Multi-Robot System for Mapping and Localization

Glace Varghese T[1], Thambbireddygare G. C. Reddy[1], Anjana K Menon[1], Ajay Paul[2], Sreeja Kochuvila[1], Varma Divya R[2], Ravichandra Bhat[2] and Navin Kumar[1]
[1]Department of Electronics and Communication Engineering, Amrita Vishwa Vidyapeetham, Bangalore, India
[2]Robotics - SX/PJ1-EM, Bosch Global Software Technologies, Bengaluru, India
anjanamenon1203@gmail.com, goweshreereddyt@gmail.com, glacevargheset96@gmail.com, ajay.paul@in.bosh, k_sreeja@blr.amrita.edu, divya.radhakrishnavarma@in.bosch.com, ravichandra.bhat@in.bosch.com, navinkumar@ieee.org

2023 8th International Conference on Robotics and Automation Engineering (ICRAE) | 979-8-3503-2765-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICRAE59816.2023.10458504

***Abstract*—In this work, we developed algorithm for coordinated multi-robot system. It is developed for simultaneous localization and mapping (SLAM). Many novel applications of multi-robot system especially, in the industrial indoor environment and complex/dangerous environment are being envisioned. The proposed algorithm focuses on optimizing navigation and execution of tasks within an environment. We used TurtleBot3 Waffle model and robot operating system (ROS) platform and simulated in Gazebo for localization, navigation and comprehensive coverage. The algorithm achieves effective mapping, localization, navigation, and coverage. Algorithm also ensures seamless coordination among multiple robots within a unified transform-tree (tf-tree) facilitating synchronized movement and a holistic understanding of the environment. Furthermore, an innovative coverage path planning and submap division is achieved using this technique. It is expected that such development would lead to practical applications in the real environment.**

***Keywords*—*Gazebo World, TurtleBot3 Waffle, Simultaneous Localization and Mapping (SLAM), Grid-based FastSLAM Mapping (Gmapping), Map Segmentation, ROS Visualization (RViz), Adaptive Monte Carlo Localization (AMCL), Multi-Robot Coordination.*

I. INTRODUCTION

Localization and mapping, better called as simultaneous localization and mapping (SLAM) [1] is a technique of estimating a map of the environment. In addition, simultaneously it should localize itself that is, where is it in the environment. It becomes very important specifically for robots whenever they move into unknown or partially known environments. Map information then can be used for many other tasks like obstacle avoidance, route planning and so on. This technology is found very useful in the industrial set up for supply chain, cleaning, clearing and many others. Similarly, it is of high importance in difficult area or dangerous terrain. In the absence of SLAM technology, they are not able to map the environment and localize themselves. Parking a self-driving car in an empty space or delivering packages by navigating robots in an unknown environment are some of the other important applications. With Industry 4.0 evolution, the SLAM technology is getting lot of attention in manufacturing, supply chain, warehouse in most of the industry verticals.

Complexity of the SLAM increases when multi robot [2-3] is put into action. A multi-robot system refers to a group or collection of autonomous robots that work together to accomplish a common goals. It is required because activity cannot be completed in time or as scheduled using a single robot. Thus, much other functionality such as coordinating among them, dividing the tasks and areas, scheduling the tasks, communicating to each other, and so on would be required. These systems are designed to leverage the benefits of cooperation and coordination among multiple robots, enabling them to perform tasks more efficiently and effectively than a single robot operating alone. Multi-robot systems offer a viable alternative to single robots as they exhibit enhanced efficiency, flexibility, and fault tolerance. These systems are gaining popularity across various domains due to the advantages they provide over single robot systems. By distributing the workload among multiple robots, tasks can be completed faster and more efficiently [2]. This advantage is particularly significant in complex or time-consuming applications. Moreover, multi-robot systems offer redundancy, meaning that if one robot fails, the remaining robots can continue performing the task, thereby enhancing system reliability. Another benefit of multi-robot systems is their scalability. They can be easily scaled up or down to meet the requirements of specific tasks, making them suitable for a wide range of applications. Furthermore, multi-robot systems are highly flexible and can be programmed to collaborate in various configurations and undertake diverse tasks, providing greater adaptability and versatility. Finally, multi-robot systems excel at collaborative work towards a shared objective, making them well-suited for scenarios where multiple tasks need simultaneous execution. Overall, the advantages of multi-robot systems make them invaluable in industries such as manufacturing, logistics, healthcare, agriculture, and more. One crucial aspect of multi-robot systems is the ability to accurately map the environment and localize the robots within it and navigate to the desired location. Mapping and localization play a fundamental role in enabling robots to navigate, collaborate, and perform tasks effectively in complex and dynamic environments. Few studies are reported in literature for various SLAM algorithms.

In [1], the authors introduced a simulation environment in robotic operating system (ROS) and Gazebo to test mobile robots, allowing them to autonomously generate maps of the environment and navigate within it. They employed the extended Kalman filter [4] and particle filter for exploration, followed by finding the shortest path for traversal. Additionally, other coverage path planning algorithms such as

Dijkstra's and A* were incorporated in [5] to determine the shortest route to a destination. The robots use different sensors to localize themselves and plan the path efficiently by avoiding obstacles [6-7]. Unlike previous work that focused on single robot coverage, [8] delved into multi-robot coverage algorithms, specifically utilizing the deterministic mark and cover (MAC) algorithm. This approach involved employing short-lived navigational markers for navigation and indirect communication, presenting a promising solution to the coverage problem in multi-robot systems. Vision sensors can also be used in addition to localize and traverse the path effectively [9].

In this work, we have successfully developed a novel coordinated multi-robot system tailored for indoor applications. It focuses on optimizing navigation and task execution within a simulated environment, utilizing the TurtleBot3 Waffle model and Gazebo platform. By integrating various algorithms including SLAM, Adaptive Monte Carlo localization (AMCL) [10], navigation, and coverage path planning, the system achieves effective mapping, localization, navigation, and comprehensive coverage. The original contribution lies in the seamless coordination of multiple robots within a unified transform-tree (tf-tree), facilitating synchronized movement and a holistic understanding of the environment. Furthermore, the work introduces innovative coverage path planning algorithms and submap division, enhancing the system's efficiency. The proposed system leverages the capabilities of ROS, including its communication infrastructure and various packages, to facilitate the coordination and cooperation among multiple robots. Through extensive simulations, we demonstrate the system's effectiveness in achieving accurate mapping, localization, navigation and coverage path planning, paving the way for efficient and reliable multi-robot applications. This work serves as a strong foundation for future research and development in the realm of coordinated multi-robot systems for indoor applications.

Rest of the article is organized as follows. Section II describes the system model and architecture while in Section III, we presented the algorithms. Section IV contents the results which are discussed. Finally, Section V concludes the paper.

II. System Model and Architecture

Overall system architecture of SLAM is shown in Figure 1 It involves several components that work together to achieve accurate robot positioning and environment mapping. The localization and mapping process starts with sensor data collected from light detection and ranging (LiDAR) [11] and inertial measurement unit (IMU) sensors. These sensors provide measurements of the environment, including landmarks and features. The robot's LiDAR sensor scans the surrounding environment, emitting laser beams and measures the time it takes for the laser pulses to return after hitting objects in the environment. This generates a set of 2D or 3D point cloud data, representing the distances and angles to detected obstacles.

From the LiDAR point cloud data, distinctive features or landmarks are extracted. These features can include corners, edges, and other unique points in the environment that can be reliably detected and tracked over time [12]. Odometry data from wheel encoders or inertial sensors provide initial estimates of the robot's motion including position and orientation changes.

Using the associated LiDAR measurements and robot's motion information, the SLAM algorithm constructs and updates the map of the environment. The map can be represented in various forms, such as occupancy grids or point clouds, with each cell or point in the map representing a location or obstacle in the environment. Simultaneously with mapping, the SLAM algorithm estimates the robot's position and orientation (pose) within the constructed map. This is achieved by refining the initial pose estimate using the associated LiDAR measurements and the known positions of the extracted features in the map.

As the robot moves through the environment, it may revisit previously visited locations, leading to loop closures [13]. The SLAM algorithm detects these loop closures and uses them to correct any accumulated errors in the map and robot's pose. By all these, individual workload of each robot in the multi robot system is reduced by distributed work [14].

The final output of the localization and mapping process is an accurate map of the environment and an improved estimate of the robot's position and orientation. This information is crucial for the robot's navigation and path planning. It allows effective and qautonomous movement in the given environment while avoiding obstacles and enabling it to reach the specified destinations.

III. Algorithm Design and Description

Our focus is to create a simulated environment, performing simultaneous localization and mapping (SLAM) using the TurtleBot3 SLAM node and the Grid-based FastSLAM mapping (Gmapping) algorithm. Additionally, used autonomous navigation with the ROS navigation stack, map segmentation and, planning paths for full coverage. Full coverage path planner (FCPP) with the backtracking spiral algorithm is used for the coverage. Figure 2 illustrates the step-by-step algorithm for achieving autonomous exploration and mapping of an environment using multiple TurtleBot3 Waffle models in a Gazebo world. It involves the following major tasks.

A. Gazebo World Creation and TurtleBot3 Deployment

We begin by constructing a simulated environment using Gazebo, consisting of multiple rooms. Within this world, we deploy three TurtleBot3 Waffle models. These robots serve as mobile agents, responsible for exploring the environment and mapping the unknown areas.

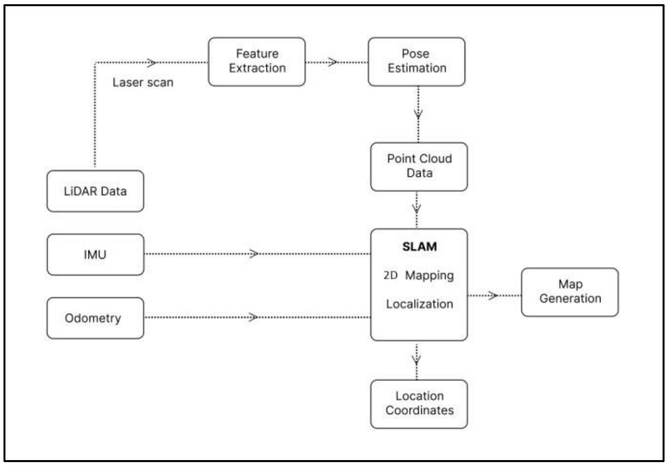

Figure 1: System Architecture of SLAM

B. SLAM with TurtleBot3 - Grid-based FastSLAM Mapping (Gmapping)

To create an accurate map of the Gazebo world, we employ the TurtleBot3 SLAM node. The TurtleBot3 SLAM node utilizes sensor data from the TurtleBot3 robots to achieve simultaneous updating of their positions (localization) while constructing a map of the environment. The SLAM process involves combining information from laser range finders and odometry data to estimate the robots' positions and orientations as they navigate through the environment. Specifically, we utilize the Grid-based FastSLAM mapping algorithm, an efficient Rao-Blackwellized particle filter. Gmapping leverages laser data to generate grid maps of the environment. As the TurtleBot3 robots move through the Gazebo world, the Gmapping algorithm processes the incoming sensor data and performs probabilistic calculations to update the robot's pose. Simultaneously it builds an accurate representation of the environment.

C. Visualization with RViz and Localization with AMCL

To visualize the 2D map constructed through Gmapping and to accurately determine the position of the TurtleBot3 robots within the environment, we use the ROS visualization tool RViz, in conjunction with the adaptive Monte Carlo localization (AMCL) algorithm. RViz enables us to interactively observe the map and the TurtleBot3 robots' positions within the simulated environment. AMCL is a probabilistic localization method that compares sensor data from laser range finders and/or depth sensors to the known map to estimate the robot's pose.

`

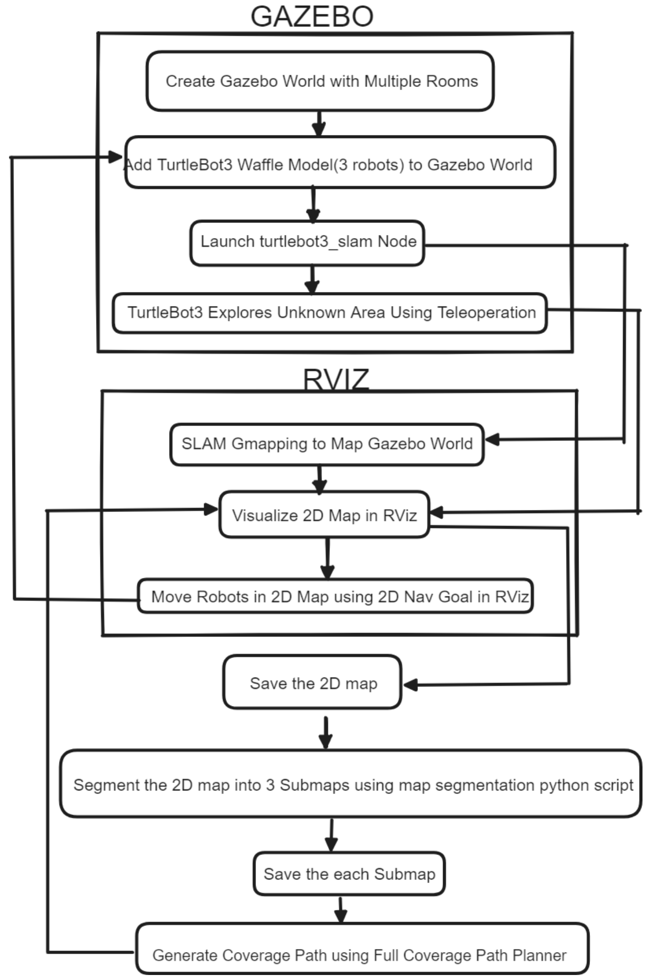

Figure 2: Process for generating Coverage Path

D. Visualization with RViz and Localization with AMCL

To visualize the 2D map constructed through Gmapping and to accurately determine the position of the TurtleBot3 robots within the environment, we use the ROS visualization tool RViz, in conjunction with the adaptive Monte Carlo localization (AMCL) algorithm. RViz enables us to interactively observe the map and the TurtleBot3 robots' positions within the simulated environment. AMCL is a probabilistic localization method that compares sensor data from laser range finders and/or depth sensors to the known map to estimate the robot's pose.

E. Autonomous Navigation with ROS Navigation Stack

To enable autonomous navigation, we utilize the ROS navigation stack, a comprehensive set of ROS packages that provide localization, mapping, path planning, and obstacle avoidance algorithms. Using the navigation stack, the TurtleBot3 robots can autonomously navigate towards 2D navigation goals specified in RViz, allowing them to execute planned exploration routes without tele-operation.

F. Map Segmentation and Submap Division

Once the environment has been thoroughly explored, we apply map segmentation techniques to divide the obtained map into several submaps. Map segmentation involves partitioning the map into non-overlapping sections, each assigned to a specific robot for exploration. This division is achieved using a python script specifically developed for this purpose. The script calculates the width and length of the environment based on the dimensions of the 2D map and determines the optimal way to partition the area into three approximately equal parts. By dividing the map into segments, each robot is assigned a specific segment to explore and map, ensuring comprehensive coverage of the environment and efficient area division among the multi-robot team.

G. Full Coverage Path Planner (FCPP) with Backtracking Spiral Algorithm

The full coverage path planner is a critical module in the ROS Melodic framework that generates paths for autonomous robots to achieve full coverage of a given area. It plays a crucial role in applications such as cleaning robots, surveillance systems, or agricultural robots, where complete coverage of the environment is required. The FCPP algorithm employs the Backtracking spiral algorithm (BSA) as one of its techniques for systematic exploration. The BSA starts from a central point and follows a spiral pattern expanding outward while exploring the area. It systematically navigates the environment and, when encountering obstacles or revisiting previously covered regions, it backtracks to the previous position and continues exploration in a different direction. This ensures comprehensive coverage of the entire area with minimal overlap, without missing any portions of the environment.

Thus, the algorithm enables efficient and accurate mapping and exploration of complex environments. The use of Gazebo and RViz facilitates simulation, visualization, and localization, providing valuable insights for further research and development in robotic exploration and mapping domains.

IV. Results and Discussion

The resulting outcomes from the steps mentioned in Section III are presented and discussed in this section. We have performed mapping localization and navigation in a simulated building environment depicted in Figure 3 It represents a Gazebo world with multiple rooms and three robots created using the building editor tool. This virtual space aimed to simulate an environment for real world applications.

Figure 4 presents a two-dimensional map of the environment generated using laser scan data and the SLAM algorithm. The map visually represents the environment, including open areas and obstacles. To create the map, laser scan data is collected by a sensor mounted on the robot as it navigates through the environment. The SLAM algorithm processes this sensor data, combining it with information from the robot's odometry to simultaneously estimate the robot's position and map the surrounding environment.

With the help of the 2D map, the ROS navigation stack generates trajectories and velocity commands for the robots to avoid obstacles and to safely navigate towards their intended destination locations efficiently.

In Figure 5, the initial positions of the three robots are depicted, along with their movement towards the designated destination locations. In Figure 6, the robots utilize the ROS navigation stack to achieve these tasks.

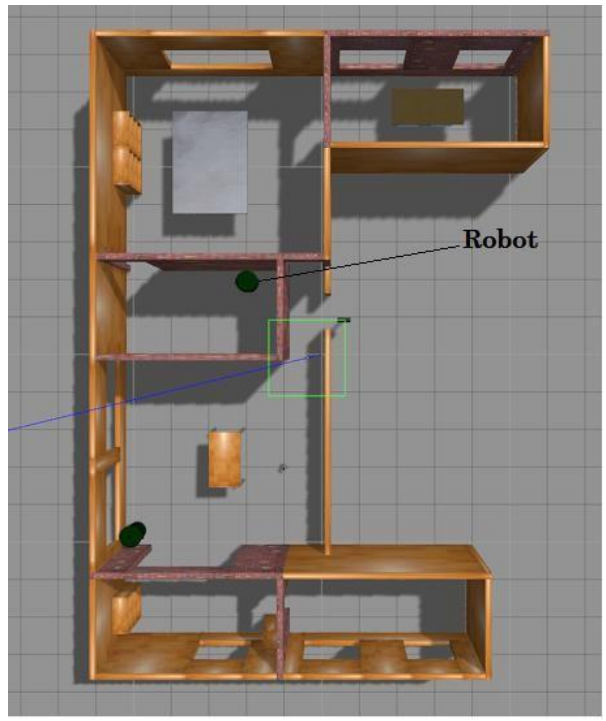

Figure 3: Gazebo World (created environment)

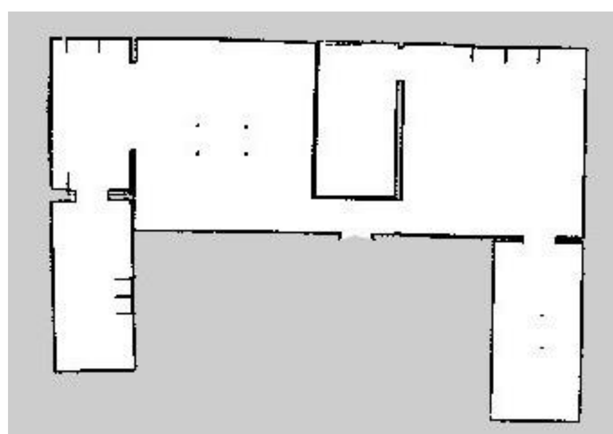

Figure 4: 2-D Map of the Environment

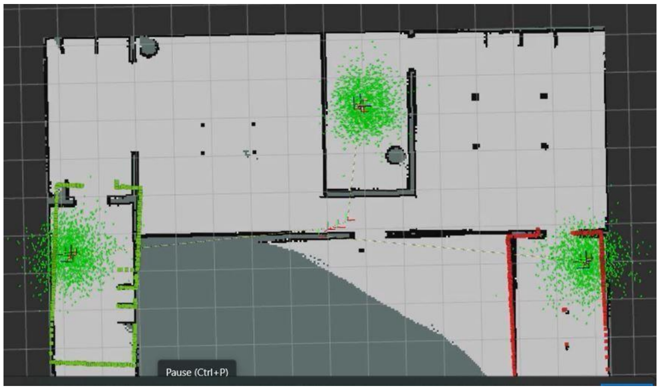

Figure 5: Localization

Figure 6: Navigation

Further, we divided the area among three logical segments so that three robots can be assigned to cover their respective areas. That is, initially generated 2D map is divided into three segments to enable efficient area coverage. 2D map dimension is used to divide the area. Algorithm ensures the optimal way to partition the area. The division of the map into segments is crucial for multi-robot systems as it allows the robots to collaborate and efficiently cover the entire area. Each robot is assigned a specific segment to explore and map, ensuring comprehensive coverage of the environment. Figure 7 illustrates three segments of the map which each of the robots would cover individually.

After the area has been divided, we devised the backtracking spiral algorithm to generate the coverage path for each segment of the environment. The purpose of this algorithm is to generate a path demracated in blue in Figure 8.

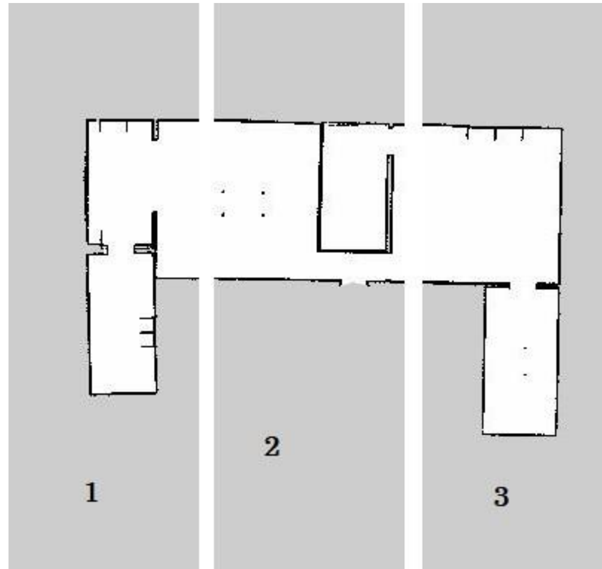

Figure 7: Segmented Maps

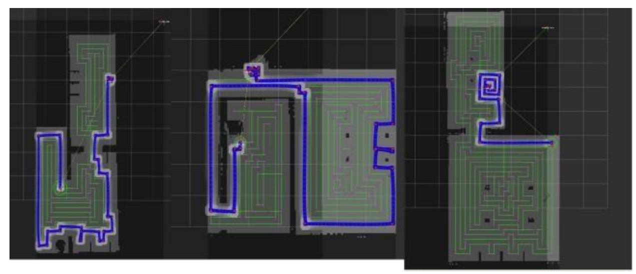

Figure 8: Path Planning

This ensures comprehensive coverage of the entire area with minimal overlap.

V. Conclusion

In this work, algorithm is developed for simultaneous localization and mapping. Using 3-robots in coordination, a reasonable complex environment is accurately mapped which is shown through simulation. Three robots in the environment were also localized and each one could individually navigate to the destination location provided. Furthermore, the 2D map of the world was segmented into three parts and fed as an input to the backtracking spiral algorithm to generate coverage paths to efficiently traverse the given area. Although, we achieved mapping and coverage, but coordinating the movements of multiple robots in a manner that ensures efficient and safe navigation within the environment poses a significant obstacle. Addressing this challenge entails the development of algorithms capable of managing intricate interactions among the robots, including collision avoidance, coordination of actions, and task allocation. Another challenge in multi-robot navigation is dealing with communication issues between robots. Communication is essential for robots to coordinate their actions and share information about the environment. However, communication can be unreliable in some environments and robots may have limited bandwidth for transmitting data. This can make it difficult to maintain an accurate and up-to-date map of the environment which can lead to collisions or other navigation errors. Although, we have simulated the work but the same may be validated and demonstrated in real environment, it would be interesting.

REFERENCES

[1] S. Jain, U. Agrawal, A. Kumar, A. Agrawal and G. S. Yadav, "Simultaneous Localization and Mapping for Autonomous Robot Navigation," 2021 International Conference on Communication, Control and Information Sciences (ICCISc), Idukki, India, 2021, pp. 1-5, doi: 10.1109/ICCISc52257.2021.9484883.

[2] S. Duan, Z. Jiang, Y. Ren and H. Wei, "Coordination of Multi-Robot Systems," 2022 International Conference on Electronics and Devices, Computational Science (ICEDCS), Marseille, France, 2022, pp. 361-367, doi: 10.1109/ICEDCS57360.2022.00086

[3] Lintle Tsiu and Elisha Didam Markus, "A Survey of Formation Control for Multiple Mobile Robotic Systems," International Journal of Mechanical Engineering and Robotics Research, Vol. 9, No. 11, pp. 1515-1520, November 2020. DOI: 10.18178/ijmerr.9.11.1515-1520

[4] H. T. Tran et al., "Extended Kalman Filter (EKF) Based Localization Algorithms for Mobile Robots Utilizing Vision and Odometry," 2022 IEEE 21st Mediterranean Electrotechnical Conference (MELECON), Palermo, Italy, 2022, pp. 91-96, doi: 10.1109/MELECON53508.2022.9843066.

[5] C. S. Tan, R. Mohd-Mokhtar and M. R. Arshad, "A Comprehensive Review of Coverage Path Planning in Robotics Using Classical and Heuristic Algorithms," in IEEE Access, vol. 9, pp. 119310-119342, 2021, doi: 10.1109/ACCESS.2021.3108177.

[6] R. K. Megalingam, A. R, D. HemaTejaAnirudhBabu, G. Sriram and V. S. YashwanthAvvari, "Implementation of a Person Following Robot in ROS-gazebo platform," 2022 International Conference for Advancement in Technology (ICONAT), Goa, India, 2022, pp. 1-5, doi: 10.1109/ICONAT53423.2022.9726010.

[7] R. K. Megalingam, A. Rajendraprasad and S. K. Manoharan, "Comparison of Planned Path and Travelled Path Using ROS Navigation Stack," 2020 International Conference for Emerging Technology (INCET), Belgaum, India, 2020, pp. 1-6, doi: 10.1109/INCET49848.2020.9154132.

[8] Choset, H. Coverage for robotics – A survey of recent results. Annals of Mathematics and Artificial Intelligence 31, 113–126 (2001). https://doi.org/10.1023/A:1016639210559

[9] A. V. S. Bharath Chandra, P. C. Manojkumar, M. B. Tharun, G. Rahul and P. Suresh Kumar, "Object detection, 3-D coordinate extraction and pose estimation for an autonomous medical robot," 2021 5th International Conference on Computer, Communication and Signal Processing (ICCCSP), Chennai, India, 2021, pp. 229-234, doi: 10.1109/ICCCSP52374.2021.9465501.

[10] M. -A. Chung and C. -W. Lin, "An Improved Localization of Mobile Robotic System Based on AMCL Algorithm," in IEEE Sensors Journal, vol. 22, no. 1, pp. 900-908, 1 Jan.1, 2022, doi: 10.1109/JSEN.2021.3126605.

[11] P. Agrawal, A. Iqbal, B. Russell, M. K. Hazrati, V. Kashyap and F. Akhbari, "PCE-SLAM: A real-time simultaneous localization and mapping using LiDAR data," 2017 IEEE Intelligent Vehicles Symposium (IV), Los Angeles, CA, USA, 2017, pp. 1752-1757, doi: 10.1109/IVS.2017.7995960.

[12] H. Yoshisada, Y. Yamada, A. Hiromori, H. Yamaguchi and T. Higashino, "Indoor Map Generation from Multiple LIDAR Point Clouds," 2018 IEEE International Conference on Smart Computing (SMARTCOMP), Taormina, Italy, 2018, pp. 73-80, doi: 10.1109/SMARTCOMP.2018.00076.

[13] Lluvia I, Lazkano E, Ansuategi A. Active Mapping and Robot Exploration: A Survey. Sensors (Basel). 2021 Apr 2;21(7):2445. doi: 10.3390/s21072445. PMID: 33918107; PMCID: PMC8037480.

[14] Jaehyun Kim and Chanwoo Moon, "Multi-Robot System Maintained with Distributed Renewable Energy Sources," International Journal of Mechanical Engineering and Robotics Research, Vol. 11, No. 6, pp. 389-398, June 2022. DOI: 10.18178/ijmerr.11.6.389-398.