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 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

Accurate and efficient mapping of real-world environments is critical for a wide range of robotic applications. Traditional mapping methods, such as occupancy grid maps, are widely used for their simplicity but are limited in their ability to represent continuous variations in terrain. In scenarios such as landscape reconstruction, terrain-aware path planning, or inspection tasks, robots need to understand not just whether a space is occupied, but how the terrain varies in elevation. Applications could include using UAVs for creating detailed maps of natural landscapes, supporting environmental monitoring, or assisting in search and rescue by providing real-time elevation maps of affected areas.

In this work, we propose a method that extends the conventional 2D occupancy grid map to the 2.5D elevation map, which is able to capture the vertical dimension of the environment. Our system uses a stereo camera called ZED 2i, which is paired with a depth sensor to accurately estimate 3D point clouds and the camera's 6-DoF pose for each frame. By filtering these point clouds to remove noise and outliers, we extract meaningful elevation data that can be used to construct the elevation map. Furthermore, the algorithm is computationally efficient and implemented in ROS2 with C++, making it suitable for real-time applications.

In summary, we developed a fast and accurate elevation mapping framework that combines dense 3D point cloud processing with real-time visual localization. This system enables autonomous robots to build detailed elevation maps of their surroundings while tracking their own pose.

[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

II. RELATED WORK

Occupancy grid mapping was first introduced by Moravec and Elfes [3], offering a probabilistic framework for environment representation in mobile robotics. Their method discretizes space into a grid where each cell contains the probability of being occupied or free, providing a computationally efficient and robust mapping solution. This framework has been widely adopted in robotic systems due to its simplicity and compatibility with various sensor modalities.

However, the binary nature of traditional occupancy grids imposes a limitation when modeling complex or uneven environments. To address this, elevation maps have been proposed [4], extending occupancy grids by encoding each cell with height information. These maps are especially useful for ground robots navigating 3D terrains, allowing for obstacle detection and path planning in non-flat environments. Our work builds on this concept by integrating elevation as a continuous state, rather than a binary or probabilistic classification.

Simultaneous Localization and Mapping (SLAM) is another key component in autonomous navigation. Classic occupancy grid mapping assumes perfect localization input, which is often unrealistic. SLAM algorithms [2, 5] address this by jointly estimating both the map and the robot pose. Visual SLAM approaches, such as ORB-SLAM [6], utilize camera data for localization and mapping in real time. Inspired by these methods, our system employs visual tracking to estimate the camera pose, enabling real-time elevation mapping without relying on external localization.

Finally, our implementation in ROS using `rclcpp` aligns with a broader trend of modular, open-source robotic development frameworks [7]. This allows for seamless integration with other ROS-based tools and facilitates real-time performance, which is essential for deployment in practical robotics applications.

III. METHODOLOGY

A. Localization

In our system, localization is achieved by fusing visual and inertial data obtained from the ZED 2i stereo camera. The ZED 2i is equipped with an RGB stereo camera, an Inertial Measurement Unit (IMU), and GNSS; however, in our experiments, we utilize only the IMU and RGB images to estimate the camera's 6-DoF pose.

We employ a Visual-Inertial Odometry (VIO) approach provided by the ZED SDK, where visual data from the stereo camera provides depth and feature correspondence between frames, while the IMU supplies acceleration and angular velocity measurements. The fusion of these data sources allows for more robust and accurate pose estimation, especially in conditions where visual tracking alone might degrade, such as in low-texture environments or during rapid motion.

By fusing stereo vision and IMU, we accurately estimate the camera's 6-DoF pose in real-time without GNSS, ensuring consistent elevation mapping and frame alignment.

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.

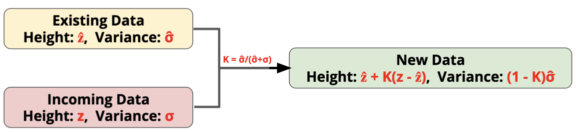


Fig. 1: Kalman Filter in one dimension

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.

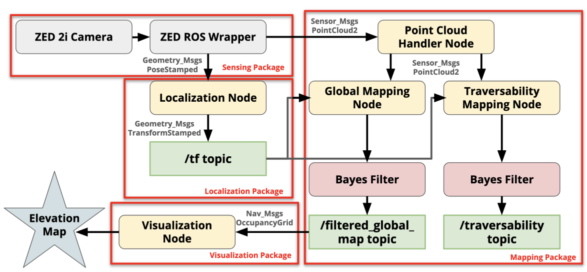


Fig. 2: Cyberphysical stack of elevation mapping algorithm

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.

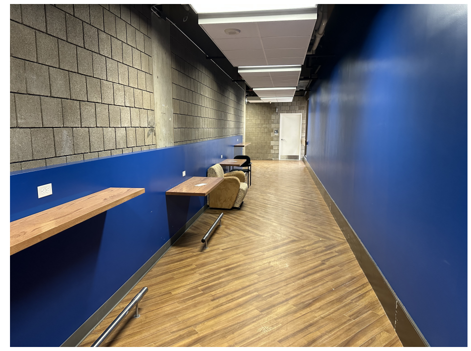


Fig. 3: Environment 1 – Straight hallway with tables and chairs along the left side

IV. CODE

The GitHub repository for our ROS implementation is available here:

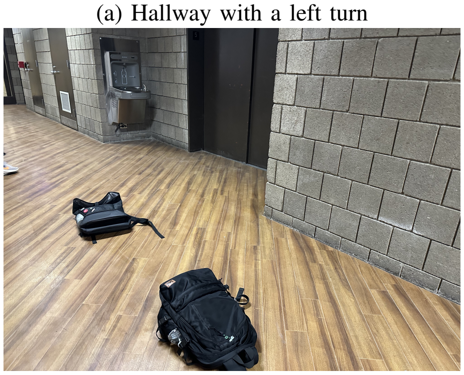
`https://github.com/CMU-SLAM25/Elevation-Mapping`

V. DATASET

To support the development and evaluation of our elevation mapping system, we collected a real-time dataset using an RGB-D camera at 30 frames per second. The dataset was gathered on the 5th floor of Wean Hall as well as Newell-Simon Hall 1305 and covers three distinct indoor environments representative of typical navigation scenarios.

- **Environment 1: Straight hallway**
  As shown in Figure 3, a long corridor with tables and chairs placed along the left side. This scene provides a structured layout with moderate clutter.
- **Environment 2 Hallway with a left turn**
  As shown in Figure 4, a hallway that includes a left turn, two bags placed on the floor, and an elevator on the right. This environment introduces fine-grained elements and occlusions for testing robustness.
- **Environment 3: Lecture hall with rows of tables and chairs**
  As shown in Figure 5, a lecture room containing multiple rows of tables and chairs. This environment provides a more complex and densely populated layout.

(a) Hallway with a left turn



(b) Hallway with two bags on the floor and an elevator on the right

Fig. 4: Environment 2 – Hallway with a left turn with two bags on the floor and an elevator on the right

Fig. 5: Environment 3 – Lecture hall with rows of tables and chairs

VI. EXPERIMENT

To evaluate the performance of the elevation mapping system, experiments were conducted in indoor environments at Carnegie Mellon University. The first environment was a straight hallway on the 5th floor of Wean Hall with tables and chairs placed along the left side, providing a structured scene with moderate static obstacles. The second environment was also on the 5th floor of Wean Hall and consisted of a hallway with a left turn, two bags placed on the floor, and an elevator on the right side, introducing localized elevation variations and occlusions. In addition, we conducted a third experiment in Room 1305 of Newell-Simon Hall, a lecture hall with rows of tables and chairs organized in a classroom format. During each trial, the robot traversed the environment while capturing synchronized stereo images and point cloud data using a ZED 2i stereo camera. Pose estimation and elevation mapping were performed in real time using the developed ROS framework. The resulting elevation maps were compared qualitatively to the known layouts to assess the system's ability to reconstruct both large-scale geometry and small object-level features.

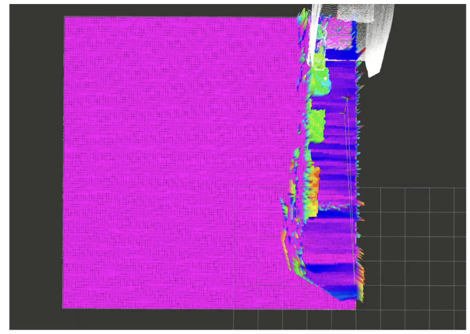


Fig. 6: The generated elevation map for Environment 1

VII. RESULTS

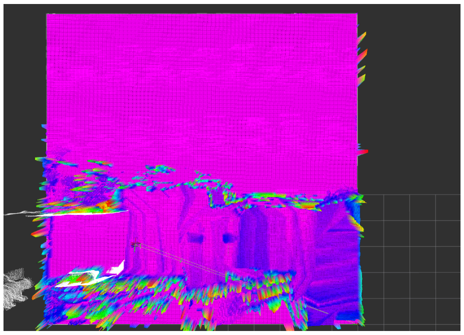
Our elevation mapping system successfully captured the structural features of both testing environments in Wean Hall. The generated elevation maps reflect the geometry and elevation changes in the scene with high fidelity, validating the effectiveness of our approach.

In **Environment 1** (the straight hallway), as shown in Figure 6, the map clearly identifies the structure of the straight hallway and the positions of the tables and chairs placed along the left side. These objects appear as distinct elevated regions relative to the flat hallway floor, demonstrating the map's ability to represent static obstacles with elevation differences.

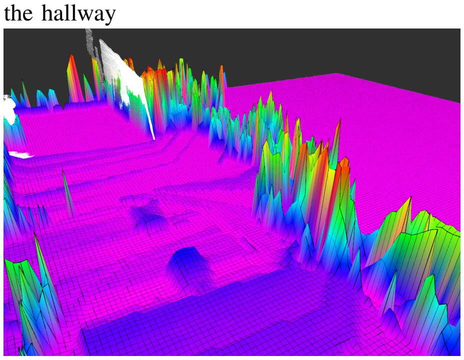
In **Environment 2** (the hallway with a left turn), as shown in Figure 7, the elevation map correctly captures the left turn in the hallway, enabling the layout of the hallway to be reconstructed. Additionally, two localized bumps detected in the elevation map correspond to the bags placed on the floor, while a recessed region in the wall aligns with the location of the elevator. These details confirm that the system is capable of mapping fine-grained features in indoor environments.

In contrast, results in **Environment 3** (the lecture hall), as shown in Figure 8, show significant limitations. The elevation map exhibits high noise levels and suffers from noticeable drift. The repetitive layout of rows of tables and chairs, combined with limited visual texture, likely contributed to degraded tracking performance and accumulated localization error. This highlights the challenges of operating in densely populated, visually ambiguous environments and suggests a need for improved robustness in pose estimation under such conditions.

Overall, the system performs well in structured and moderately cluttered spaces, but its performance degrades in highly cluttered environments, indicating important areas for future improvement.



(a) The elevation map captured the left turn in the hallway



(b) The elevation map captured two bags and the elevator on the right side

Fig. 7: The generated elevation map for Environment 2

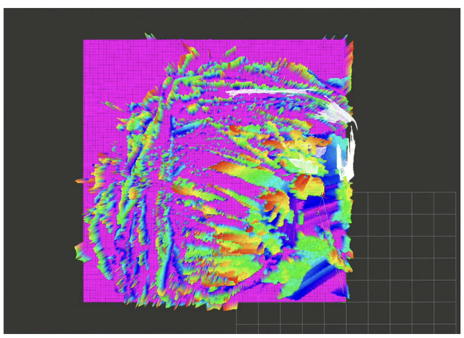


Fig. 8: The generated elevation map for Environment 3

VIII. CHALLENGES

One of the main challenges in our project was accurate localization. Initially, we used a RealSense camera as our sensor, obtaining point clouds from the camera's built-in processing of RGB and depth images. We manually computed surface normals from these point clouds and implemented the Iterative Closest Point (ICP) algorithm in C++ for pose estimation. However, the results were unstable, with frequent issues such as pose drift and flickering, causing the estimated camera pose to deviate significantly from the ground truth.

We suspected two possible reasons for the results. First, the quality of the point clouds was highly dependent on the visual richness of the environment. In low-feature areas, such as flat walls, floors, or smooth surfaces, the point cloud became ambiguous, and reduce the reliability of the ICP alignment. Second, our custom implementation of ICP may have suffered from suboptimal parameter settings, i.e. insufficient iterations or inappropriate distance thresholds.

To address these issues, we switched to using the ZED 2i stereo camera as mentioned, which offers more robust and accurate depth sensing. In addition to stereo depth, the ZED 2i also integrates an IMU, allowing us to leverage visual-inertial fusion for improved pose estimation. By combining stereo-based point cloud generation with high-frequency inertial data, we achieved more stable and accurate 6-DoF localization. In our experiments, the visual-inertial odometry (VIO) provided by ZED's SDK proved to be more reliable and stable than our initial ICP-based approach.

IX. FUTURE WORK

While the current elevation mapping system provides a strong foundation for terrain representation and navigation, several directions remain open for future improvement:

1) **Improving Robustness in Cluttered Environments**
In cluttered or dynamic scenes, traditional elevation mapping approaches can struggle due to noisy or unstable feature matches during pose estimation. To address this, we propose incorporating semantic segmentation to identify and prioritize reliable, static structures such as ground, walls, and permanent obstacles. Additionally, filtering out unstable or transient features—such as vegetation or moving objects—during the pose estimation step can lead to more stable map updates and improved overall robustness.
2) **Extending to Full 3D Reconstruction**
Standard elevation maps are inherently limited to a 2.5D representation, which cannot accurately model overhangs, bridges, or vertical structures. To overcome this, we aim to extend the elevation mapping framework to support full 3D reconstruction. This could involve integrating volumetric mapping techniques such as truncated signed distance fields (TSDFs) or surfel-based representations to capture the full geometric complexity of the environment.
3) **Integrating Dynamic Obstacle Detection**
For real-world deployment, especially in environments shared with humans or moving vehicles, the system must be able to detect and react to dynamic obstacles. Future work will focus on integrating real-time dynamic obstacle detection into the mapping pipeline. This would enable the elevation map to support more advanced navigation tasks, such as trajectory replanning or collision avoidance, making the system suitable for practical autonomous navigation in dynamic, unstructured settings.

X. CONCLUSION

This project extends traditional occupancy grid mapping by incorporating continuous elevation estimation and simultaneous localization through visual-inertial odometry. The system was implemented in ROS and demonstrated the ability to reconstruct indoor environments with high structural fidelity in real time. Experiments conducted in structured hallway environments showed that the system could accurately capture both the overall layout and smaller obstacles, such as tables, chairs, and bags. In more cluttered environments, such as a lecture hall in Newell-Simon Hall, the system faced challenges with mapping accuracy and localization stability due to dense object arrangements and noisy sensor measurements. These results demonstrate that the proposed elevation mapping system is effective in structured spaces and highlight its current limitations in highly cluttered indoor environments.

XI. REFERENCES

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

[2] Hugh Durrant-Whyte and Tim Bailey. "Simultaneous localization and mapping: part I". In: *IEEE robotics & automation magazine* 13.2 (2006), pp. 99–110.

[3] Alberto Elfes. "Using occupancy grids for mobile robot perception and navigation". In: *Computer* 22.6 (1989), pp. 46–57.

[4] Péter Fankhauser et al. "Robot-centric elevation mapping with uncertainty estimates". In: *Mobile Service Robotics*. World Scientific, 2014, pp. 433–440.

[5] Giorgio Grisetti, Cyrill Stachniss, and Wolfram Burgard. "Improved techniques for grid mapping with rao-blackwellized particle filters". In: *IEEE transactions on Robotics* 23.1 (2007), pp. 34–46.

[6] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. "ORB-SLAM: A versatile and accurate monocular SLAM system". In: *IEEE transactions on robotics* 31.5 (2015), pp. 1147–1163.

[7] Morgan Quigley et al. "ROS: an open-source Robot Operating System". In: *ICRA Workshop on Open Source Software*. 2009.