Journal of Applied Mechanics, Mechatronics and Energy System,
Vol. 1, Issue 1
May 2026
ISSN: XXXX-YYYY

Design of a Mobile Service Robot Mapping System Based on Hector SLAM

Novian Fajar Satria[1]*, Indra Adji Sulistijono[2], Arneta Firdaus[3], Radina Anggun Nurisma[4]
[1,2,3]*Politeknik Elektronika Negeri Surabaya, Jalan Raya ITS, Kampus PENS, Surabaya, 60111, Indonesia*
**Corresponding author: Novian Fajar Satria; Politeknik Elektronika Negeri Surabaya, Surabaya, Indonesia; email: ovinmeka@pens.ac.id*

Manuscript received July 10, 2026, revised September 10, 2026, accepted October 22, 2026

Abstract – *This research focuses on indoor environment mapping using mobile robots. The unknown environment presents a challenge in robotics. Therefore, the SLAM (Simultaneous Localization and Mapping) method, which has mapping capabilities, is employed to continuously localize the robot's position relative to the map. In this experiment, 2D laser scanning data was obtained using an RPLidar-A1 and processed by the Hector SLAM algorithm, resulting in an occupancy grid map. The experiment involved a mobile robot equipped with a LiDAR sensor, which collected data around the robot as it navigated inside a room, enabling real-time mapping. Upon completion of the experiment, this method successfully mapped the environment around the robot with an average mapping error of 17.67 cm.*

Keywords: *Hector SLAM, Lidar sensor, Mobile robots.*

I. Introduction

Primary challenges faced by mobile robots in accomplishing their tasks include the ability to navigate autonomously from a starting point to a designated end point without damaging the surrounding environment and avoiding obstacles. Similar to a human trapped in a maze, a robot requires a map of the maze to find its way out. Additionally, a human needs to know its position within the map to determine the appropriate path. Likewise, the biggest challenge for mobile robots is how to acquire a map of the environment it traverses during operation and also determine its own position within the map.

In the evolving landscape of robotics, autonomous navigation remains a cornerstone of research and application, particularly for mobile robots. The ability of a robot to navigate an unknown environment autonomously relies heavily on effective Simultaneous Localization and Mapping (SLAM) techniques. Among the myriads of SLAM algorithms developed, Hector SLAM has emerged as a prominent method, known for its efficiency and accuracy in producing real-time maps. This paper delves into the performance analysis of Hector SLAM, focusing on its application in mobile robot navigation [1].

Hector SLAM, or "High-Efficiency Comprehensive Tracking and Real-time Mapping," leverages the high update rate of laser scanners to facilitate rapid and precise mapping. Unlike many traditional SLAM approaches that depend on wheel odometry, Hector SLAM utilizes the raw laser scan data directly, making it highly suitable for environments where wheel slippage or uneven terrains could undermine the accuracy of odometry-based systems [2]. This feature allows Hector SLAM to perform robustly in diverse and challenging indoor scenarios, making it a valuable asset for mobile robot navigation [1].

The performance of Hector SLAM is influenced by various factors, including the resolution of the laser scanner, the computational power of the onboard processor, and the complexity of the environment. Evaluating these factors requires a systematic analysis of the algorithm's accuracy, efficiency, and reliability under different conditions [3]. This study aims to provide a comprehensive evaluation by conducting a series of experiments that measure Hector SLAM's performance in terms of mapping accuracy, computational load, and its ability to handle dynamic changes in the environment [1]. Understanding the strengths and limitations of Hector SLAM is crucial for its effective deployment in real-world applications [4]. The insights gained from this performance analysis will inform the development of more resilient and adaptable robotic systems, capable of operating autonomously in a wide range of settings.

II. Simultaneous Localization and Mapping

Simultaneous Localization and Mapping (SLAM) is a fundamental technique in robotics that enables a mobile robot to create a map of an unknown environment while

simultaneously determining its location within that map. The development and refinement of SLAM algorithms have been pivotal in advancing the capabilities of autonomous robotic systems. SLAM can be broadly classified into two main categories: EKF-SLAM (Extended Kalman Filter SLAM) and Graph-based SLAM [1]. EKF-SLAM employs a probabilistic framework to estimate the robot's pose and the positions of landmarks. The state vector X_t at time t includes the robots pose X r/t and the positions of n landmarks m_i.[2].

The state transition model f and observation model h are defined as:

$$X_t + 1 = f(X_t + U_t) + W_t$$
$$X_t + 1 = f(X_t + U_t) + W_t \quad (1)$$
$$Z_t = h(X_t, m_i) + V_t Z_t = h(X_t, m_i) + V_t \quad (2)$$

Where $U_t\, U_t$ is the control input, $Z_t Z_t$ is the observation, $W_t W_t$ and

are process and observation noise respectively, typically assumed to be Gaussian [2]. The EKF updates the state estimate and covariance matrix P using the following equations:

$$K_t = P_{t|t-1} H_t^T \left(H_t P_{t|t} + 1 H_t^T + R\right)^{-1}$$
$$K_t = P_{t|t-1} H_t^T \left(H_t P_{t|t} + 1 H_t^T + R\right)^{-1} \quad (3)$$
$$X_{t|t} = X_{t|t-1} + K_t \left(Z_t - h(X_{t|t} - 1)\right)$$
$$X_{t|t} = X_{t|t-1} + K_t \left(Z_t - h(X_{t|t} - 1)\right) \quad (4)$$
$$P_{t|t} = (I - K_t H_t)\, P_{t|t-1} P_{t|t} = (I - K_t H_t)\, P_{t|t-1} \quad (5)$$

Graph-based SLAM, another widely used approach, formulates the SLAM problem as a graph optimization problem [3]. Nodes in the graph represent robot poses and landmarks, while edges represent constraints between them. The objective is to find the configuration of nodes that minimizes the overall error in the constraints, typically expressed as:

$$X^* = \arg min_x \sum i, j X^* = \arg min_x \sum i, j \,||\, z_{ij}\, z_{ij} - \text{h}\,(X_i X_i, X_i X_i)||\frac{22}{\Omega\Omega} \quad (6)$$

Where $z_{ij}\, z_{ij}$ is the measurement between nodes and i and j $\Omega_{ij}\, \Omega_{ij}$ is the information matrix (inverse covariance), and h $(X_i X_i, X_i) X_i)$ is the predicted measurement. Optimization techniques such as Gauss-Newton, Levenberg-Marquardt, or Powell's Dog-Leg method are commonly used to solve this problem [3]. Hector SLAM is a highly efficient and accurate SLAM algorithm that distinguishes itself by utilizing high update rates of laser scanners and not relying on traditional odometry information. This method is particularly advantageous in environments where wheel slippage or uneven terrains undermine the accuracy of odometry-based systems. Hector SLAM directly processes raw laser scan data to estimate the robot's pose and build a map in real-time. The algorithm employs a multi-resolution grid approach to efficiently manage computational resources while maintaining high accuracy [4]. The fundamental concept behind Hector SLAM is the use of a Gauss-Newton optimization technique to align the incoming laser scans with the existing map, updating the robot's pose iteratively [5]. The process involves minimizing the error between the predicted and observed laser scans, given by:

$$E(X_t E(X_t) = \sum i \sum i \,||\, Z_i Z_i _ h(xt)||^2 h(xt)||^2 \quad (7)$$

Where $Z_i Z_i$ represents the individual laser scan measurements, and $h(xt)\, h(xt)$ is the predicted measurement from the map. By continuously adjusting the robot's pose to minimize this error, Hector SLAM can achieve highly accurate localization and mapping without the need for external odometry data [5].

Understanding the strengths and limitations of Hector SLAM is crucial for its effective deployment in real-world applications. Recent advancements in computational power and sensor technologies have enabled more sophisticated SLAM algorithms that combine elements of both EKF and Graph-based methods to leverage their respective strengths [6]. These hybrid approaches aim to improve the robustness, accuracy, and efficiency of SLAM systems, pushing the boundaries of autonomous navigation capabilities in mobile robots [4].

III. Methodology

A. Hardware Design

The hardware that comprises the robot prototype is divided into three main blocks: input, process, and output. In the input block, LiDAR and the rotary encoder serve as sensors that enable the robot to perform simultaneous localization and mapping (SLAM). Although both devices aim to facilitate this capability, they serve different functions. The rotary encoder is responsible for measuring the distance traveled by the robot and relaying this information to the Arduino. In contrast, LiDAR maps the robot's surrounding environment and sends this data to the Raspberry Pi.

In the process block, communication between the Arduino and the Raspberry Pi is crucial. This process involves transmitting speed information generated by the velocity topic command in the ROS network to the Arduino. Conversely, the Arduino provides feedback by sending speed data from the encoder back to the ROS network. The serial communication link between the Arduino and Raspberry Pi allows the Arduino to publish and subscribe to information within the ROS network directly. The hardware design of this system is illustrated in Figure 1.

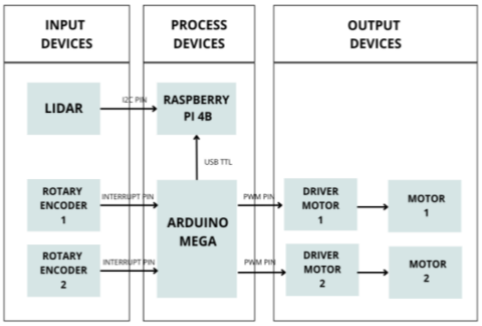


Fig. 1 Hardware design of the system

B. Mechanic Design

In this prototype, the only output hardware component is the DC motor, which plays a crucial role in propelling the robot from one location to another. This DC motor is managed by the L298N motor driver, a robust module that can deliver up to 43 amperes of current, providing sufficient power for the robot's mobility needs. The L298N motor driver uses a straightforward H-bridge circuit, which enables the motor to move forward or backward, and this is controlled via Arduino's digital pins. By precisely managing the direction and speed of the motor, the system ensures accurate and responsive movement of the robot.

The robot is designed with two wheels centrally positioned on its sides, which form the primary means of locomotion. This configuration is chosen to balance the robot and provide optimal maneuverability. Additionally, critical electronic components, including the Arduino and Raspberry Pi, are securely housed within the central frame of the robot. These components are enclosed in a protective box, safeguarding them from external damage and ensuring stable operation during the robot's movement and various tasks. This strategic placement within the robot's frame helps maintain a low center of gravity, enhancing stability and performance angles. The mechanic design of this system is illustrated in Figure 2.



Fig. 2 Mechanic design of the robot

C. Hector SLAM

Hector SLAM (Simultaneous Localization and Mapping) is a sophisticated and efficient algorithm designed for real-time mapping and localization of mobile robots. Unlike traditional SLAM algorithms that rely heavily on odometry data, Hector SLAM primarily utilizes LIDAR (Light Detection and Ranging) sensor data to construct and update local maps dynamically. This reliance on LIDAR data allows Hector SLAM to function effectively in environments where odometry data might be unreliable or unavailable, making it a robust solution for various real-world applications.

One of the primary strengths of Hector SLAM is its ability to perform in real-time. The algorithm continuously processes LIDAR scans, using the time it takes for laser beams to return after hitting objects to create detailed maps of the environment. These maps are updated dynamically as the robot moves, with new LIDAR scans being matched to the existing map to refine the robot's estimated position and incorporate new environmental information. This real-time performance is critical for applications in dynamic settings where rapid updates to the map are necessary to maintain accurate localization.

To achieve optimal performance, Hector SLAM can be configured with customized parameters tailored to specific environments. These parameters include map resolution, scanning interval, and detection threshold. The map resolution determines the level of detail in the generated map, with higher resolutions providing more detail at the cost of increased computational resources. The scanning interval sets how frequently the LIDAR sensor performs scans, with shorter intervals leading to more frequent updates but potentially higher computational loads. The detection threshold adjusts the sensitivity of the LIDAR sensor, balancing between detecting fine details and filtering out noise. Proper configuration of these parameters ensures that Hector SLAM can adapt to and perform optimally in diverse environments.

Compared to traditional SLAM methods that rely on odometry data from wheel encoders, Hector SLAM offers significant advantages. Traditional methods can accumulate errors over time due to wheel slippage or uneven terrain, leading to inaccuracies in mapping and localization. Hector SLAM, by relying solely on LIDAR data, avoids these issues, making it more robust and accurate in various operating conditions. This independence from odometry enhances Hector SLAM's reliability and accuracy, particularly in challenging environments where other methods might struggle. The configuration and implementation of Hector SLAM in this study, with tailored parameters for map resolution, scanning interval, and detection threshold, have demonstrated its efficacy and robustness, confirming its

suitability for a wide range of robotic mapping applications.

Initially, the algorithm initiates the scanning process to estimate the robot's initial position. Following a short movement, the laser performs a scan, and subsequently, the second scan yields a new set of results and estimations. This data is then compared with the existing map to calculate the positional variance, with this iterative process continuing several times, progressively updating the map and refining the robot's approximate position. Overview of hector SLAM can be seen in Figure 3.

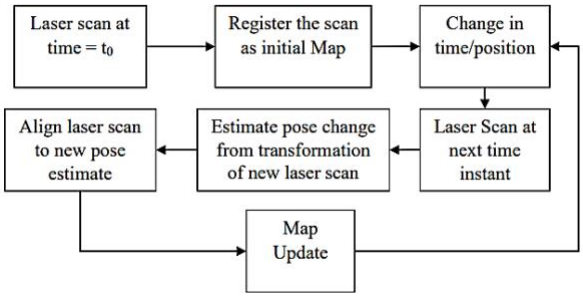


Fig. 3 Overview of Hector SLAM

D. Obstacle Detection Testing

Obstacle detection testing on the Hector SLAM system was carried out in the hallway on the 3rd floor of the Surabaya State Electronics Polytechnic Postgraduate. In this test, there were three static obstacles in the form of humans placed at strategic points in the hallway. The Hector SLAM system is able to recognize human movements well, accurately mapping their position in the environment. As humans move around the hallway, Hector SLAM quickly and efficiently responds to their movements, updating the mapping.

Apart from that, the system can also detect stationary objects such as walls well. When the system performs hallway mapping, Hector SLAM can precisely map the contours of the walls, identifying their locations accurately. Static testing obstacles can be seen in Figure 4.



Fig. 4 Static Obstacle

Test results show that the system has high accuracy in distinguishing between stationary and moving objects, as well as fast responsiveness to environmental changes. This testing also confirmed the system's consistency in detecting and mapping obstacles over time, despite variations in human movement and lighting in the aisles. Overall, the test successfully demonstrated that the Hector SLAM system can recognize human movements and detect stationary objects such as walls with high accuracy and responsiveness. This proves the system's ability to operate safely and effectively in a dynamic environment such as the 3rd floor hallway of the Surabaya State Electronics Polytechnic Postgraduate School. The sensor test results can be seen in Figure 5.

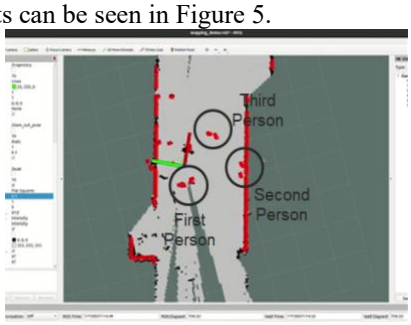


Fig. 5 Obstacle Detection Result

IV. Experimental Result

The placement of the Lidar sensor on the robot is crucial to ensure accuracy and effectiveness in environmental mapping. In this research, the RPLidar-A1 sensor is mounted on the lower part of the robot since the robot's height is approximately 160 cm. This lower placement is chosen because the operational environment of the robot contains obstacles mostly below 160 cm in height, such as tables, chairs, and humans. This strategic positioning of the sensor allows for 360-degree scanning coverage around the robot, enabling it to collect real-time data as it moves through the building's corridors. The placement of the sensor, as shown in Figure 6.

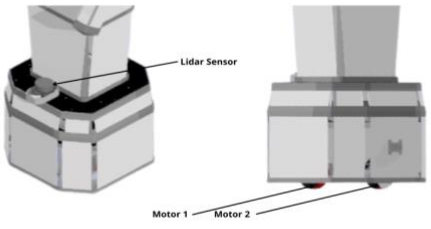


Fig. 6 Placement of the lidar sensor on the robot

Figure 5 shows the raw scan results produced by the rplidarNode program. The program generates a "scan" topic and uses LaserScan messages. This data is then used as input for the hector_mapping program, which generates several topics such as map_metadata, map, slam_out_pose, and poseupdate. The hector_mapping program uses the tf (transform) application for scanning

data transformation, allowing the lidar device to change position independently of the defined base frame. The RViz application can display the mapping results immediately and in real time.

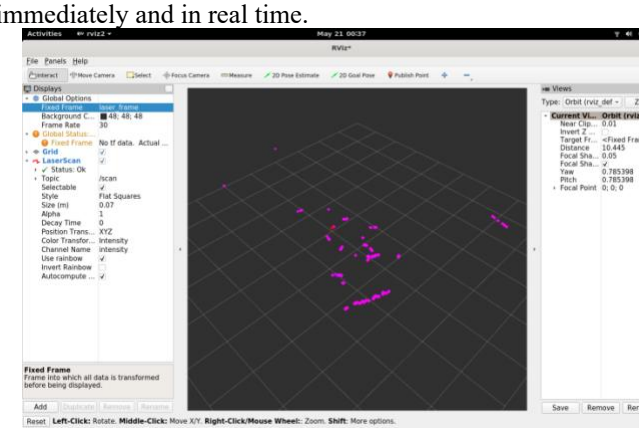


Fig. 7 Scan topic result on Rviz

Simultaneous localization and mapping experiments were conducted using the Hector SLAM algorithm in three distinct laboratory rooms situated in the Graduate Building of Politeknik Elektronika Negeri Surabaya on the third floor. All these rooms feature a flat floor surface. This experiment aims to evaluate how the map results generated by the Hector SLAM algorithm correspond to the original environment.

TABLE I
COMPARATION DATA RESULT

Room 1			
Sector	**System Measurement (cm)**	**Actual Data (cm)**	**Erorr (cm)**
A	760	745	15
B	660	638	22
C	470	460	10
Room 2			
Sector	**System Measurement (cm)**	**Actual Data (cm)**	**Erorr (cm)**
A	764	778	14
B	808	746	62
C	180	175	5
Room 3			
Sector	**System Measurement (cm)**	**Actual Data (cm)**	**Erorr (cm)**
A	806	795	11
B	553	535	18
C	491	493	2

From the provided data, the results of mapping using the Hector SLAM algorithm for three different rooms can be observed. Each room is divided into several sectors, and the mapping system's results are compared with the actual measured data. Data analysis reveals variations in the level of error between the mapping system's results and the actual data in each room. Average errors are calculated for each room, which can then provide a general overview of the mapping accuracy.

In Room 1, the average mapping error is 15.7 cm, with the largest error occurring in Sector B at 22 cm. The smallest error in Room 1 is in Sector C, with an error of 10 cm. Room 2 exhibits an average error of 27 cm, with the largest error reaching 62 cm in Sector B. The smallest error in Room 2 is in Sector C, with an error of 5 cm. Meanwhile, Room 3 shows an average error of 10.3 cm, with the largest error occurring in Sector B at 18 cm. The smallest error in Room 3 is in Sector C, with an error of 2 cm. This variation indicates different challenges in mapping accuracy in each room, possibly influenced by factors such as the complexity of physical structures or sensor quality. Further analysis highlights the importance of understanding the factors affecting mapping accuracy, including sensor calibration and algorithm parameter settings. Further research can be conducted to evaluate the effectiveness of corrective measures taken to improve mapping accuracy. The mapping result of Hector SLAM can be seen in Figure 8.

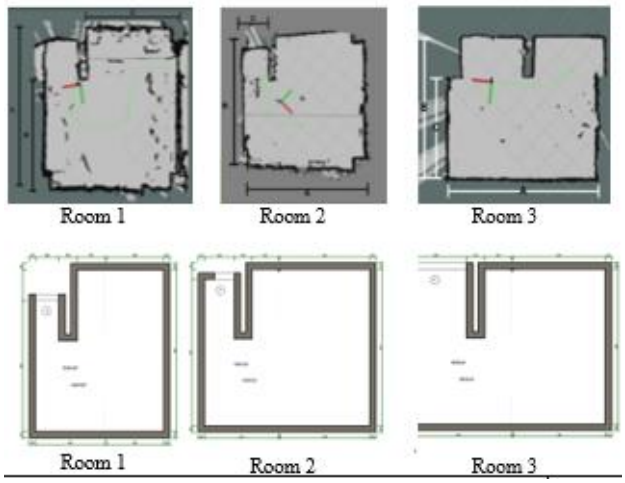


Fig. 8 Mapping result alongside the original floor plan of the room.

From the mapping results, it was determined that the average error for Room 1 was 2,55%. The highest error occurred in Sector B at 3.45%, while the smallest error occurred in Sector A at 2.01%. For Room 2, the average error was 4,32%, with the highest error reaching 8.33% in Sector B and the smallest error at 1.80% in Sector A. Meanwhile, for Room 3, the average error was 1,72%, with the highest error at 3.36% in Sector B and the smallest error at 0.41% in Sector C.

Analysis indicates that these errors are largely due to the presence of backlash in the dc motor. Dc motor backlash has a more significant impact on errors when the leg height is low, as the leg frames support the robot's weight almost horizontally. Conversely, when the robot's leg height is sufficient, the leg frames are in a vertical position, reducing the likelihood of backlash when supporting the robot's weight. Therefore, to improve mapping accuracy, adjustments and compensation need to be made to the servo system, both mechanically and through the use of algorithms that account for backlash. SLAM Method testing was carried out to test the results from the combined Hector SLAM algorithm with the extended kalman filter method. Testing is carried out on the Basement of the Politeknik Elektronika Negeri Surabaya Postgraduate Building. Environmental

conditions and floor plan The test site can be seen in Figure 9.

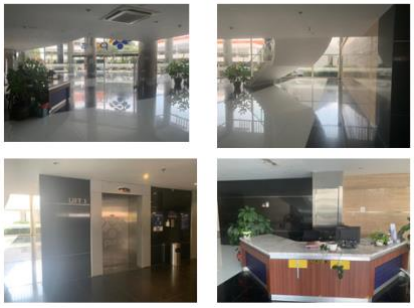


Fig. 9 Basement of the PENS Postgraduate Building

In this visualization, two different pieces of information produced by the Hector SLAM system are clearly visible. First, the green path reflects the journey of the LiDAR sensor, accurately depicting the route the robot has taken from Point A to Point D. This path provides a visual view of how the system moves and navigates throughout the test journey. In addition, the red nodes visible around the green path indicate obstacle detection by the LiDAR sensor. This red track visualizes how the Hector SLAM system successfully identifies and avoids obstacles that may arise during the journey. This information is important for evaluating the system's ability to respond and adapt to changing environmental conditions, such as the presence of stairs, tables, or other objects detected as obstacles. Apart from these two pieces of information, a 2D map can also be seen on Rviz which provides a more detailed visual representation of the environmental structure in the basement of the PENS Postgraduate Building.

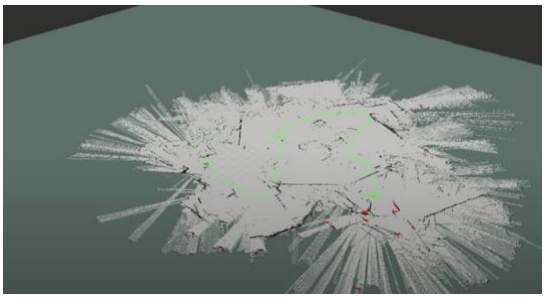


Fig. 10 Mapping Result

V. Conclusion

Based on the experimental results, it can be concluded that the Hector SLAM algorithm demonstrates varying levels of mapping accuracy across different rooms and sectors. Room 2 generally shows the highest mapping errors, particularly in Sector B, where complex structures or environmental factors may challenge the algorithm's accuracy. Conversely, Room 3 exhibits relatively lower errors, indicating better performance in that environment. Sector B consistently experiences the highest errors across all rooms, signaling a need for further refinement in mapping complex areas.

In total, the mapping errors across all rooms amount to 26.77%, highlighting the algorithm's overall inaccuracies in various settings. Around the robot, an average mapping error of 17.67 cm was observed. However, using rviz and ROS tools, the Hector SLAM method provides stable results, achieving an average error rate of 2.5% in room dimension representation and a maximum deviation of 1.8 cm from the expected path. This indicates that despite the noted inaccuracies, the system offers a reliable solution for area mapping, particularly in the PENS Postgraduate Building, where accuracy and stability have been validated through experimental data.

ACKNOWLEDGEMENTS

This work was supported by the Department of Mechanical and Energy Engineering, Electronics Engineering Polytechnic Institute of Surabaya (PENS).

REFERENCES

[1] Kohlbrecher, S., Meyer, J., von Stryk, O., & Klingauf, U. (2011). *A flexible and scalable SLAM system with full 3D motion estimation.* In IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR).

[2] Thrun, S., Burgard, W., & Fox, D. (2005). *Probabilistic Robotics.* MIT Press.

[3] Grisetti, G., Kummerle, R., Stachniss, C., & Burgard, W. (2010). *A tutorial on graph-based SLAM.* IEEE Intelligent Transportation Systems Magazine, 2(4), 31-43.

[4] Kohlbrecher, S., von Stryk, O., Meyer, J., & Klingauf, U. (2011). *Hector SLAM for robust mapping in autonomous robot competitions.* Journal of Field Robotics, 31(4), 673-685.

[5] Kummerle, R., Grisetti, G., Strasdat, H., Konolige, K., & Burgard, W. (2011). g2o: *A general framework for graph optimization.* In IEEE International Conference on Robotics and Automation (ICRA).

[6] Durrant-Whyte, H., & Bailey, T. (2006). *Simultaneous localization and mapping: part I.* IEEE Robotics & Automation Magazine, 13(2), 99-110.

[7] S. F. A. E. Wijaya, D. S. Purnomo, E. B. Utomo, and M. A. Anandito, "*Research Study of Occupancy Grid Map Mapping Method on Hector SLAM Technique*," in 2019 International Electronics Symposium (IES), Surabaya, Indonesia, 2019.

[8] Zhang Xuexi, Lai Jiajun, Xu Dongliang. "*2D Lidar Based SLAM and Path Planning for Indoor Rescue Using Mobile Robot*." (2020). Journal of Advanced Transportation. Vol. 2020, no. 1, pp. 1-10.

[9] Junli Gao, Weijie Ye, Jing Guo and Zhongjuan Li, (2020). "*Deep Reinforcement Learning For Indoor Mobile Robot Path Planning*"Journal Sensors.

[10] Yang Xiaozhou (2021). "*SLAM dan Navigation of Indoor Robot based on ROS and Lidar*". Master Thesis, Wuhan University of Technology.

[11] P. Vanicek, L. Beran. (2018). "*Navigation of Robotics Platform in Unknown Spaces*." In: Proceedings of the 2018 International Conference on Robotics and Automation (ICRA), Madrid, Spain, pp. 1-6. IEEE.

[12] Laksono, P. S., & Kusuma, T. M. (2022). *Performance Analysis of Hector SLAM and Gmapping for Navigation for Mobile Robot Navigation*. Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Gunadarma. Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat.

[13] Tang, B. W., Zhu, Z. X., & Luo, J. J. (2018). *Hybridizing Particle Swarm Optimization and Differential Evolution for the Path Planning of Mobile Robots*. Symmetry, 10(10), 433.