

The goal of this project is to develop an intelligent Smart Hotel System to cater to the needs of Poe, the manager of The Raven Hotel whose neural networks have severely damaged. The neural network is assumed to provide security and surveillance and intelligent monitoring of different subsystems.

My proposed solution is accomplished by designing, modelling, constructing and assembling a plethora of sensors and robots across multiple software platforms like freeCad, Blender, gazebo and interconnected and collaborating among themselves.

I would be using the following sensors and robots in my proposed solution which would try to offset the lack of the existing neural networks and give better security and surveillance not only through use of cutting technology and using the different high end sensors but also actively collaborating and Sharing the different sensor inputs to take critical decisions.

### **Velodyne HDL-32 LIDAR** for obstacle detection and distance detection

- created an SDF model of the HDL-32 sensor

- improved the model's appearance and data output and added Mass/Inertia to the model

- Velodyne has a STEP file[A STEP file is a 3D model file formatted in STEP (Standard for the Exchange of Product Data)] for the HDL-32 located on their website. Gazebo can only use STL, OBJ or Collada files, so we'll have to convert this file and then add it to our model.

- So,I used freeCad software to acquire Meshes, Blendr software to refine the metric system and Gazebo model editor to model the Velodyne Lidar structure. We can optionally download additional textures from the net subsequently.

- visualized the sensor data in Gazebo and RViz.

Noisy Laser to discriminate distance and to support critical decisions making for my robotic systems.

- implemented a basic Hokuyo fake laser scanner [(hokuyo\_node) in ROS] which is a simple steady laser range finder and is used to detect obstacles in its field of vision.

- tweaking the Gaussian noise with respect to the mean and standard deviation of the distribution results in different resolution of the scan samples.

- the laser scan video demonstrates how the laser detects the relative spacing between the laser node and the nearby obstacle and displays the resultant distance as a float value and returns "inf" when there's no obstruction.

## **Noisy Camera** for surveillance both indoor and outdoor

-implemented a simple Noisy camera which works very similar to the noisy Laser node.

-one can visualize the imagery reading via the Gazebo Topic Selector from the menu to get the image samples of the noisy camera. This can be intelligently positioned at several locations inside and outside of the hotel to serve as a CCTV camera.

-Again, one can tweak the values of the mean and standard deviation of the Gaussian distribution to get the desired signal-to-noise ration via the model.sdf file.

The Relevant Parameters of the same are as follows:

```
<noise>
  <type>gaussian</type>
  <mean>"The desired mean value"</mean>
  <stddev>"The desired standard deviation value"</stddev>
</noise>
```

## **TurtleBot Laser and Vehicle Depth Camera Sensor** used for continuous patrolling and imaging for surveillance

-Simulated the ROBOTIS waffle\_pi/burger TurtleBot3 provided by the ROBOTIS team, Open Source Robotics Foundation (OSRF) et al.

- The TurtleBot3 can provide on-command patrol at a desired location via the ROS Gazebo Teleop package which internally translates the keyboard bindings into velocity commands and publishes them to the robot through cmd\_vel topic.

-Constructed a simple vehicle mounted with a depth camera sensor using Gazebo model editor which can be parked in vehicle lot and provide the static camera readings via the depth camera feed.

## **Clearpath Husky UGV**

-Simulated the Husky UGV, which is an "Outdoor Field Research Robot" provided by Clearpath Robotics in Gazebo .

-Ideology:

The Husky due to its average stature can manoeuvre through nooks and corners of the hotel indoors (or the hotel outdoors including the Parking lot) providing intelligent Gmapping SLAM (Simultaneous Localization and Mapping) with two SICK LMS511 LIDARs. This could reach the various locations of the hotel and cater to needs of an emergency situation and also provide a moving surveillance of the interior of the hotel.

-Model working:

#### MAPPING===>

With reference to the ROS navigation stack, the husky references the 360 degrees laser scan with the LIDARs, the Odometry readings from its wheel encoders and various other sensor transforms(using ROS TF) since the various parts of the robot co-exist with different frames of references to construct a local costmap of the region.

#### LOCALISATION===>

-After mapping of the environment, we proceed onto the localization since without knowing the local position of the robot with respect to the map renders the map useless. The AMCL(Adaptive Monte Carlo Localization) package provides a systematic approach to estimate the position of the robot using a Particle Filter(In short: the robot randomly guesses all future poses of robot and also full description of those poses. When the robot moves, it disregards the erroneous guesses via sensor readings thus resulting in convergence of the particle guesses).

#### NAVIGATION/PATH PLANNING==>

Consequently, the robot proceeds to implement the following Path planning algorithms to reach the desired destination or cater to needs of an emergency situation.

Global Planners:

A\*

Dijkstra

Local Planners:

Dynamic Window approach

TEB local planner

We can simultaneously visualize the sensor readings via Rviz.

Side note: We can also control the Husky Bot using the teleop command and a PS3 joystick.

Further ideas:

We could additionally enhance the security of the Hotel by providing Aerial surveillance through Quadcopters.

Important Note:

Due to the lack of powerful graphics computing hardware, I am simulating the model in 2 different instances.

#### **Concluding Remarks:**

Although, I am currently a novice at ROS programming with C++/Python, I assure you that I am a very passionate and hard-working student (2nd year Manipal Computer Science and Engineering) with a profound desire to learn.

I am currently constrained by the everyday college schedule and am also pursuing 2 internships one from Machine Learning through Microsoft and another one on Data Analytics through United Nations TakenMind thus resulting in the delay of the project submission. Upon completion of those internships, I would be equipped with various machine learning and computer vision techniques that I could use to contribute to the team. If you feel my qualifications are a good fit for the position, i'd love to get a chance to intern at your Robotics Startup.

### **Resources used for the project:**

Robot Ignite Academy

<http://emanual.robotis.com/docs/en/platform/turtlebot3/simulation/#ros-1-simulation>

<http://wiki.ros.org/navigation>

[http://wiki.ros.org/hokuyo\\_node](http://wiki.ros.org/hokuyo_node)

[http://wiki.ros.org/laser\\_filters](http://wiki.ros.org/laser_filters)

<https://clearpathrobotics.com/husky-unmanned-ground-vehicle-robot/>

[http://gazebosim.org/tutorials?tut=sensor\\_noise&cat=sensors](http://gazebosim.org/tutorials?tut=sensor_noise&cat=sensors)

[http://gazebosim.org/tutorials?cat=guided\\_b&tut=guided\\_b3](http://gazebosim.org/tutorials?cat=guided_b&tut=guided_b3)

<https://github.com/idincern/idincern-husky>

<http://wiki.ros.org/amcl>

[http://gazebosim.org/tutorials/?tut=ros\\_depth\\_camera](http://gazebosim.org/tutorials/?tut=ros_depth_camera)

<http://wiki.ros.org/kinect>

[https://answers.ros.org/question/274564/difference-between-dwa-local\\_planner-and-teb-local\\_planner/](https://answers.ros.org/question/274564/difference-between-dwa-local_planner-and-teb-local_planner/)

[http://wiki.ros.org/mcl\\_pi](http://wiki.ros.org/mcl_pi)

<http://gazebosim.org/tutorials>