CREATING A 2D OCCUPANCY GRID USING OVERHEAD CAMERAS

Starlet Ben Alex, Alphonsa Abraham, Navneeth Krishnan, Sidharth V Menon, Vyshnavi Dipu
Dept. of Electronics Engineering
Saintgits Group of Institutions, Kottayam, Kerala India

***Abstract*—Abstract: This project explores the development and implementation of a 2D occupancy grid mapping system using overhead cameras. The primary goal is to enhance envi ronmental perception and navigation capabilities for autonomous systems in indoor set tings. Overhead cameras provide a continuous and wide field of view, capturing dynamic changes in the environment. The system processes the video feed to generate real-time oc cupancy grids, representing the presence or absence of obstacles within a predefined grid space. Key components include transformation techniques to map camera coordinates to grid coordinates, and real-time updating mechanisms for the occupancy grid. The effec tiveness of the approach is validated through a series of experiments in controlled environ ments, demonstrating its accuracy, responsiveness, and potential applications in robotics, security, and smart infrastructure management. This method offers a cost-effective and scalable solution for dynamic environment monitoring, contributing to advancements in autonomous navigation and intelligent system design. Keywords: 2D occupancy grid, overhead cameras,environmental perception,autonomous systems**

I. INTRODUCTION

The emergence of intelligent infrastructure and autonomous systems has made so phisticated techniques for navigation and environmental awareness necessary. The 2D occupancy grid mapping system is one such technique that makes use of overhead cameras to track and map changing indoor settings. The goal of this project is to put such a system into operation using Gazebo, a potent robotics simulation tool, and the ROS2 (Robot Operating System 2) framework. Areliable and adaptable framework for creating and integrating intricate robotic systems is offered by ROS2. Its support for modularity, scalability, and real-time connectivity makes it the perfect option for occupancy grid mapping.

When ROS2 and Gazebo are used together, the occupancy grid mapping system may be thoroughly tested and validated prior deployment in real-world scenarios The first step in the implementation process is to set up the simulation environment in Gazebo. This involves modeling an indoor virtual space and adding both static and dynamic obstacles. The surroundings is continuously captured on video feeds by simulating overhead cameras. After that, these feeds are analyzed by ROS2 nodes, which carry out image processing functions creating a stitched image from 4 overhead cameras and converting into a Occupancy Grid Map. A two-dimensional occupancy grid is mapped each 1s to keep it continuously updated.

II. RELATED WORKS

III. Summary of Papers for 2D Occupancy Grid Creation from Overhead Camera Systems

Gazebo, as described in "Design and use paradigms for Gazebo, an open-source multi-robot simulator" [1], provides a robust platform for simulating the environment and overhead camera setup. This simulation is crucial for generating synthetic camera data, which serves as a valuable resource for developing and testing image processing and occupancy grid generation algorithms. Furthermore, Gazebo's ability to simulate real-world factors, such as noise and lighting variations, enhances the robustness of the system. It also allows for the simulation of robot movement, enabling the testing of how the occupancy grid updates as the robot navigates, providing a safe and controlled environment for development.

The Robot Operating System (ROS), as detailed in "Programming Robots with ROS: A Practical Introduction to the Robot Operating System" [2], offers the middleware necessary for integrating all components of the system. ROS facilitates the use of nodes for handling each step of the process, from camera input to occupancy grid creation and robot navigation. The message passing capabilities of ROS enable seamless data exchange between nodes. Additionally, ROS provides essential tools for visualization (RViz) and debugging, streamlining the development and testing process.

To extract meaningful information from the overhead camera images, feature detection algorithms such as SIFT, as presented in "Distinctive image features from scale-invariant keypoints" [3], are employed. SIFT, or similar algorithms like SURF or ORB, helps in identifying and tracking objects or landmarks within the images. These features are instrumental in determining the position and orientation of objects, identifying obstacles, and potentially correcting for camera perspective distortions. Moreover, they aid in generating the occupancy grid by distinguishing between static and moving objects.

The core concept of representing the environment as an occupancy grid is thoroughly discussed in "Using occupancy grids for mobile robot perception and navigation" [4]. Occupancy grids provide a discrete, probabilistic representation of the environment, where each cell indicates the probability of an obstacle being present. This paper lays the theoretical foundation for creating and updating occupancy grids from sensor data, specifically camera data in this case. It also provides the mathematical basis for updating cells based on new sensor input, which is essential for accurate and dynamic environment representation.

Visualizing the occupancy grid and robot movement in real-time is made possible by RViz, a 3D visualization tool for ROS, as described in ”RViz: A 3D visualizer for ROS” [5]. RViz allows for the display of the grid as 2D or 3D representations, overlaying it on camera images. It also facilitates the visualization of the robot’s position and path. This tool is invaluable for monitoring and debugging the system, providing a clear understanding of the data generated by the occupancy grid.

IV. Integration of Concepts

The integration of these concepts involves several key steps. First, Gazebo is used to simulate the environment and camera setup, generating training and testing data. Next, ROS integrates all components through nodes and message passing. Feature detection algorithms like SIFT extract relevant features from camera images to identify objects and obstacles. These features and camera data are then used to populate and update the occupancy grid. Finally, RViz visualizes the grid and robot movement in real-time, aiding in monitoring and debugging.

V. PROPOSED METHOD

In this we have main requirements such as a 3D model of an environment with Overhead cameras attached, ROS2 workspace that has turtlebot3, Rviz and Gazebo installed to simu late and access the environment. To implement these the following step were used:

1) **Launching the environment:** With the required packages installed we are able to deploy and interact with 4 overhead cameras provided by the Industry Mentor and view the topics of these cameras from which the camera feed can be accessed. Launching the environment gives access the the camera feed topics:
2) **Collecting Camera Feed:** This step requires the creation of a ROS2 package that has access to the ROS2 topics and can acquire images using the pub-sub model used by ROS2.
3) **Image Stitching**: We have obtained images from 4 different cameras at 4 different angles, for the preparation of an Occupancy Grid we need to stitch these distinct images together to form a complete top down image that can be mapped.
4) **Color image to Binary image:** To convert an image to an occupancy grid it needs to be a Binary image filled with binary values. This is done using OpenCV’s image manipulation function.
5) **Generating a 2D Occupancy Grid:** A 2D occupancy grid is generated using ROS2 package OccupancyGrid. This package uses the information in the Binary image to generate a Occupancy grid of the environment in the image. This published occupancy grid topic is then used by the mapsaver package to save the Occupancy grid as a ’.pgm’ file.
6) **Occupancy Grid Evaluation:** The Occupancy grid that is generated is then opened in Rviz and we measure the key distances that have been prescribed for evaluation.

References

[1] A. Koenig and N. Howard, ”Design and use paradigms for Gazebo, an open-source multi-robot simulator,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2004, vol. 3, pp. 2149-2154.

[2] M. Quigley, B. Gerkey, and W. Smart, *Programming Robots with ROS: A Practical Introduction to the Robot Operating System*, O’Reilly Media, Inc., 2015.

[3] D. G. Lowe, ”Distinctive image features from scale-invariant keypoints,” *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91-110, 2004.

[4] A. Elfes, ”Using occupancy grids for mobile robot perception and navigation,” *Computer*, vol. 22, no. 6, pp. 46-57, 1989.

[5] W. Morris, ”RViz: A 3D visualizer for ROS,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA) Workshop on Open Source Software*, 2010.