Guide for Running the `lidar_filtering_assignment` Docker Image

Dr Mohamed H H Abdelwahab

January 17, 2025

Contents

1 Introduction

This document explains how to **pull** the image from Docker Hub and run the project locally, including configuration for ROS bag playback and network settings.

2 Prerequisites

- Docker installed on the local machine.

- ROS installed on the host if you plan to run GUI tools like `rviz` natively, though this is not strictly required if everything runs in the container.

3 Usage Guide for New Users

Once the image is published on Docker Hub, any user can pull and run the project by following these steps.

3.1 Pull the Docker Image

```
docker pull mohheshmat/lidar_filtering:latest
```

3.2 Run the Container

A simple run command might look like this:

```
docker run -it --rm \
    --env="DISPLAY" \
    --env="QT_X11_NO_MITSHM=1" \
    --volume="/tmp/.X11-unix:/tmp/.X11-unix:rw" \
    -v /home/user/Data:/home/m/Data \
    mohheshmat/lidar_filtering:latest
```

Explanation:

- `-it`: Creates an interactive terminal.

- `--rm`: Removes the container upon exit.

- `--env="DISPLAY"` and `--env="QT_X11_NO_MITSHM=1"`: Passes display environment variables into the container so GUI tools (like `rviz`) can run (on native Linux).

- `-v /tmp/.X11-unix:/tmp/.X11-unix:rw`: Maps the X11 socket for GUI forwarding (again for native Linux).

- `-v /home/user/Data:/home/m/Data`: Maps a local folder `/home/user/Data` into `/home/m/Data` inside the container, so you can place bag files there.

3.3 ROS Environment Inside the Container

Once inside the container, **you need to commented out the RVIZ node in the launch file** and then:

```
source /opt/ros/noetic/setup.bash
source /catkin_ws/devel/setup.bash
roslaunch lidar_filtering_assignment lidar_filtering.launch
```

Then, in a separate shell or background:

```
rosbag play /home/m/Data/LiDARFilteringAssignment.bag --loop
```

3.4 Run `rviz` on the Host

If you prefer to keep `rviz` on your host machine, ensure:

1. `ROS_MASTER_URI` is set to the container's IP.

2. The container and host can communicate (may require bridging or `--network=host` if on a native Linux machine).

Then simply run `rviz`:

```
rviz
```

And add the relevant topics (e.g., `/filtered_points`, `/removed_points`) for visualization.

4 Conclusion

By publishing your Docker image to Docker Hub, **any user** can quickly:

1. `docker pull myuser/lidar_filtering:latest`

2. `docker run` the container with the recommended environment settings

3. Start **ROS** nodes or **play** bag files to demonstrate the filtering project.

This enables a reproducible setup with minimal local dependencies, other than **Docker** itself.