Mobile Robotic-Arm Development for a Small-Scale Inter-Room Logistic Delivery using 2D LIDAR-guided Navigation

Hadha Afrisal [1,3*], Ghanis Kauchya Nugraha [1], Aan Aria Nanda [1], Ahmad Didik Setiyadi [1], Olimjon Toirov [4], Rifky Ismail [2,3], Wahyul Amien Syafei [1], Munawar Agus Riyadi [1,3], Iwan Setiawan [1]

[1] *Department of Electrical Engineering, Faculty of Engineering, Universitas Diponegoro*
[2] *Department of Mechanical Engineering, Faculty of Engineering, Universitas Diponegoro*
[3] *Center for Biomechanics, Biomaterials, Biomechatronics, and Biosignal Processing (CBIOM3S)*
Jl. Prof. Soedarto, SH, Kampus UNDIP Tembalang, Semarang, Indonesia 50275
[4] *Department of Electrical Machine, Tashkent State Technical University*
University Street No.2, Tashkent City, Uzbekistan 100095

Abstract

This research aims to develop a low-cost mobile robotic arm for an indoor delivery system. The current pandemic and the possible future threat of communicable disease have become challenging scenarios in developing an unmanned logistic delivery system with minimum human involvement, especially for inter-room items delivery inside highly regulated buildings such as hospitals, clinics, pharmaceuticals, foods, and beverages industries. In this paper, a prototype of the mobile robotic arm is designed to achieve an autonomous level of navigation utilizing a 2D LIDAR with guided remote monitoring and control of object selections for the loading/unloading process. The mobile robotic arm is an integrated robotic system of a mobile robot with an attached robotic arm in its body. The base of the mobile robotic arm utilizes a differential-drive configuration equipped with a wheels odometry system. The robotic arm is configured with a 4-DoF SCARA-like structure. The 2-dimensional environment map is generated using the LIDAR sensor utilizing the Hector SLAM method prior to navigation. The autonomous navigation is performed using a 2D LIDAR-based technique by employing an A* algorithm for path planning and tracking mechanism. Experimental works were conducted in a small building environment consisting of some rooms and narrow corridors. The result of experiments shows that the prototype of the mobile robotic arm can safely and effectively navigate through the testing environment and, subsequently, load and unload objects from one room to another room without colliding with objects and obstacles.

Keywords*: mobile robotic arm; inter-room delivery; mapping and navigation; pick and place*

1 . Introduction

In conjunction with the negative impacts of the recent pandemic, the demand for unmanned robotic systems has been significantly increasing, especially in the healthcare, medicine, and food industries (Shen et al., 2020). The main objective of the development is to provide services with minimum human involvement and biological contact in all aspects of the processes (Magid et al., 2021). In recent years, the focus of technology development in the medical field has also transformed from focusing on disease treatment to a prevention-oriented effort (Kwon et al., 2022). Some of the leading

technologies have been implemented to improve preventive actions against the spreading of coronavirus and other communicable diseases, one of which, the unmanned system such as in the form of mobile robots, are widely developed and deployed in many scenarios to mitigate the risks of virus transmission and to maintain a low-level of biological contamination (Tavakoli, Carriere, & Torabi, 2020).

Robotics, automation, and cyber-physical systems can be adapted to provide assistive technology for workers in risky tasks within the contagious environment or maintain a low level of human involvement in the processes (Haidegger et al., 2020; Evans, Medina, & Dwyer, 2018). There are some scenarios of robot utilization in those environments, such as (1) disinfection and cleaning, (2) logistic delivery and services, (3) tele-

---

*) Corresponding author.
E-mail: hadha.afrisal@elektro.undip.ac.id

operation and tele-presence, and (4) detection and control (Sierra Marín et al., 2021). The concept of the smart hospital (Kwon et al., 2022) and smart industry (Chitrakar, Zhang, & Bhandari, 2021) have also been introduced to develop an integrative, intelligent system that can support these applications

Autonomous robots and tele-robotics applications are the potential to be utilized in delivering items in highly regulated environments to minimize the risks of communicable disease transmission (Niemeyer et al., 2016). Tele-robotics is a sub-field of robotics that aims to develop robots and autonomous systems that can be operated from a remote distance, either guided or unguided, with advanced sensors. Tele-robotics is an inseparable field of robotics that aims to provide services remotely. Thus it does not require direct physical contact and presence (Avgousti et al., 2016). Mobile robotics systems in the medical field can be widely applied in diagnostic procedures, treatments, and medical interventions carried out remotely. Meanwhile, mobile robotics are mainly used in the food industries for sorting and delivery with minimum human complicity and biological contamination (Chitrakar, Zhang, & Bhandari, 2021).

Tavakoli et al. (2020) classify tele-robotics applications into 4 primary categories in order to tackle COVID-19 and potential harms of communicable diseases, such as (1) emergency or critical conditions, (2) primary prevention and healthcare support, (3) household and long-term home care, and (4) medical education and training. In some scenarios, mobile robotic technology can be used as a transport for delivering materials in the form of samples, containers, or foods to maintain a high level of cleanliness and hygiene from biological contaminations (Sierra Marín et al., 2021; Chitrakar, Zhang, & Bhandari, 2021).
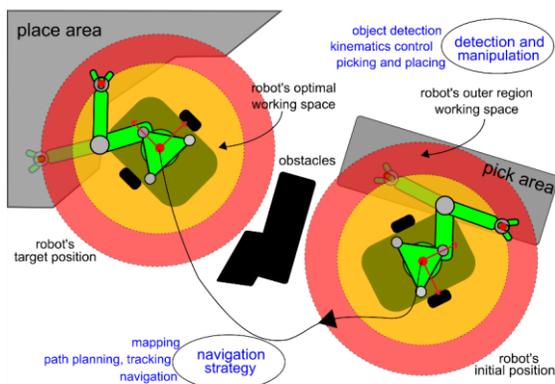
In response to those recent issues and studies, this research focuses on developing a mobile robotic arm to transport materials from one room to another utilizing a 2D 360-degree LIDAR sensor for mapping, path planning, and materials loading/unloading. The mobile robotic arm is the integration of a mobile robot with a robotic arm attached to its body (Pandey, Pandey, & Parhi, 2017). The base of our mobile robotic arm utilizes differential-drive wheels equipped with wheels odometry and the robotic arm is configured with a 4-DoF SCARA-like structure. Autonomous navigation is performed using a 2D LIDAR-based technique, and the path planning and tracking method is developed using an A* algorithm. The environment mapping is carried out using the Hector SLAM technique before navigation. All experiments are conducted in a small building consisting of some rooms and hallways with some obstacles placed in the environment.

2. Materials and Method

A mobile robotic arm system is designed for the implementation scenario of the inter-room delivery process, which includes mapping, navigation (path planning and tracking), and loading/unloading process, as illustrated in Figure 1. The macro-development stages can be partitioned into 2 parts. The first is the design of the whole system for mechanical and electronic systems, which is consecutively presented in Figure 2 and Figure 4. The second is the robot's control strategy which includes low-level (pose, movement, and position control) and high-level control strategy (mapping, path planning, tracking, load/unload mechanism) as presented
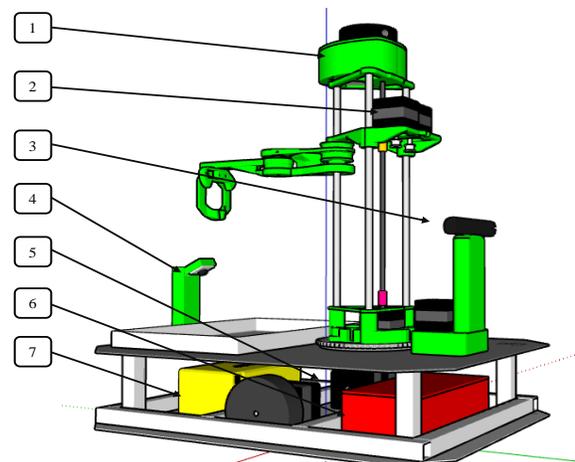

Figure 1. Implementation scenario of mobile robotic arm for delivery with load/unload mechanism.

Figure 2. Design of mobile robotic arm system consisting of (1) RPLidar 360 laser scanner, (2) robotic arm joints, (3 and 4) detection cameras, (5) motors with built-in rotary encoders, (6) robot's controller, (7) battery.
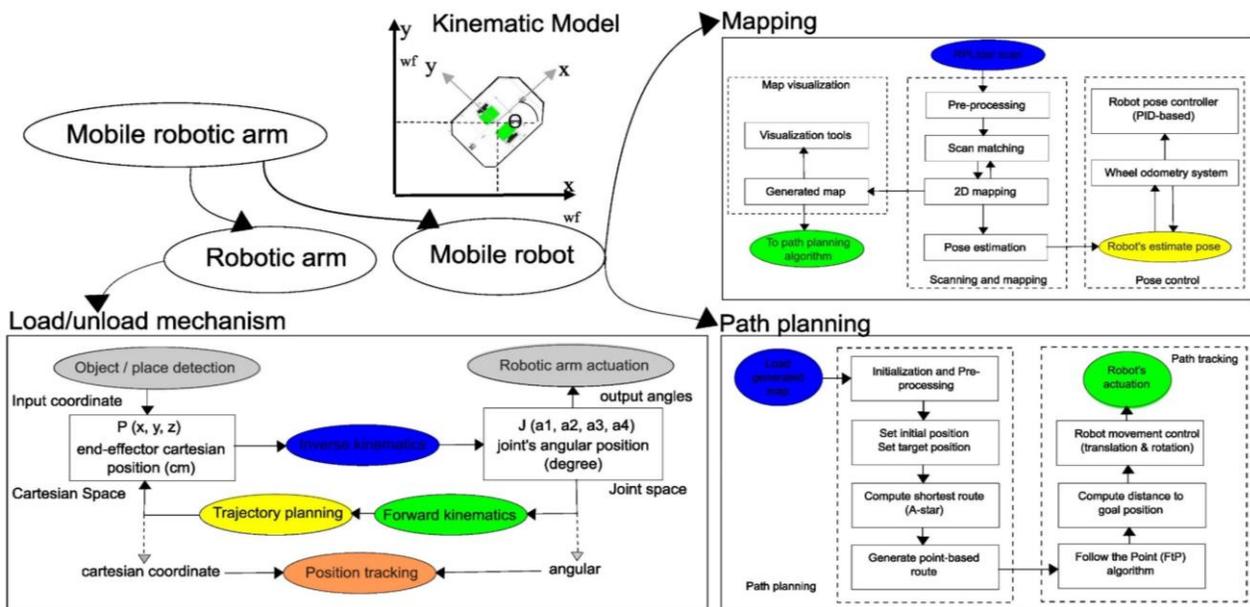
Figure 3. An integrative view of the mobile robotic arm's control strategy includes the development of mapping and path planning techniques for mobile robots and the loads/unload mechanism for the robotic arm.

in Figure 3. The body of the mobile robotic arm is primarily constructed to comply with navigation through narrow corridors and doors. The mechanical system is designed using SolidWorks and presented in Figure 2. The mechanics of the mobile robotic arm is built of aluminum metals and 3-D printed industrial plastics. The mobile robotic arm is equipped with sensors such as RPLidar 360 for building a 2-dimensional map and

Figure 4. Electronics system block diagram of mobile robotic arm consisting of sensors, actuators, and controllers.

guiding navigation, rotary encoders (600 PPR) for wheels odometry, and 2 cameras (Logitech C920) for conducting object selection.

2. 1. Design of Mobile Robotic Arm

The mobile robotic arm is built of 2 main parts: (1) mobile robot base and (2) robotic arm, as presented in the block diagram of electronic system design in Figure 4. The mobile robot is supported by a couple of differential drive wheels of DC motors with the built-in rotary encoders 600 PPR. To control the DC motor, a full H-bridge motor driver of BTS7960 is utilized. The robotic arm is designed as a 4-DoF SCARA-like configuration of 4 pieces NEMA stepper motors with mechanically linked rotary encoders 600 PPR. The stepper motors are controlled using the stepper motor controller TB6600. The dimension of the mobile robotic arm is 50 cm in length, 40 cm in width, and 45 cm in height. The advantage of utilizing the mobile robotic arm configuration is that the robot can navigate agilely from one place to another. The robot can also perform the task of grasping, making it possible to move objects from one room to another without being limited by constraints of the robotic arm's range (Yamamoto et al., 2018; Deng, Xiong, & Xia, 2017).

The main challenge in the development stage is that a mobile robot with an attached robotic arm is a type of robot which is relatively complex to control. Thus a control system is needed to control the robot optimally (Kolhatkar & Wagle, 2021). In accordance with that, this research attempts to reduce the complexity of robot control by proposing a scheme of control strategy, as
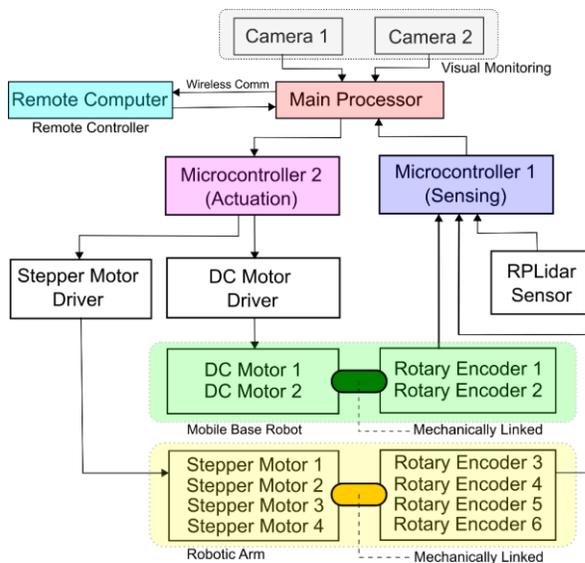
presented in Figure 3, to perform sequential processes of mapping, path planning and tracking, and loading/unloading mechanisms. This research will also discuss optimal navigation in the items' delivery process. To be added, although the control strategy is processed sequentially, the mobile robot can respond to moving obstacles using the information from LIDAR by pausing the movement for safety.

2. 2. Robot Control Strategy

The robot's control strategy can be elaborated into low-level and high-level control strategies. The low-level control strategy deals with the self-maneuver control of the robot, such as controlling the mobile robot's pose, movement, and position relative to the world coordinate frame. Using the kinematics model, this low-level control strategy is implemented by controlling the speed of 2 DC motors as the differential drive wheels.

The kinematics model of the mobile robot is visualized in Figure 3. Mobile robot kinematics $(\dot{x}, \dot{y})$ can be obtained using linear velocity v with simple trigonometric regards to the angle θ of mobile robot in 2-dimensional (top view) Cartesian space (X-axis and Y-axis) using Equations (1) and (2). Angular velocity is the first derivative of angular position, thus by considering the radius of the right and left wheels of the mobile robot, the formulation of linear velocity v and angular velocity ω of the mobile robot can be calculated using Equations (3) and (4), in which ω_r and ω_l are the angular velocity of right and left wheels, and D and R are the diameter and radius of the wheels. Then, the angular velocity of the right and left wheels of the robot can be computed using Equations (5) and (6).

$$\dot{x} = v \cdot \cos\theta \qquad (1)$$

$$\dot{y} = v \cdot \sin\theta \qquad (2)$$

$$v = \frac{\omega_r \cdot R_r + \omega_l \cdot R_l}{2} \qquad (3)$$

$$\omega = \frac{\omega_r \cdot R_r - \omega_l \cdot R_l}{D} \qquad (4)$$

$$\omega_r = \frac{\left(v + \frac{\omega \cdot D}{2}\right)}{R_r} \qquad (5)$$

$$\omega_l = \frac{\left(v - \frac{\omega \cdot D}{2}\right)}{R_l} \qquad (6)$$

As visualized in Figure 3, the high-level robot control strategy is designed to satisfy the most basic requirements for autonomous robot navigation, namely (1) mapping, (2) path planning and tracking, and (3) loading/unloading mechanism. In this research, the mapping process is performed utilizing the Hector SLAM technique, which combines the sparse data points gathered from the LIDAR sensor and the wheels' odometry. The path planning technique utilizes an A* algorithm to plan the shortest route of the robot's movement from an initial location to the goal location, while the path tracking algorithm is done using Follow the Point (FtP) algorithm to follow the generated path to the goal location. In the loading/unloading mechanism, 2 cameras are employed to detect objects and monitor the loading/unloading process; subsequently, a robotic arm with a gripper picks up the detected object (loading) and places it in the goal location (unloading).

The mapping process utilizes the Hector SLAM algorithm (Zhang et al., 2021) to generate a 2-dimensional map of the environment using RPLidar 360 prior to robot navigation. Regarding mapping accuracy, Hector SLAM is comparable to the GMapping method (Norzam, Hawari, & Kamarudin, 2019). While the GMapping technique utilizes a Rao-blackwellized particle filter, the Hector SLAM utilizes a scan-matching technique which is relatively simpler and faster in implementation (Olalekan et al., 2021). Hector SLAM relies on the high-frequency data from LIDAR to estimate the robot's position and orientation relative to the robot's body. By using the transformation method, the position and orientation of the robot with regard to the world frame can be estimated (Harik, Korsaeth, & others, 2018). The process of developing the Hector SLAM method is presented in Figure 3, in the section of the mapping block system.

A path planning algorithm is developed to find the best route to the targeted location. We utilize an A* algorithm that combines the reading of the LIDAR sensor in real time and the generated map from the mapping process. A* is a search algorithm commonly used for finding the shortest path in a graph (Gul, Rahiman, & Nazli Alhady, 2019). The A* algorithm utilizes a heuristic function to determine the shortest path from the starting point to the target point (Le et al., 2018). The A* algorithm is similar to the Best First Search (BFS) algorithm but modified using a heuristic function (Yu & Xiang, 2017; Cheng & Wang, 2018). Thus, an optimal shortest route can be achieved. Equations 7 and 8 are used to calculate the cost function of the A* algorithm.

$$f(n) = g(n) + h(n) \qquad (7)$$

$$f(n) = a_0 + \sum_{n=1}^{\infty}\left(a_n + b_n \sin\frac{n\pi x}{L}\right) \qquad (8)$$

Where $f(n)$ is the function of evaluation, $g(n)$ is the cost incurred from the initial state to node n, and $h(n)$ is the estimation of costs incurred from state n or node n until reaching the destination. To calculate the distance between two distinct points in the Cartesian coordinate system, we employ a Euclidean distance-based equation such as elaborated in Equation 9,

$$h(n) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \qquad (9)$$

Table 1. Pseudo-code of A* algorithm used in mobile robotic arm

Algorithm A* (A-Star)
OPEN consists of nodes that have been visited but not expanded.
CLOSE consists of nodes that have been visited and expanded
start: start node
goal: goal node
m: current node
n: successor node
n_cost: successor current cost
1: Put *start* in the OPEN list with f(*start*) = h(*start*)
2: **while** the OPEN list is not empty
3:       Take from the open list the node *m* with the lowest
4:       f(*m*) = g(*m*) + h(*m*)
5:       **if** *m* is *goal* we have found the solution; **break**
6:       Generate each state *n* that come after *m*
7:       **for each** *n* of *m*
8:            Set *n_cost* = g(*m*) + w(*m*, *n*)
9:            **if** *n* is in the OPEN list
10:                **if** g(*n*) ≤ *n_cost* **continue**
11:            **else if** *n* is in the CLOSED list
12:                **if** g(*n*) ≤ *n_cost* **continue**
13:                Move *n* from CLOSED list to the OPEN list
14:            **else**
15:                Add *n* to the OPEN list
16:                Set h(*n*) to be the heuristic distance to *the goal*
17:            Set g(*n*) = *n_cost*
18:            Set the parent of *n* to *m*
19:       Add *m* to the CLOSED list
20: **if** (*m* != *goal*) exit with error

Where (x_1, y_1) is the initial coordinate and (x_2, y_2) is the final/target coordinate. The pseudo-code of the A* algorithm that we develop in our mobile robotic arm is elaborated in Table 1.

The robotic arm performs a loading/unloading mechanism with an attached gripper. End-effector positioning and movement are manipulated using the forward and inverse kinematics method with variability in height controller. In this research, the 4-DOF SCARA-like robotic arm is segmented into an RRR representation (joints 1, 3, and 4) with height control in joints 2 (prismatic), as presented in Figure 4. This model segmentation can benefit in reducing computation time than using a whole direct 4-DOF inverse kinematic computation (Zhen et al., 2020).

Inverse kinematic computation is required to map the known targeted point of the end-effector in Cartesian space to derive the angular position of motors in the joint space (Corke, 2017). In this research, the Jacobian Pseudo-Inverse method is used to find the solution to our inverse kinematic problem for the robotic arm, as it provides a better solution without using direct matrix inversion (Gupta et al., 2018). Jacobian pseudo-inverse is used to find each joint's angular position from the end-effector's cartesian position. The formulation and derivation of the solver are provided in Equations 10 to 13, with $\dot{X}$ is the state of end effector in cartesian space,

Figure 5. The mobile robotic arm navigates through the hallway to the room during the experimental environment.

$\dot{\theta}$ is the state in joint space, J is the Jacobian matrix, J^T is the transpose of Jacobian matrix, and J^+ is the pseudo-inverse Jacobian matrix.

$$\dot{X} = J(\theta)\,\dot{\theta} \tag{10}$$
$$J^T \dot{X} = J^T J(\theta)\,\dot{\theta} \tag{11}$$
$$(J^T J)^{-1} J^T \dot{X} = (J^T J)^{-1} J^T J(\theta)\,\dot{\theta} \tag{12}$$
$$\dot{\theta} = J^+ \dot{X} \tag{13}$$

In order to generate a smooth trajectory for the end-effector's movement in Cartesian space, we utilize a sigmoid trajectory method in the joint space. Sigmoid trajectory function is applied in the angular velocity control for each joint space; hence it provides the smooth actuation of the end-effector. This trajectory planning is implemented to guarantee the production of a smooth path movement of the end-effector from the initial to the targeted position (Fang et al., 2019).

3. Results and Discussion

Experiments are conducted in a small indoor building consisting of 2 fully-open rooms and a hallway. Half-opened rooms are added in the experiment to observe the robot's ability to detect the most available rooms to navigate. The testing environment is chosen to mimic the setting of a small-size clinic or a controlled environment in the food industry, as presented in Figure 5. The ground truth data is built by using a direct measurement with a standard precise instrument, as depicted in Figure 6. a. The visualization of the preliminary mapping experiment of the testing environment is presented in Figure 6. b. Small-size items with a volume of fewer than 10x10x10 cm³, such as small boxes, test tubes, and scissors, are used as experiment
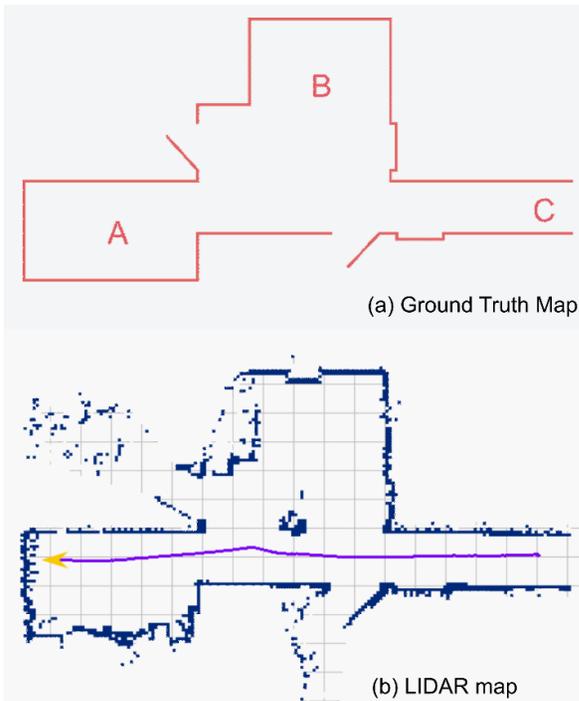
Figure 6. The comparison of (a) the ground-truth map and (b) the 2D LIDAR-generated map of the experimental environment for inter-room navigation and delivery.
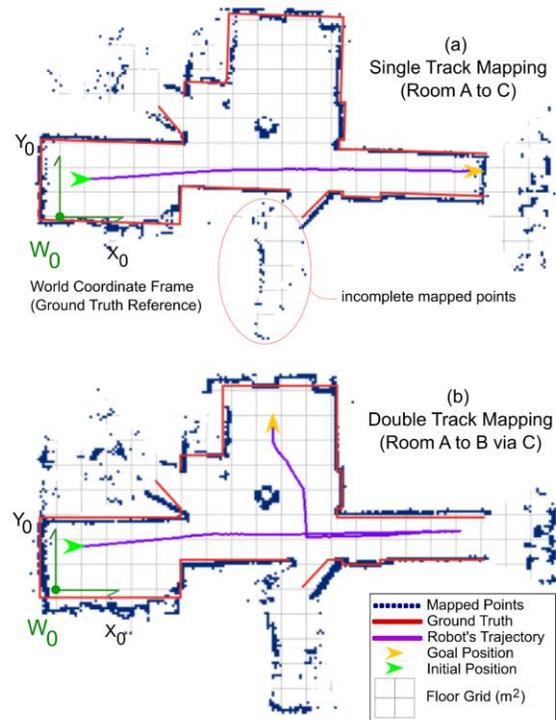

Figure 7. The comparison of generated maps resulted from (a) single track mapping and (b) double track mapping process utilizing the Hector SLAM algorithm in the prepared environment.

objects to load and unload by a robotic arm. The mobile robotic arm is equipped with a rack with a dimension of 25×25 cm² to carry some items altogether from an initial position to the goal position. The experiments are conducted in 4 main stages: (1) mapping, (2) path planning and tracking, (3) load and unloading, and (4) item delivery.

In the first experiment - mapping, the mobile robotic arm demonstrates the process of building a 2D map utilizing RPlidar 360 degrees scanning. In this step, the movement of the mobile robotic arm along the path/trajectory is guided via a remote computer running on ROS (Robot Operating System). The speed of the mobile robotic arm is set to 8 cm/s to perform optimal mapping (low speed). The raw map file of the environment consists of sparse 2D point clouds representing the distance of the walls and objects relative to the LIDAR sensor attached to the body of the mobile robotic arm. In parallel, this input is regulated by the Hector SLAM technique to generate the 2-dimensional map, as seen in Figure 6.b.

The mapping experiments are conducted in 2 parts: (1) single track, and (2) double tracks mapping. Both parts are intended to analyze the result of one-way mapping compared to a round trip mapping. The results

of the experiments are presented in Figures 7a and 7b. From the data presented in Figure 7, the double-track mapping has a higher density of point cloud maps by a factor of 160% compared to the single-track mapping experiment. However, since the same algorithm regulates both experiments with the same initial position, the accuracy of the generated map is almost the same ($\pm$10%) relative to the ground truth data.

The second experiment – path planning and tracking- is performed particularly to demonstrate and analyze a mobile robotic arm's path planning and tracking ability. This step is performed in 2 different scenarios. The first is navigation from position A to C - which is from a room to the hallway. The second is navigation from position A to C, then to B, which is started from a room to the hallway and then to another room, as visualized in Figure 7. As seen in Figure 7, the mobile robotic arm can successfully demonstrate autonomous navigation tasks without colliding with walls and obstacles detected in the environment.

In order to analyze the accuracy and precision of the path planning and tracking algorithm, 10 experiments are conducted for both scenarios (single-track and double tracks navigation). The data of path generation time, navigation time, and path length estimated from each

Table 2. Experiment data of mobile robotic-arm's path planning and tracking for navigation, experiment with a single track mapping and navigation

Exp No.	Path Generation Time [msec]	Navigation Time [sec]	Path Length [m]
1	1.938	18.921	4.181
2	1.872	17.719	4.164
3	1.823	16.821	4.187
4	1.865	16.996	4.177
5	1.901	18.703	4.162
6	1.848	17.112	4.184
7	1.899	17.395	4.181
8	1.928	16.794	4.169
9	1.905	18.223	4.172
10	1.834	16.851	4.166
Mean Value	**1.881**	**17.554**	**4.174**

Table 3. Experiment data of mobile robotic-arm's path planning and tracking for navigation, experiment with double tracks mapping and navigation

Exp No.	Path Generation Time [msec]	Navigation Time [sec]	Path Length [m]
1	4.331	46.763	13.362
2	4.628	47.752	13.321
3	4.619	44.949	13.357
4	4.446	45.848	13.335
5	4.393	46.621	13.354
6	4.557	44.545	13.359
7	4.379	47.213	13.361
8	4.423	44.834	13.352
9	4.627	46.333	13.351
10	4.325	44.902	13.329
Mean Value	**4.473**	**45.976**	**13.348**

experiment is presented in Table 2 and Table 3. The robot can maintain relatively high accuracy (less than 5% of position error) and high precision/repeatability (less than 2% of variations).

The shortest path generation time of both experiments is relatively fast (no more than 4.5 ms) and with a higher navigation speed (average of ±23 cm/s for single track and ±28 cm/s for double tracks experiment). In these experiments, the base speed is set to 23 cm/s. However, during the double-track navigation, the robot does not arrive precisely in the transit position. Hence it makes an early movement to leave the transit position to the goal position. Thus, it speeds up robot movement.

The third experiment – loading and unloading, is conducted to test and analyze the performance of the robotic arm in picking and placing items to perform load and unload tasks. Before the experiments are conducted, we carefully analyze the effective working space of the

Table 4. Result of the experiment for arrival location at the targeted location.

No	Arrival point [m]		Error [m]		Distance from Goal Position [m]	Success Rate
	X	Y	X	Y		
1	7.036	9.021	0.036	0.021	0.041	100%
2	6.927	8.904	0.073	0.096	0.121	100%
3	6.895	9.026	0.105	0.026	0.108	100%
4	7.092	8.896	0.092	0.104	0.139	100%
5	7.097	8.883	0.097	0.117	0.152	100%
6	6.952	9.043	0.048	0.043	0.064	100%
7	7.097	9.014	0.097	0.014	0.098	100%
8	6.904	9.034	0.096	0.034	0.102	100%
9	7.018	9.094	0.018	0.094	0.096	100%
10	7.076	9.073	0.076	0.073	0.106	100%
Mean Value	**7.009**	**8.999**	**0.074**	**0.062**	**0.103**	100%

robotic arm relative to its primary rotational axis, as visualized in Figure 9. The effective working space for picking and placing tasks is from 19 cm (inner-bound limit) to 37 cm (outer-bound limit). Subsequently, ten consecutive experiments of pick and place are conducted utilizing random items (small boxes, test tubes, and scissors), as presented in Figure 10. From all experiments, the robotic arm demonstrates 100% successful attempts in picking and placing items from the loading zone to the unloading zone. There is a relatively small position error in placing items (actuated point compared to the targeted point), with 3-dimensional position errors ranging from 0.6 to 2.6 mm. However, this positional error can be negligible as it is relatively insignificant compared to the size of the working space and unloading zone.

The fourth experiment – items delivery-demonstrates and analyzes the mobile robotic arm's capability to deliver items. This experiment utilizes the pre-generated map from the mapping process, which has been stored in the main processor. The initial position and the goal position is set manually by tele-operator, and the process of picking an item is performed using tele-operation with guidance from the camera. The mobile robotic arm begins to navigate autonomously with guidance from the LIDAR sensor and wheel odometry using a reference of a pre-generated map.

In this part of the experiment, the mobile robotic arm performs a 100% successful rate to deliver items to the unloading zone. However, during 10 conducted experiments, the arrival points of the robot relative to the targeted point slightly deviated, as presented in Figure 8 and Table 4. We analyze the accuracy of arrival points of the mobile robotic arm relative to the goal/targeted position by conducting 10 attempts of delivery experiment and then plotting the arrival points of the robot, as presented in Figure 8a. The robot's arrival
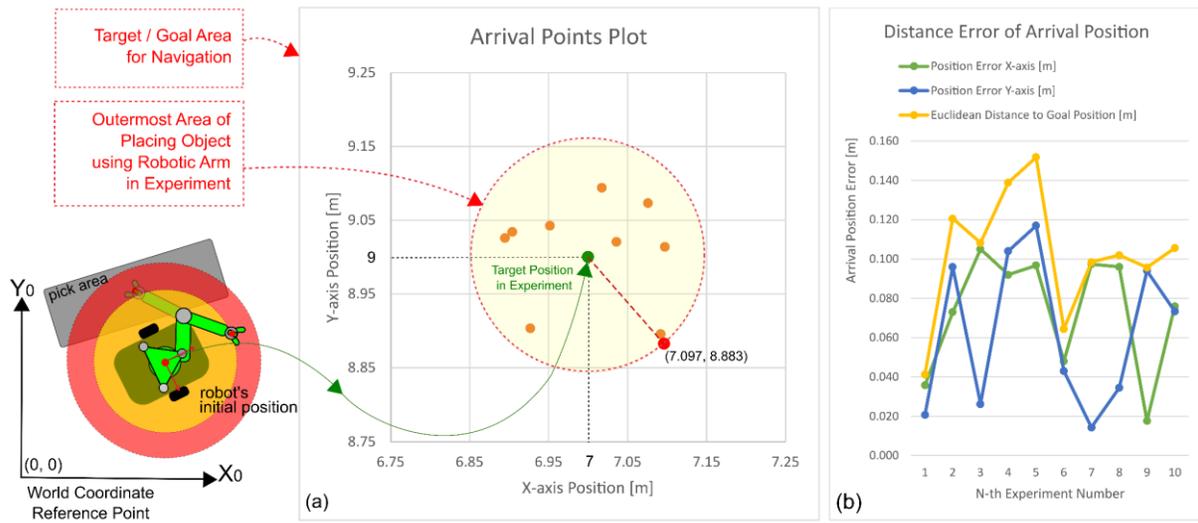
Figure 8. Experimental results of mobile robot's navigation in the context of (a) robot's arrival positions within 10 consecutive experiments. The data shows that the arrival positions vary subject to (b) the distance error relative to the goal/targeted point.
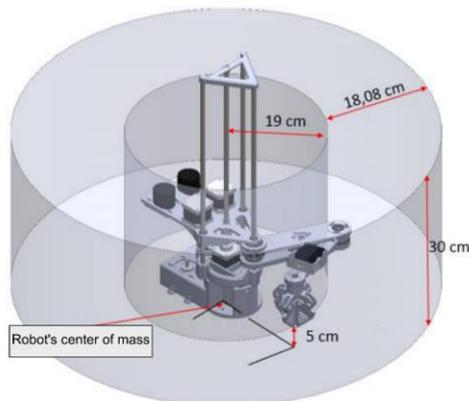

Figure 9. 3D effective working space for robotic arm relative to the main rotational axis of robot's torso.

Figure 10. The result of distance error measurement from the robotic arm's 10 pick and place experiments.

position data is needed to analyze the accuracy and precision of the mobile robotic arm arriving at the goal point. As specified in Table 4, the outermost arrival point from 10 experiments is at a distance of 15.2 cm (experiment number 5), and the closest arrival point is 4.1 cm to the goal point (experiment number 1). The average distance error of 10 consecutive experiments is 10.3 cm. By comparing to the effective working space of the robotic arm (as presented in Figure 9), which has a range of 18.08 cm, all the arrival points from those 10 experiments are still within the operating range of the robotic arm, and the process of pick and place can be performed successfully with 100% success rate.
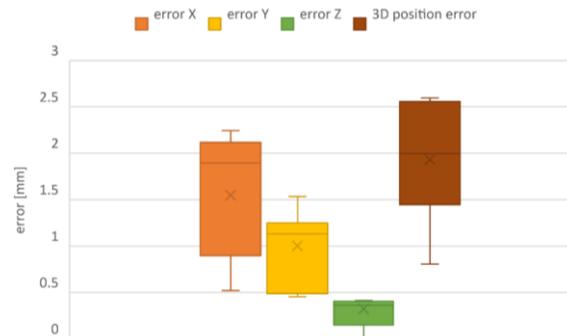
4. Conclusions

The experimental result of the mobile robotic arm mapping technique demonstrated a relatively precise generated map. The generated points' map is projected with low noises, as some noises have been filtered out using the Hector SLAM method. However, this mapping process should be conducted at a low speed with a stable robot movement, with a maximum speed of 10 m/s. A faster mapping speed is aimed at the future upgrade.

Our mobile robotic arm's path planning and tracking capability are relatively accurate and precise. The robot is able to maintain relatively high accuracy with less than 5% of position error, and it has high

repeatability with less than 2% of variations. From several experiments, the robot can successfully navigate the environment without bumping into the wall and obstacles. The navigation time is also quite fast with a stable motion. However, we tested our navigation capability in a relatively simple environment. Hence in the future, we will demonstrate navigation performance in a more complex environment approaching the real healthcare facilities in Indonesia.

The experimental result of the robotic arm also demonstrated accurate and precise pick and place tasks, with a recorded error rate of lower than 2.6 mm. This shows that our robotic arm can satisfy the application in the natural environment. However, the actuation speed is relatively low; to pick some items, our mobile robotic arm needs about 20 - 25 seconds to grab the object successfully. Therefore, we will upgrade the design of the joint motors with a higher speed specification.

In terms of the full capability of delivering items, we have demonstrated that our mobile robotic arm can perform successful mapping, path planning, tracking, load and unloading, and delivery tasks in a sequential process. The arrival position of the mobile robotic arm to the targeted area can be satisfied with a relatively minimum distance error. Thus, all tasks of delivery can be performed with a 100% successful rate. However, the pick and place tasks are still undergone in a guided mode using tele-operation while the navigation is done autonomously. In the near future, we aim to improve the level of autonomy in all aspects and processes, including the process of picking and placing objects.

References

Avgousti, S., Christoforou, E. G., Panayides, A. S., Voskarides, S., Novales, C., Nouaille, L., Vieyres, P. (2016). Medical telerobotic systems: current status and future trends. *Biomedical engineering online, 15*, 96.

Cheng, Y., & Wang, G. Y. (2018). Mobile robot navigation based on lidar. *2018 Chinese Control And Decision Conference (CCDC)*, (pp. 1243–1246).

Chitrakar, B., Zhang, M., & Bhandari, B. (2021). Improvement strategies of food supply chain through novel food processing technologies during COVID-19 pandemic. *Food Control, 125*, 108010.

Corke, P. (2017). *Robotics, vision and control: fundamental algorithms in MATLAB® second, completely revised* (Vol. 118). Springer.

Deng, H., Xiong, J., & Xia, Z. (2017). Mobile manipulation task simulation using ROS with Moveit. *2017 IEEE International Conference on Real-time Computing and Robotics (RCAR)*, (pp. 612–616).

Evans, C. R., Medina, M. G., & Dwyer, A. M. (2018). Telemedicine and telerobotics: from science fiction to reality. *Updates in surgery, 70*, 357-362.

Fang, Y., Hu, J., Liu, W., Shao, Q., Qi, J., & Peng, Y. (2019). Smooth and time-optimal S-curve trajectory planning for automated robots and machines. *Mechanism and Machine Theory, 137*, 127–153.

Gul, F., Rahiman, W., & Nazli Alhady, S. S. (2019). A comprehensive study for robot navigation techniques. *Cogent Engineering, 6*, 1632046.

Gupta, A., Bhargava, P., Agrawal, S., Deshmukh, A., & Kadam, B. (2018). Comparative Study of Different Approaches to Inverse Kinematics. *International Conference on Advances in Computing and Data Sciences*, (pp. 556–563).

Haidegger, T., Virk, G. S., Herman, C., Bostelman, R., Galambos, P., Györök, G., & Rudas, I. J. (2020). Industrial and medical cyber-physical systems: Tackling user requirements and challenges in robotics. In *Recent Advances in Intelligent Engineering* (pp. 253-277). Springer.

Harik, E. H., Korsaeth, A., & others. (2018). Combining hector slam and artificial potential field for autonomous navigation inside a greenhouse. *Robotics, 7*, 22.

Kolhatkar, C., & Wagle, K. (2021). Review of SLAM Algorithms for Indoor Mobile Robot with LIDAR and RGB-D Camera Technology. *Innovations in Electrical and Electronic Engineering*, 397–409.

Kwon, H., An, S., Lee, H.-Y., Cha, W. C., Kim, S., Cho, M., & Kong, H.-J. (2022). Review of Smart Hospital Services in Real Healthcare Environments. *Healthcare Informatics Research, 28*, 3–15.

Le, A. V., Prabakaran, V., Sivanantham, V., & Mohan, R. E. (2018). Modified a-star algorithm for efficient coverage path planning in tetris inspired self-reconfigurable robot with integrated laser sensor. *Sensors, 18*, 2585.

Magid, E., Zakiev, A., Tsoy, T., Lavrenov, R., & Rizvanov, A. (2021). Automating pandemic mitigation. *Advanced Robotics, 35*, 572–589.

Niemeyer, G., Preusche, C., Stramigioli, S., & Lee, D. (2016). Telerobotics. In *Springer handbook of robotics* (pp. 1085-1108). Springer.

Norzam, W. A., Hawari, H. F., & Kamarudin, K. (2019). Analysis of mobile robot indoor mapping using GMapping based SLAM with different parameter. *IOP Conference Series: Materials Science and Engineering*, *705*, p. 012037.

Olalekan, A. F., Sagor, J. A., Hasan, M. H., & Oluwatobi, A. S. (2021). Comparison of Two SLAM Algorithms Provided by ROS (Robot Operating System). *2021 2nd International Conference for Emerging Technology (INCET)*, (pp. 1–5).

Pandey, A., Pandey, S., & Parhi, D. R. (2017). Mobile robot navigation and obstacle avoidance techniques: A review. *Int Rob Auto J, 2*, 00022.

Shen, Y., Guo, D., Long, F., Mateos, L. A., Ding, H., Xiu, Z., . . . others. (2020). Robots under COVID-19 pandemic: A comprehensive survey. *IEEE Access, 9*, 1590–1615.

Sierra Marín, S. D., Gomez-Vargas, D., Céspedes, N., Múnera, M., Roberti, F., Barria, P., . . . Cifuentes, C. A. (2021). Expectations and Perceptions of Healthcare Professionals for Robot Deployment in Hospital Environments During the COVID-19 Pandemic. *Frontiers in Robotics and AI, 8*, 102.

Tavakoli, M., Carriere, J., & Torabi, A. (2020). Robotics, Smart Wearable Technologies, and Autonomous Intelligent Systems for Healthcare During the COVID-19 Pandemic: An Analysis of the State of the Art and Future Vision. *Advanced Intelligent Systems*, 2000071.

Yamamoto, T., Terada, K., Ochiai, A., Saito, F., Asahara, Y., & Murase, K. (2018). Development of the research platform of a domestic mobile manipulator utilized for international competition and field test. *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, (pp. 7675–7682).

Yu, H.-F., & XIANG, W. E. (2017). Application of A-Star Algorithm in Robot for Catering Service Based on Laser Radar Navigation. *DEStech Transactions on Computer Science and Engineering*.

Zhang, X., Fang, Z., Lu, Z., Xiao, J., Cheng, X., & Zhang, X. (2021). 3D Reconstruction of Weak Feature Indoor Scenes Based on Hector SLAM and Floorplan Generation. *2021 IEEE 7th International Conference on Virtual Reality (ICVR)*, (pp. 117–126).

Zhen, S., Zhao, Z., Liu, X., Chen, F., Zhao, H., & Chen, Y.-H. (2020). A Novel Practical Robust Control Inheriting PID for SCARA Robot. *IEEE Access*.