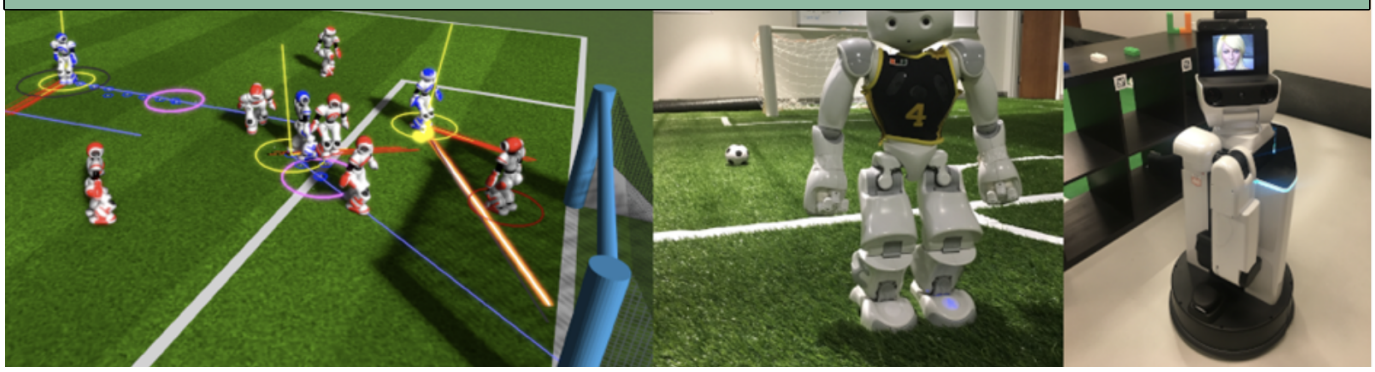
Fall 2024 - CSC398 Autonomous Robots - Assignment 4



Due date: 10/17/2024, 12:30pm, before class starts. This assignment is worth 20 points.

The goal of this exercise is to work with data that was recorded on a simulated HSR robot. The recorded bag file contains sensor measurements from odometry `/hsrb/odom`, the inertial measurement unit (IMU) `/hsrb/base_imu/data` and a laser scanner `/hsrb/base_scan`. Your task is to use this raw sensor data to localize the robot with a provided state estimation node that implements an extended Kalman filter (EKF). Don't worry: you don't have to program this, it is a ROS node that you simply add your project. To see the results, plot the output of the localization node using `rqt_multiplot` and visualize the laser data in RViz.

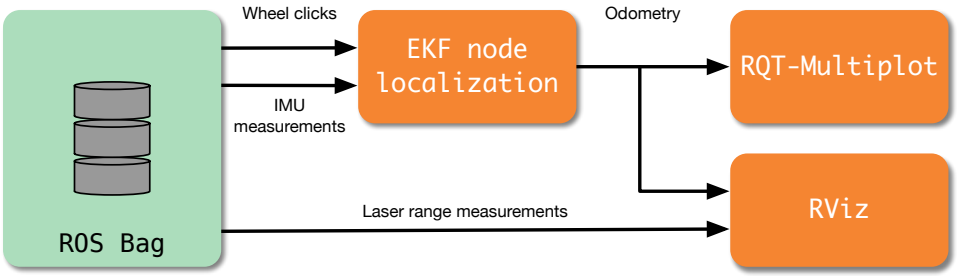


Figure 1: Connection and data flow for the HSR ROS bag.

1. We made some small modifications to the simulation world. `git pull` (from your catkin workspace) to update.
2. Run the Isaac Simulator using the start script with the RoboCanes Lab. We have created an environment where the robot can move to four pre-defined poses.
3. Check the topic `/odometry/filtered` and find out what the node `ekf_localization` is doing. What is it subscribing to and what is it publishing? Visit http://docs.ros.org/kinetic/api/robot_localization/html/index.html and [MS14] for further information.
4. `cd` into the `src/` directory off your catkin workspace. Clone the new assignment: `git clone` https://classroom.github.com/a/x-gnz8qc. Run the Python script in the "script" directory called `init_package.py`. It will change the package name to match yours (ex. assignment-4-username). Compile and source the package. If the system tells you the file cannot be run because of permissions change the permissions with `chmod +x init_package.py`.
5. Use `rqt_mulitplot` to plot the path of the simulated robot in the x/y-plane by using the topic `/odometry/filtered`.
6. rosrun the script `predefposes.py` and record the session with rqt_multiplot.
7. Download the provided rosbag `hsr_navigation.bag` and investigate the content with the command `rosbag info`.
8. Use the provided `hsrb_bag.launch` file, add a node that starts an `ekf_localization_node` subscribing to the topics mentioned in the above description of the assignment. Load the parameters from the config file `localization.yaml`, which can be found in the `/config` directory in your package.
9. Use `rqt_mulitplot` to plot the path of the recorded robot in the x/y- plane.
   - Remember to set the parameter `/use_sim_time` to `true`: http://wiki.ros.org/Clock.
   - Play the bag-file with: `rosbag play` *hsrb_navigation.bag* `--clock` which also publishes the time of the recorded data http://wiki.ros.org/rosbag/Commandline.

10. Visualize the motion of the HSR using TF markers in RViz. Add a `robot_state_publisher` (http://wiki.ros.org/robot_state_publisher) node to your launch file. Now you can visualize the HSR model in RViz.

11. Add following files to your package: a) a screenshot of the x/y- plane of the *simulated* HSR in rqt_multiplot, b) a screen shot of the x/y plane of the *recorded* HSR in rqt_multiplot.

12. Add, commit, and push the entire package to github classroom.

References

[MS14] T. Moore and D. Stouch. A Generalized Extended Kalman Filter Implementation for the Robot Operating System. In *Proceedings of the 13th International Conference on Intelligent Autonomous Systems (IAS-13)*. Springer, July 2014.