Elevation Mapping: Discrete Resolution Simultaneous Elevation Mapping and Localization

Boxiang Fu[1†], Joshua Pen[2†], Ricky Yuan[3†], and Tianzhi Li[4†]

***Abstract*— For autonomous mobile robots operating in a confined static environment of established map dimensions, occupancy grid maps provide a simple and computationally-efficient method for mapping the environment. However, such maps restrictively represent each cell using a binary classifier as being either occupied or free. Furthermore, the algorithm requires localization as an input and cannot both map and localize simultaneously. We extend Moravec and Elfes' original occupancy grid algorithm by extending the classifier from a binary state (occupied/free) to a continuous state (elevation of cell). Furthermore, we estimate the camera pose using visual tracking so both localization and mapping can be simultaneously achieved. Finally, we implemented the algorithm in ROS (rclcpp) so that real-time performance can be attained.**

I. INTRODUCTION

TODO: Ricky

II. RELATED WORK

TODO: Tianzhi

III. METHODOLOGY

A. *Localization*

TODO: Ricky

B. *Bayes Filter*

Whenever a cell receives new elevation information, a Bayes filter function is called to fuse the original elevation with the new elevation information. This is done using a Kalman filter in one dimension. The mathematical derivation for the filter can be found in Reference [1], with a graphical depiction shown in Figure 1.

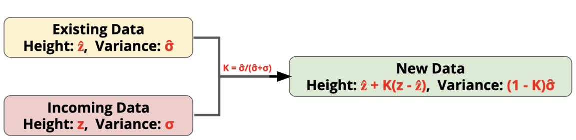

Fig. 1. Kalman Filter in One Dimension

[1]boxiangf@cs.cmu.edu
[2]jpen@cs.cmu.edu
[3]rickyy@andrew.cmu.edu
[4]tianzhil@cs.cmu.edu
[†]Robotics Institute, Carnegie Mellon University

C. *Fusion*

The algorithm was implemented in ROS using `rclcpp` to attain real-time performance requirements. The entire cyberphysical architecture is depicted in Figure 2. The ZED 2i stereo camera operates at 30 Hz and relays its camera feed at this frequency. This feed is passed through the ZED ROS wrapper packages to enable compatibility with ROS.

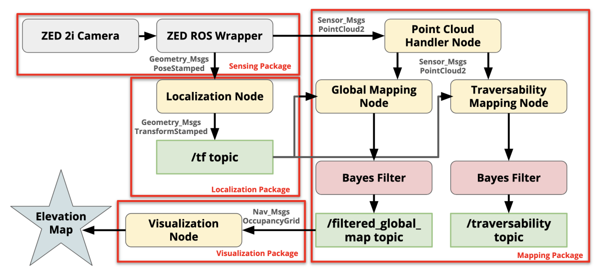

Fig. 2. Cyberphysical Stack of Elevation Mapping Algorithm

At each timestep, the wrapper calls both the `Localization` node and the `Point Cloud Handler` node. The `Localization` node parses the ROS wrapper information and publishes to the `/tf` topic on the extrinsic homogeneous transform between the map frame and the camera frame. The `Point Cloud Handler` node parses the raw point cloud from the ZED camera and filters it to remove any noise and/or outliers. The filtered point cloud is then re-published to be used.

Next, the `Global Mapping` node and the `Traversability Mapping` node subscribes to the filtered point cloud topic and the `/tf` topic. Using the extrinsic homogeneous transform between the map frame and the camera frame, the filtered point cloud is transformed from the camera frame to the map frame. For every grid cell in the map, we average all the point cloud elevation datapoints that fall inside the cell. This new averaged elevation is then sent to the Bayes filter function to be fused with the original elevation information. Once fusion is complete, the elevation map is published to the `/filtered_global_map` topic and the original binary occupancy map is published to the `/traversability` topic. The `Visualization` node subscribes to the `/filtered_global_map` topic and processes the elevation map for visualization in RViz.

IV. CODE

The GitHub repository for the ROS implementation is available here:

https://github.com/CMU-SLAM25/Elevation-Mapping

V. DATASET

TODO: Tianzhi

VI. EXPERIMENT

TODO: Josh

VII. RESULTS

TODO: Tianzhi

VIII. CHALLENGES

TODO: Ricky

IX. FUTURE WORK

TODO: Tianzhi

X. CONCLUSION

TODO: Josh

XI. REFERENCES

[1] Alex Becker. *Kalman Filter in One Dimension*. Accessed: 2025-04-20. 2025. URL: https://www.kalmanfilter.net/kalman1d.html.