
Home Assignment - Spike Detection Node in ROS 2

Project: Handheld Mobile Scanner – Sensor Trajectory Analysis

Estimated time: ~3 hours

Task Summary

Implement a ROS 2 package named `spike_detector` that detects “spikes” in a 3D trajectory stream and publishes each spike as soon as it is detected.

A **spike** is defined as a **sharp, brief drop in the Z-axis**, followed by a quick return to approximately the previous level. These spikes reflect real trajectory analysis.

Feel free to use any AI assistance (such as ChatGPT), but know your code and logic behind it. Rosbag will be provided in .mcap format with all relevant topics.

Functional Requirements

Your Node (`spike_detector_node`):

1. **Subscribe** to a trajectory topic:
 - Topic: `/state_optimizer/odometry_corrected`
 - Message type: `nav_msgs/msg/Odometry`
2. **Detect spikes** in the Z direction:
 - Implement your logic to detect spikes - for example:
Sliding window, z drop threshold, etc.
3. **Publish each spike** as soon as it is detected:
 - Topic: `/spike`
 - Message type: `determined by you.`
 - Message content: timestamp and x, y, z coordinates of the spike.

Parameters

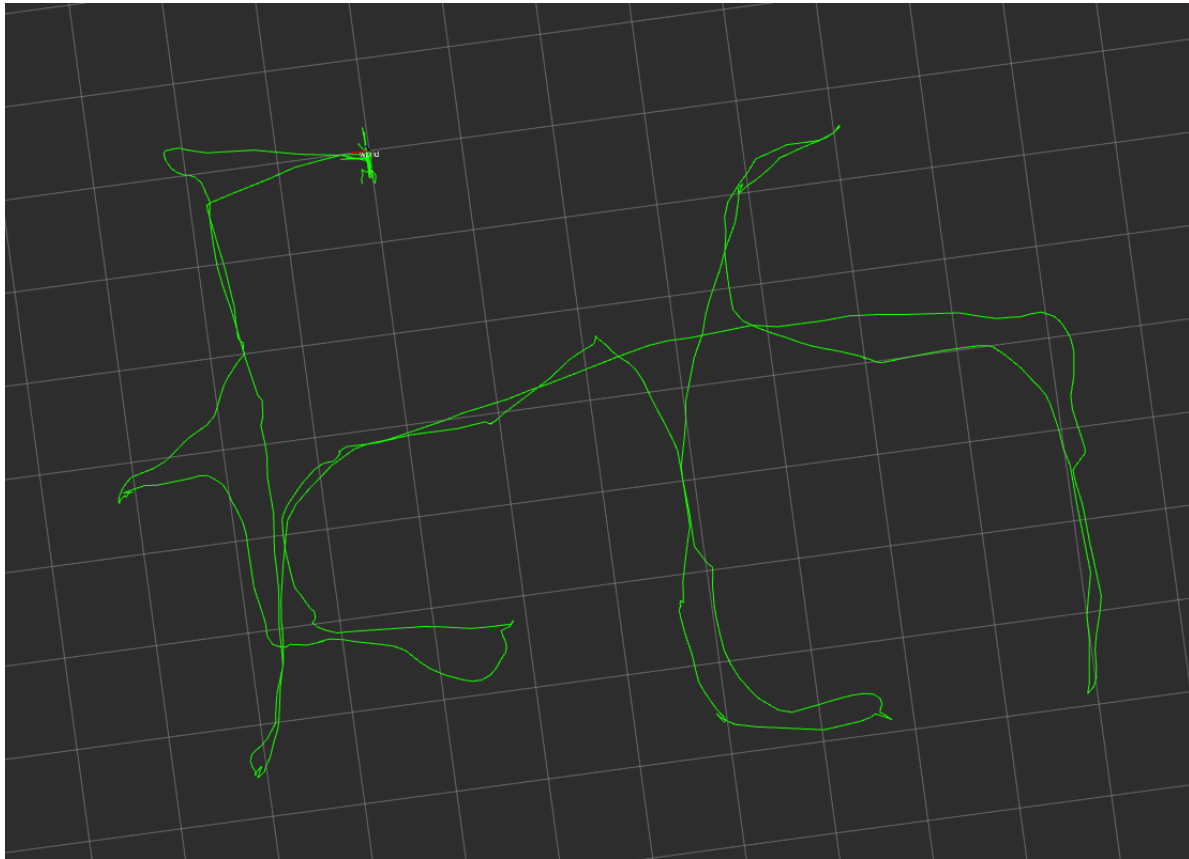
Expose any used parameters via launch or YAML config, for example:

- `z_drop_threshold` (default: 0.1)

Deliverables

1. A ROS 2 package named `spike_detector` containing:
 - Source code (Python or C++)
 - A launch file
 - A sample YAML config file
 2. A README file with:
 - How to build and launch the node
 - Short description of your detection logic
-

Trajectory - top view:



Spike - isometric view:

