High-Level Plan for LiDAR Ground & Noise Filtering

Dr Mohamed Heshmat Hassan Abdelwahab

January 17, 2025

Contents

1 Introduction

This document provides a high-level plan for filtering ground and noise points in LiDAR data from a mobile robotic platform equipped with multiple sensors. It also outlines strategies for improving the filtering process by fusing data from other available sensors (*e.g.*, cameras and IMUs).

2 Summary of ROS Bag Data

The bag file, `LiDARFilteringAssignment.bag`, contains:

- **Four camera streams:**
  - Front camera
  - Back camera
  - Left camera
  - Right camera
- **Three LiDAR data streams (Ouster-based):**

  - /mbuggy/os1/points
  - /mbuggy/os2/points
  - /mbuggy/os3/points

- **Three IMU data streams:**
  - /mbuggy/os1/imu
  - /mbuggy/os2/imu
  - /mbuggy/os3/imu

- **Additional Nodes:**
  - Nodelet managers for cameras and LiDARs: `/mbuggy/nodelet_manager_camera_(...)`, `/mbuggy/nodelet_manager_os(...)` (used to run multiple nodelets within a single process).
  - Various image topics: raw and compressed image streams.
  - `/tf` and `/tf_static` transformations.
  - `/rosout`, `/clock`, and other standard topics.
  - Bag duration: 29.2 s, size: 14.9 GB.

3 Ground and Noise Filtering for One LiDAR Stream

This section describes a step-by-step plan to remove ground and noise from a single LiDAR topic (for instance, `/mbuggy/os1/points`).

3.1 Step 1: Data Preprocessing

- **Convert ROS Message to PCL Format**: Use `pcl_conversions` and `pcl_ros` to transform `sensor_msgs::PointCloud2` into a `pcl::PointCloud`.
- **Basic Filtering**: Apply a simple pass-through filter if any known height or distance bounds are available. This quickly removes obvious out-of-range noise.

3.2 Step 2: Ground Removal (Plane Segmentation)

- **RANSAC Plane Fitting**: Perform plane segmentation (e.g., using `pcl::SACSegmentation`) to detect the largest plane, hypothesized to be the ground.
- **Extract Inliers/Outliers**: Separate inliers (ground) from outliers (above-ground data). This yields two point clouds:
  - *Ground Cloud*
  - *Non-Ground Cloud*

3.3 Step 3: Noise Removal

- **Statistical Outlier Removal (SOR)** or **Radius Outlier Removal**:
  - For instance, set `MeanK = 50` and `StddevMulThresh = 1.0` to remove spurious dust or stray points.
  - The output is a *clean* point cloud with minimal noise.

3.4 Step 4: Publish Results

- **Filtered Point Cloud**: All above-ground points that remain after noise removal.
- **Removed Point Cloud**: Combination of ground points and the noise points. Used for debugging or optional reuse.

3.5 Implementation Notes

- **ROS Node/Nodelet**: Implement as a standard ROS node or a nodelet for improved performance.
- **Topics**:
  - Input: `/mbuggy/os1/points`
  - Output (filtered): `/filtered_points`
  - Output (removed): `/removed_points`

4 Data Fusion for Improved Filtering

While the basic approach above can suffice, additional sensors significantly improve robustness and reduce false classifications.

4.1 IMU Integration

- **Orientation-Based Prior**: Use roll/pitch estimates from the IMU to initialize plane segmentation, making ground detection more accurate on slopes.
- **Motion Compensation**: IMU data can help correct for vehicle motion during LiDAR scans.

4.2 Multi-LiDAR Fusion

- **Coverage**: The three Ouster LiDARs (`/mbuggy/os1`, `/mbuggy/os2`, `/mbuggy/os3`) likely have overlapping or distinct fields of view. Merging their data can reduce blind spots.
- **Consistency Check**: A point labeled as ground in one LiDAR but not observed in another (due to occlusion or noise) might need re-checking.

4.3 Camera Integration

- **Semantic Segmentation**: A CNN can classify image pixels as "road/ground" or "non-ground". LiDAR points projected into camera frames can adopt these classifications.
- **Color/Texture Cues**: Helps distinguish road from vegetation or other near-ground clutter, reducing confusion for purely geometry-based methods.

5 Conclusion

By starting with a straightforward LiDAR-based plane segmentation and outlier removal, we can effectively remove ground and noise. We then enhance performance by:

1. Fusing IMU data (orientation, motion compensation).
2. Combining multiple LiDARs for broader coverage.
3. Using camera-based segmentation or additional cues to reduce misclassification.

This multi-sensor approach ensures a robust solution for ground filtering, minimizing erroneous removals of critical obstacles and further improving overall perception quality for autonomous driving tasks.