughing-guacamole

Roger Gomes

Sabari Manohar

Deep Doshi

3/1/2017

Presentation Overview

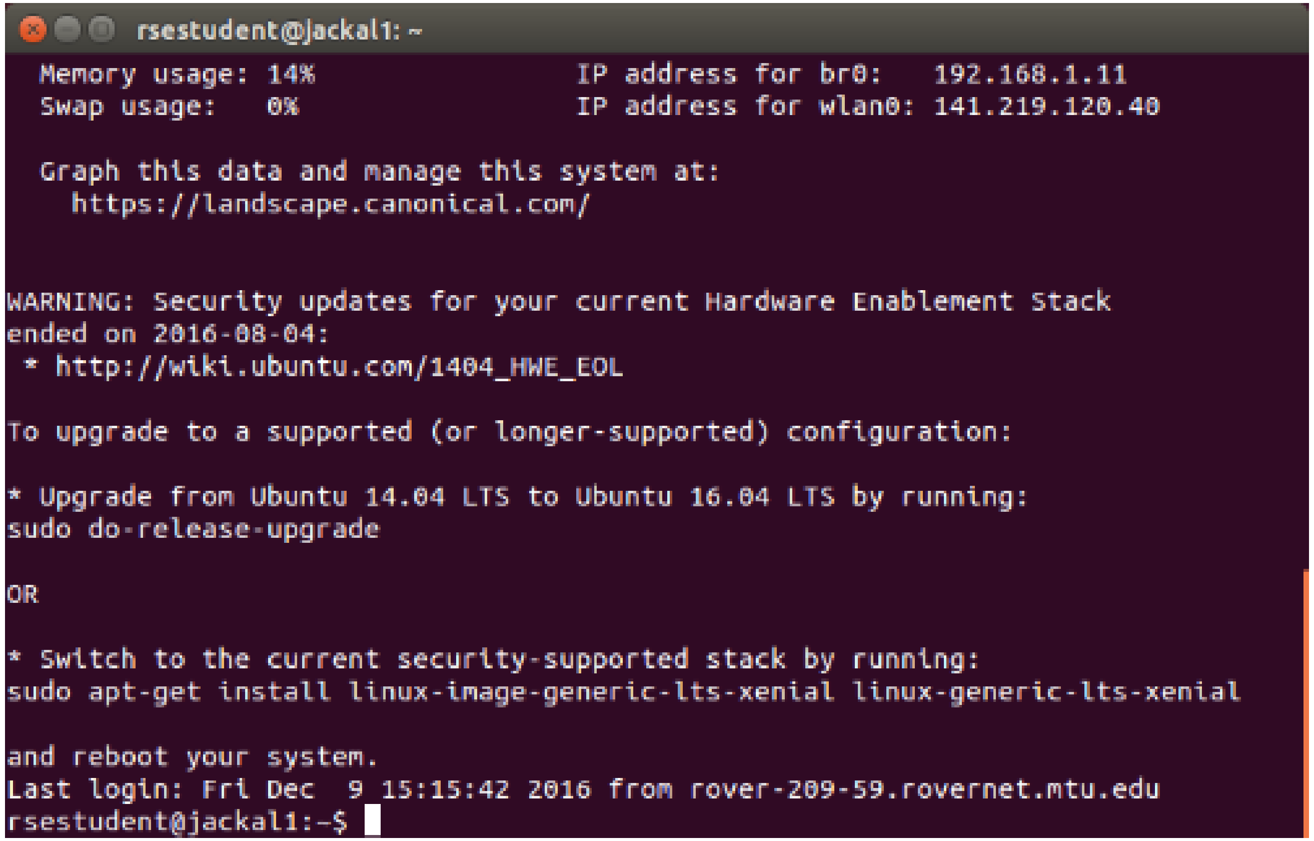
- Objective of the Project
- Communicating with Jackal
- LIDAR Integration
- Autonomous Navigation and rosbag
- rosbag, playback and Map recreation using gmapping
- Simulation Results for Maps generated
- Problems and Challenges Faced
- Credits and Distribution of Work

Project Objectives

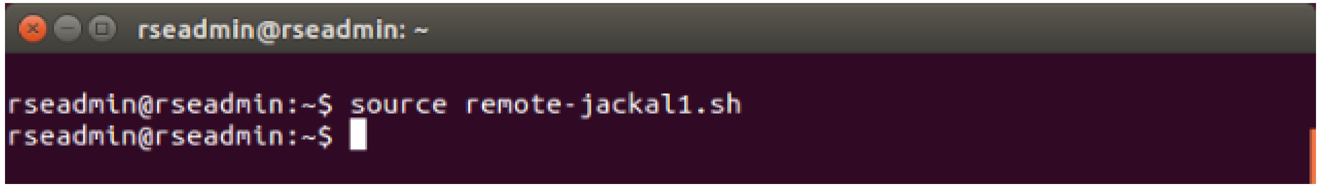
- Applying and leveraging the concepts learned from previous projects which heavily involved simulation of ClearPath Jackal in the Gazebo environment for exploring and mapping different worlds.
- Apply these concepts learned on the real world robotic platform - Clear Path Jackal equipped with SICK LMS 291 LIDAR.
- **Specific Objectives:**
  - Development of a URDF model for the LMS SICK 291 LIDAR and mounting bracket
  - The tf frame to be referenced for publishing the Laser scan data
  - Autonomous mapping routine and rosbag
  - Play back the recorded rosbag file and map reproduction using gmapping

Communicating with the Jackal

- Distributed Nature of ROS
- The Michigan Tech Jackals are registered on the MichiganTechOpen wireless network
- Editing the etc/hosts with Jackal 4 IP: **jackal4 141.219.120.42**
- ssh to the jackal
- ROS_MASTER_URI and the ROS_IP config connection details
- Allows us to run ros desktop tools for visulaization



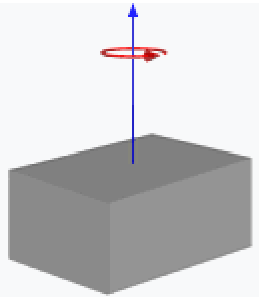
```
  Memory usage: 14%                IP address for br0:   192.168.1.11
  Swap usage:   0%                 IP address for wlan0: 141.219.120.40

  Graph this data and manage this system at:
    https://landscape.canonical.com/


WARNING: Security updates for your current Hardware Enablement Stack
ended on 2016-08-04:
 * http://wiki.ubuntu.com/1404_HWE_EOL

To upgrade to a supported (or longer-supported) configuration:

* Upgrade from Ubuntu 14.04 LTS to Ubuntu 16.04 LTS by running:
sudo do-release-upgrade

OR

* Switch to the current security-supported stack by running:
sudo apt-get install linux-image-generic-lts-xenial linux-generic-lts-xenial

and reboot your system.
Last login: Fri Dec  9 15:15:42 2016 from rover-209-59.rovernet.mtu.edu
rsestudent@jackal1:~$
```



```
rseadmin@rseadmin:~$ source remote-jackal1.sh
rseadmin@rseadmin:~$
```

SICK LMS 291 LIDAR Integration

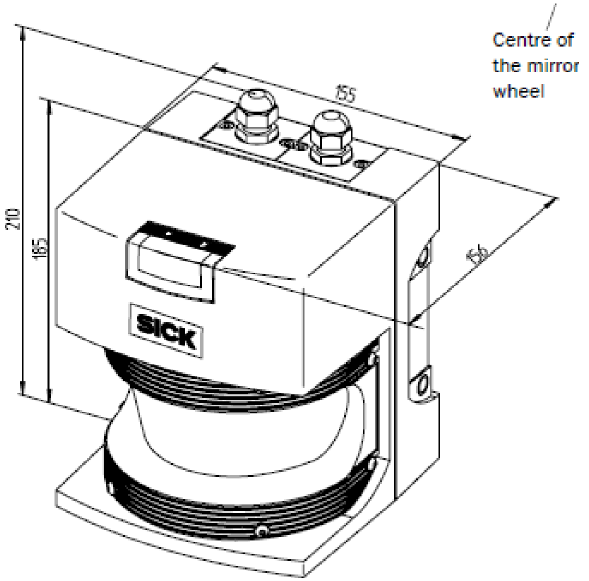
- URDF file defined the geometry, size, mass, origin and inertial values.
  - Dimension: 0.156mm x 0.155mm x 0.210 mm
  - The mass of the LIDAR was assumed to be uniform at 4.5 kg
  - Inertia values were also defined by specifying ixx, iyy and izz



$$I_h = \frac{m}{12}\left(w^2 + d^2\right)$$

$$I_w = \frac{m}{12}\left(d^2 + h^2\right)$$

$$I_d = \frac{m}{12}\left(w^2 + h^2\right)$$



```
<inertia ixx="0.02566348973" ixy="0.0" ixz="0.0" iyy="0.02937186325" iyz="0.0" izz="0.02196036621"/>
```

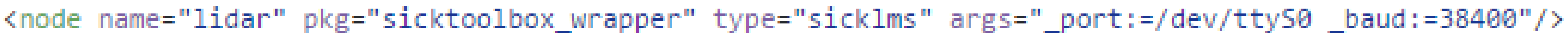
- Laser mounting bracket geometry was defined with the help of an .stl file that was referenced in the URDF file.

```
<mesh filename="package://lab_5_floor_mapper/src/meshes/LidarMount-1.stl" scale="0.0012 0.0012 0.0012" />
```

- Links and joints were also defined in this file which specified which component is mounted on what, relative to the front-mount on the jackal.

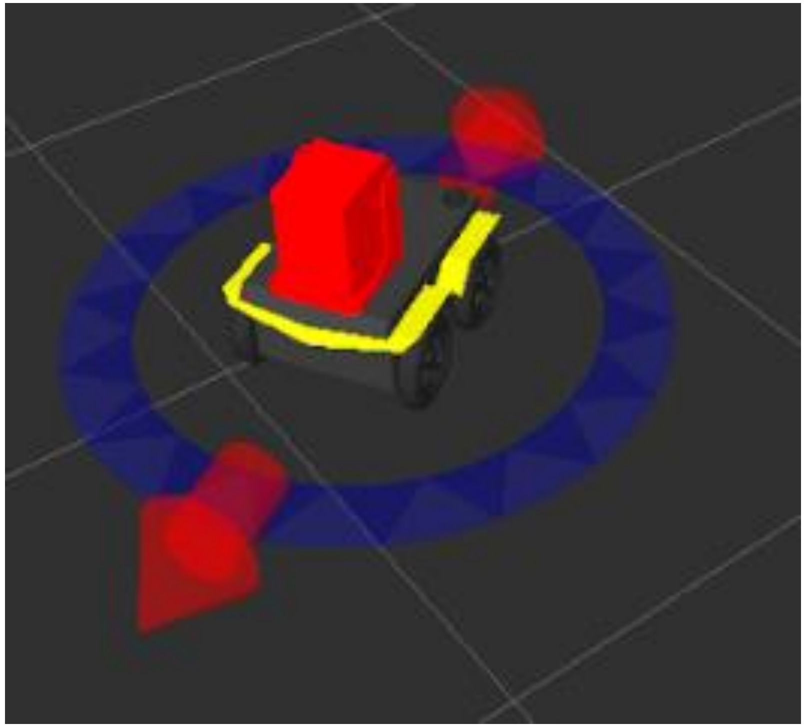
SICK LMS 291 LIDAR Integration

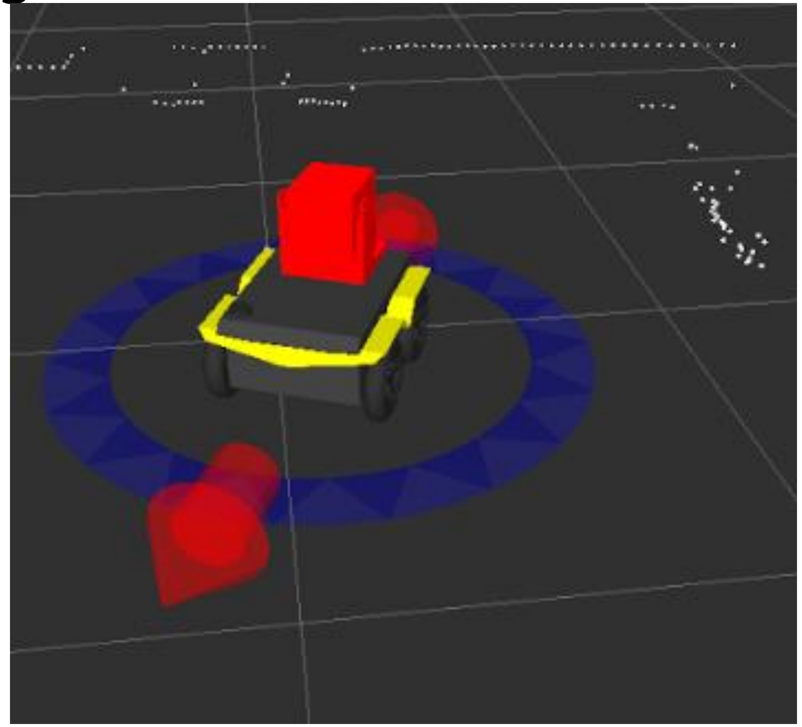
- SICK Toolbox wrapper
  - Setting of Port and Baud Rate in the launch file



```
<node name="lidar" pkg="sicktoolbox_wrapper" type="sicklms" args="_port:=/dev/ttyS0 _baud:=38400"/>
```

- Jackal with the Laser and Laser Mount Integration as seen on RViz

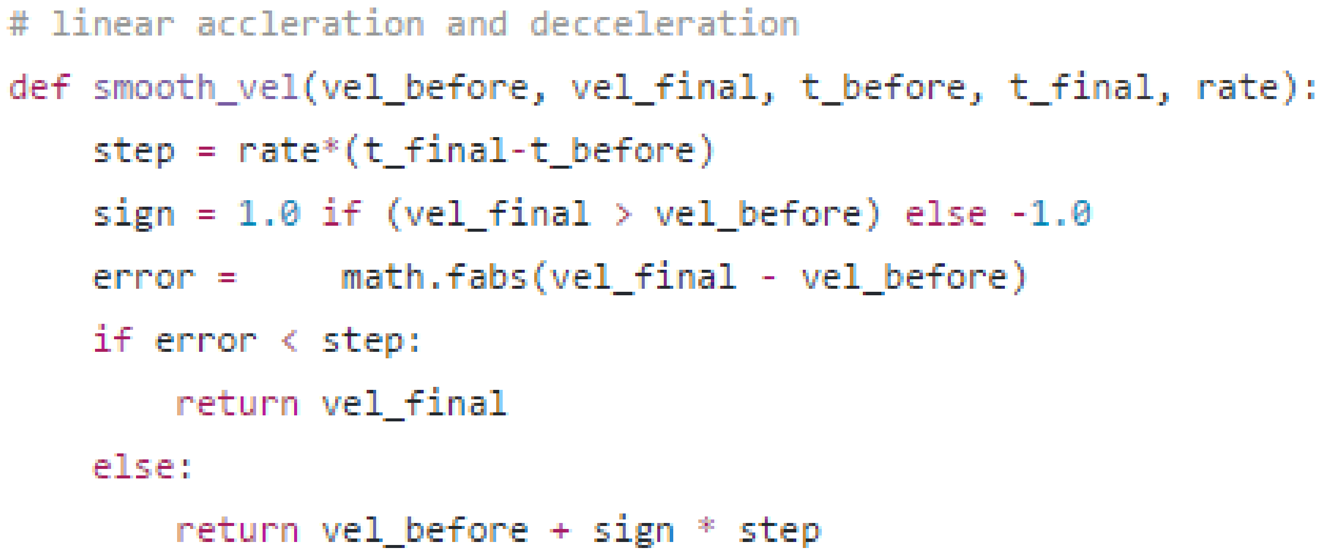




Autonomous Navigation and Mapping

- Autonomous Navigation and Mapping routine developed to map the 8th Floor of EERC
- Leverage and heavily reused a lot of code and algorithm developed by Derek, James and Akhil as a part of the Project 3: Perception Map Building
- Apart from the main algorithm, a lot of tweaking and modifications were performed on the constants for laser averaging, random bounds, and side thresholds
- To eliminate the jerkiness in movement we implemented a velocity smoothing function for linear acceleration and deceleration

Autonomous Navigation and Mapping



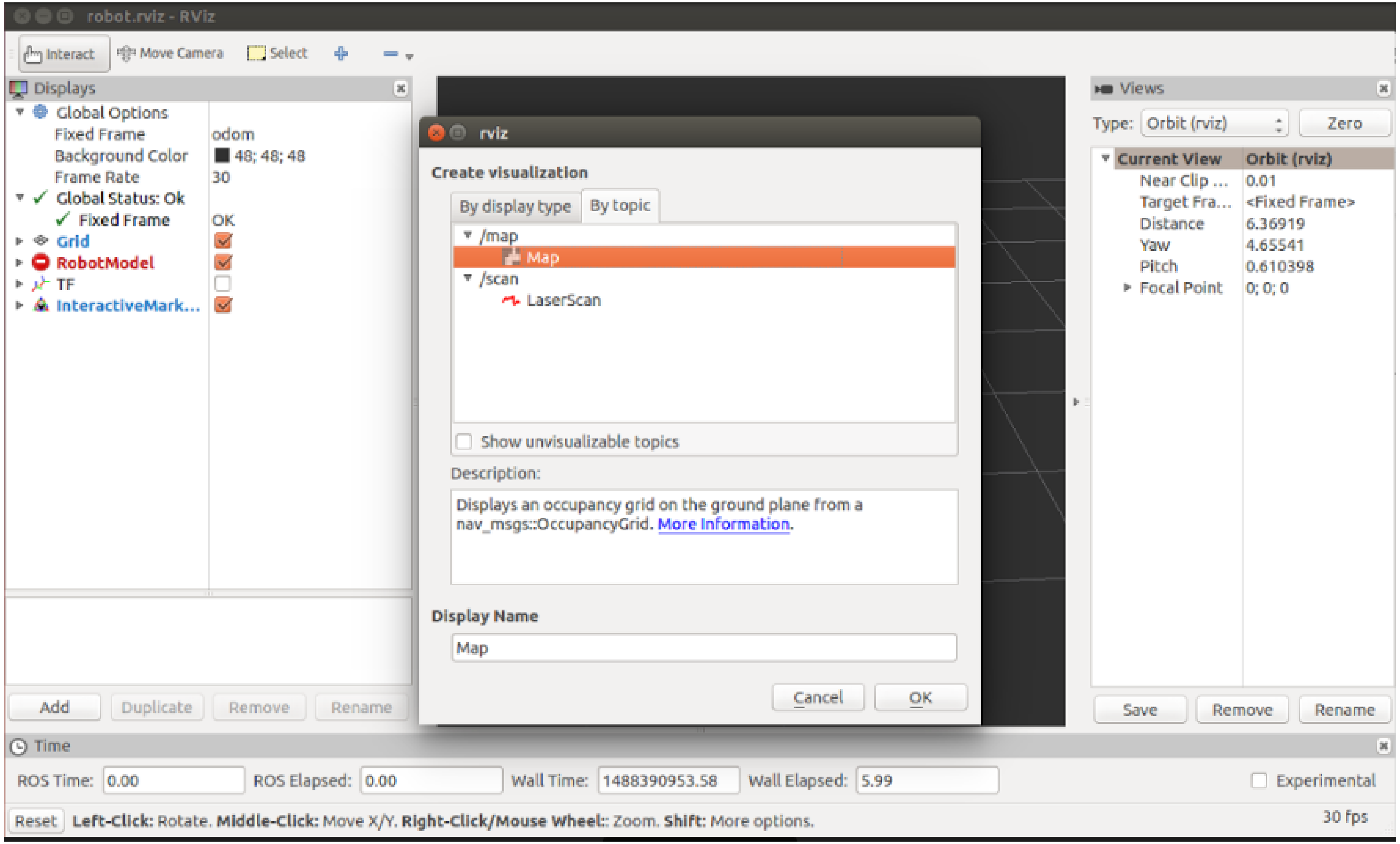
```
# linear accleration and decceleration
def smooth_vel(vel_before, vel_final, t_before, t_final, rate):
    step = rate*(t_final-t_before)
    sign = 1.0 if (vel_final > vel_before) else -1.0
    error =     math.fabs(vel_final - vel_before)
    if error < step:
        return vel_final
    else:
        return vel_before + sign * step
```

- Simultaneously Bags the /tf and /scan topics during mapping routine
- Termination of program after 5 min 30 sec
- Saves a bag file that is played back for gmapping

Rosbag, playback and Map recreation using gmapping

- A parallel invocation of the rosbag recording is also initiated from the same launch file.
- The recorded rosbag is saved once the autonomous mapping routine node's timer expires eventually killing the node.
- The recorded rosbag is played back using another launch file that also launches an instance of gmapping to reproduce the map using the /scan and /tf topics previously logged by the rosbag file.
- The process of map development can also be visualised in real-time using Rviz when the offline playback to recreate the map using gmapping is performed.

Visualizing Map Creation on RViz



Global Options
Fixed Frame: odom
Background Color: ■ 48; 48; 48
Frame Rate: 30
✓ Global Status: Ok
✓ Fixed Frame: OK
Grid ☑
RobotModel ☑
TF □
InteractiveMark... ☑

rviz

Create visualization

By display type | By topic

/map
- Map

/scan
- LaserScan

□ Show unvisualizable topics

Description:

Displays an occupancy grid on the ground plane from a nav_msgs::OccupancyGrid. More Information.

Display Name: Map

Type: Orbit (rviz)

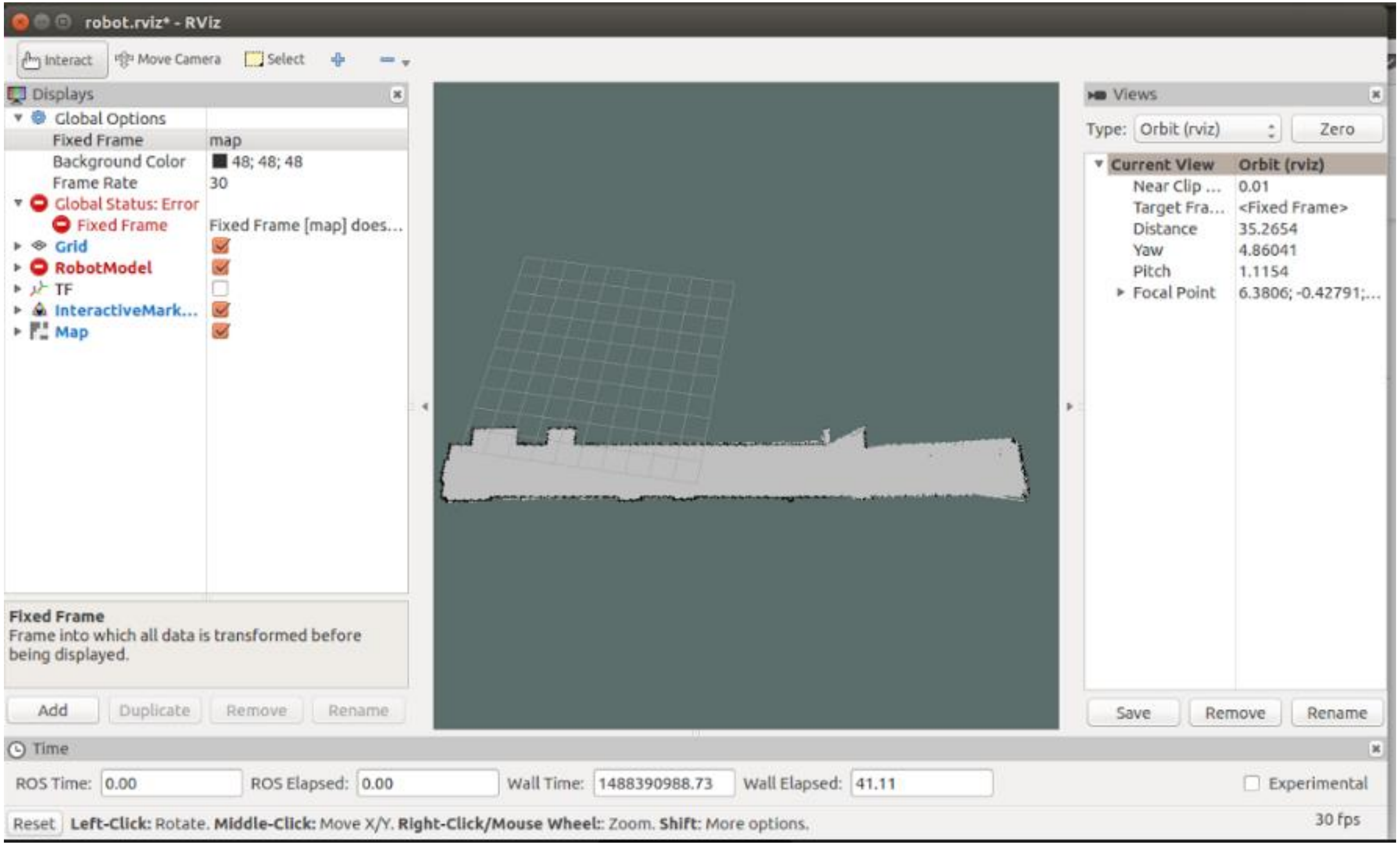
Current View	Orbit (rviz)
Near Clip ...	0.01
Target Fra...	<Fixed Frame>
Distance	6.36919
Yaw	4.65541
Pitch	0.610398
Focal Point	0; 0; 0

ROS Time: 0.00 ROS Elapsed: 0.00 Wall Time: 1488390953.58 Wall Elapsed: 5.99 □ Experimental

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

30 fps

Visualizing Map Creation on RViz



Displays	
Global Options	
Fixed Frame	map
Background Color	■ 48; 48; 48
Frame Rate	30
Global Status: Error	
Fixed Frame	Fixed Frame [map] does...
Grid	☑
RobotModel	☑
TF	□
InteractiveMark...	☑
Map	☑

Fixed Frame
Frame into which all data is transformed before being displayed.

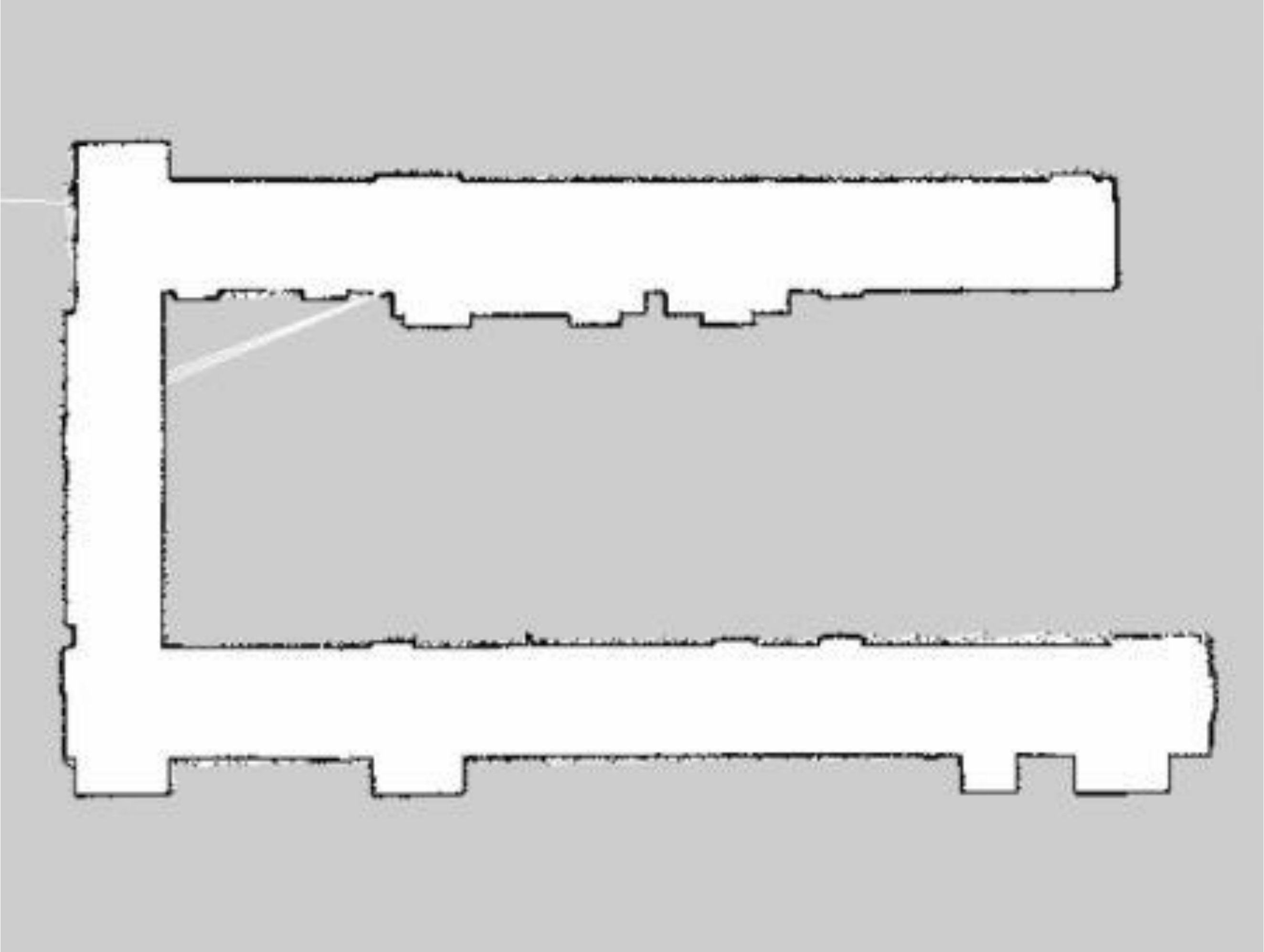
Type: Orbit (rviz)

Current View	Orbit (rviz)
Near Clip ...	0.01
Target Fra...	<Fixed Frame>
Distance	35.2654
Yaw	4.86041
Pitch	1.1154
Focal Point	6.3806; -0.42791;...

ROS Time: 0.00 ROS Elapsed: 0.00 Wall Time: 1488390988.73 Wall Elapsed: 41.11 □ Experimental

Left-Click: Rotate. **Middle-Click:** Move X/Y. **Right-Click/Mouse Wheel:**: Zoom. **Shift:** More options. 30 fps

Simulation Results for Maps Generated



Best Map generated for the 8th floor out of all the runs

Problems and Challenges Faced

- **Controlling the Jackal via Rviz through Virtual Machine:**
  - Needed to change ROS_IP from WAN to Ethernet to enable bridge connection from the host computer to the VM on which connection to the ROS_MASTER_URI was established.
- **Co-ordinate and scale adjustment of the Lidar, mounting bracket:**
  - While incorporating the LIDAR definitions and the mounting bracket on the Jackal, problems with its orientation and scaling were faced which were eliminated by adjusting its coordinates relative to the robot.
- **Malformed/Invalid URDF:**
  - Invalid xml definition for the URDF caused the rosmaster to crash and we were unable to connect to jackal's rosmaster instance. Used the URDF check utility available through the URDF tutorials before actually loading it on the jackal.
- **Loss of Wifi Connectivity:**
  - At certain locations (near the elevator) the Michigan Tech wi-fi connections were spotty causing us to loose connections. In the cases where the launch file was launched through a normal terminal connection, there were problems with node terminations. Hence, screen command was used to launch a remote terminal in jackal itself for the node to run
- **Navigating of obstacles not in LOS:**
  - Corridor with chairs situated above the height of LIDAR was problematic. Did not map that portion.

Credits and Distribution of Work

Sabari Manohar

- Initial development, modification, tweaking and calibration of thresholds for the autonomous navigation and mapping routine
- Development of velocity smoothing functionality to eliminate jerky movements
- Testing and debugging

Deep Doshi

- SICK Toolbox wrapper integration
- Development of launch files for navigation and replay routines
- Integration Testing for map reproduction, rosbag recording and replay

Roger Gomes

- Development of the URDF File for LIDAR, LIDAR Mounting bracket definitions
- Development of scripts and playback of recorded rosbag and gmapping to reproduce the map
- Tweaking, modifications and calibration for mapping and navigation routine
- Integration Testing, debugging, Team Management, Task Distribution and Documentation

Questions?