IJIE

UTHM Penerbit
Universiti Tun Hussein Onn Malaysia

INTERNATIONAL JOURNAL OF INTEGRATED ENGINEERING
ISSN: 2229-838X e-ISSN: 2600-7916

Vol. 18 No. 3 (2026) 311-326
https://publisher.uthm.edu.my/ojs/index.php/ijie

Enhanced Scan Matching Algorithm for Precise Autonomous Mobile Robot Localization

Marni Azira Markom[1,4]*, Abdul Hamid Adom[2,4], Erdy Sulino Mohd Muslim Tan[2,4], Arni Munira Markom[3], Ammar Zakaria[2,4]

[1] *Faculty of Electronic Engineering & Technology, Universiti Malaysia Perlis, Kampus Pauh Putra, 02600 Arau, Perlis, MALAYSIA*
[2] *Faculty of Electrical Engineering & Technology, Universiti Malaysia Perlis, Kampus Pauh Putra, 02600 Arau, Perlis, MALAYSIA*
[3] *School of Electrical Engineering, College of Engineering, Universiti Teknologi MARA, 40450 Shah Alam, Selangor, MALAYSIA*
[4] *Center of Excellence for Advanced Sensor Technology, Universiti Malaysia Perlis, Kampus Pauh Putra, 02600 Arau, Perlis, MALAYSIA*

*Corresponding Author: marni@unimap.edu.my
DOI: https://doi.org/10.30880/ijie.2026.18.03.021

Article Info

Received: 31 July 2025
Accepted: 4 March 2026
Available online: 30 April 2026

Keywords

Mobile robot, indoor localization, scan matching, localization algorithm

Abstract

Mobile robot localization is essential for autonomous navigation, allowing robots to accurately determine their positions within an environment. Traditional scan matching algorithms primarily only provide an area estimation of the robot's location without the exact coordinates within its environment. This paper presents a work on improving the scan matching localization algorithm using K-nearest neighbour (KNN), that is able to provide 2D coordinate information of the mobile robot position (x,y). The mobile robot with an RP lidar and high capability of a computer system is used in this study. Typical algorithms often only encompass input preparation, scan matching, and area classification stages. The proposed localization algorithm introduces a fourth stage aimed at determining the coordinates of the mobile robot within its environment. Three distance parameters (Euclidean, Mahalanobis, and Manhattan), are investigated to determine the optimal choice for KNN. The proposed improved algorithm employing Euclidean distance achieved an accuracy of 93% when classifying 100 non-reference test samples within a range of 4cm area. Furthermore, the mobile robot continuously maintains a coordinate determination accuracy of less than 3cm in 30 samples across all local maps. These findings hold promise for applications requiring precise mobile robot localization.

1. Introduction

In recent decades, the adoption of mobile robot systems has experienced a remarkable upswing, serving as valuable aids in diverse applications to support humans. Particularly, these robots have proven indispensable in tackling hazardous and volatile environments, where human intervention would be impractical or risky. Additionally, the deployment of mobile robots brings forth a host of distinct advantages, notably enhanced accuracy and efficiency in various operational tasks.

Mobile robots have found diverse and impactful applications across various fields. In the application of home appliances, robotic vacuum cleaners like Roomba have become a staple for automated cleaning, providing convenience and efficiency [1]. The military sector has embraced mobile robots for deployments in dangerous and hostile environments, where they undertake surveillance, reconnaissance, and hazardous tasks, reducing risks to human personnel [2]. In exploration missions, mobile robots have become pioneers, venturing into uncharted territories on distant planets, moons, and even underwater environments, gathering valuable data and insights [3-4]. Furthermore, mobile robots play a vital role in search and rescue operations, efficiently navigating through disaster-stricken areas to locate and assist individuals in distress, saving valuable time and lives [5].

Mapping and localization are essential components within mobile robot systems, that enable autonomous movement in specific environments. These two components are often integrated together, enhancing the capabilities and functionality of mobile robots. For instance, in order for a mobile robot to navigate to a target location, it requires information about its own position as well as the target's position. This necessitates obtaining environmental information, which can be represented in the form of a map. By leveraging the map, the mobile robot can accurately determine its location within the environment and estimate the optimal path to reach the target. This example vividly illustrates the critical role of mapping and localization in enabling mobile robots to autonomously navigate and operate within their surroundings.

One widely adopted approach in this domain is scan matching, which holds immense promise for estimating the motion of mobile robots and determining their positions [6-8]. Scan matching involves identifying the relative positions and orientations of line segments from different scans, facilitating the measurement of similarity and consistency between them. Through scan matching, researchers can assess the quality of the scanning process and the accuracy of the localization system, thereby enhancing overall performance.

A limitation of the scan matching approach for mobile robot localization algorithms is its sensitivity to environmental changes and sensor noise. Scan matching relies on aligning the robot's sensor scans with a reference map to estimate its pose accurately. However, in real-world scenarios, environmental conditions can change, and sensor readings may contain noise or errors, leading to inaccuracies in pose estimation. Additionally, most of the scan matching results only provide a position of an area in the environment, but, for certain mobile robot applications, specific coordinates of a target and mobile robot position are required. In certain environments in which the interaction between the mobile robot and objects is unavoidable, the exact location or positioning of the former is crucial. Consequently, these challenges can result in reduced localization accuracy and may require additional techniques or sensor fusion methods to improve the performance of the localization algorithm.

Furthermore, researchers have explored the use of a single sensor for mobile robot mapping and localization systems, benefiting from advancements in sensing modality technology that offer high accuracy and reliability. Among the commonly used sensors are ultrasonic and laser range finders. While ultrasonic sensors require multiple units in the system to achieve 360-degree scanning of the surrounding area, the process can be slow due to signal transmission between the transceivers, hindering real-time performance [9]. On the other hand, laser range finders have gained popularity due to their fast-sampling processing, acquiring up to 2000 data points per second. However, their high cost, especially for long scanning ranges, poses a limitation. An intriguing alternative is the RP Lidar, a relatively inexpensive laser range finder developed by Robopeak from China, offering a scanning range of up to six meters at a much lower cost [10].

Considering these factors, this study provides significant insights into a mobile robot system that leverages the RP Lidar, a cost-effective laser range finder with a scanning range of up to six meters, combined with an enhanced scan matching localization algorithm. By integrating these advancements, the proposed system supports real-time operation in various indoor environments, both known and unknown, under static conditions. Overall, this study's contributions enhance the efficiency and cost-effectiveness of mobile robot mapping and localization systems, making them suitable for practical applications in real-world scenarios.

2. Methodology

This research methodology is designed to have a few phases of work. It includes mobile robot sensing modality development along with the corresponding localization algorithms, construction of local and reference maps, and testing.

2.1 Mobile Robot Sensing Modality Setting

In this study, a mobile robot equipped with an RP Lidar was developed, as shown in Figure 1. Key components include a power system, an Intel quad-core processor with 4GB RAM, a wireless system, a laser range finder, and a navigation module. The robot measures 20 cm (W) x 30 cm (L) x 25 cm (H) and has three layers: the bottom houses the power supplies and mobility components like the motor and driver, the middle holds the computer, and the top is fitted with the lidar sensor. The system runs on Windows, with the lidar scanning data collected and monitored via a custom software-development-kit (SDK) provided by lidar manufacturer. The motor driver is a dual-channel unit capable of delivering up to 30A per channel, providing sufficient power for precise control. The robot is equipped with 4-inch diameter rubber wheels, offering strong traction and stability on various surfaces. This design and setup enhance the robot's capabilities in laser range finder image transmission and computation, necessitating the integration of a high-performance processor.

The study utilizes the RP Lidar laser range finder, which is strategically mounted atop the mobile robot, as depicted in Figure 1. The RP Lidar, a low-cost laser range finder developed by Robopeak in 2012, offers a remarkable scanning range of up to 6m with 5.5hz rotating frequency, and able to provide 2000 samples per second. Leveraging laser triangulation ranging principles and employing high-speed vision acquisition. By affording a 360-degree scanning view, the RP Lidar significantly enhances the mobile robot's perception capabilities.

The computer unit, equipped with a motherboard featuring an Intel quad-core processor and 4GB of RAM, is suitable for handling the computational demands of the localization method. The power supply is connected using a high-power direct insertion DC-ATX power module. This computer allows each scan to be processed in around 150 milliseconds. This ensures that the system can manage real-time data acquisition from the RP Lidar without experiencing substantial delays, resulting in precise and responsive localization for the mobile robot.

Next, to ensure reliable and sustained power supply, the mobile robot's power system incorporates two batteries, each with a capacity of 7200 mAh: a 12V DC battery dedicated to powering the wheels and motor driver, and a 24V DC battery used for the computer and laser range finder. This dual-battery configuration allows the robot to operate for approximately 4 to 6 hours under typical conditions, with the power system designed to efficiently meet the demands of both mobility and computational tasks, maximizing the robot's overall performance and operational longevity.

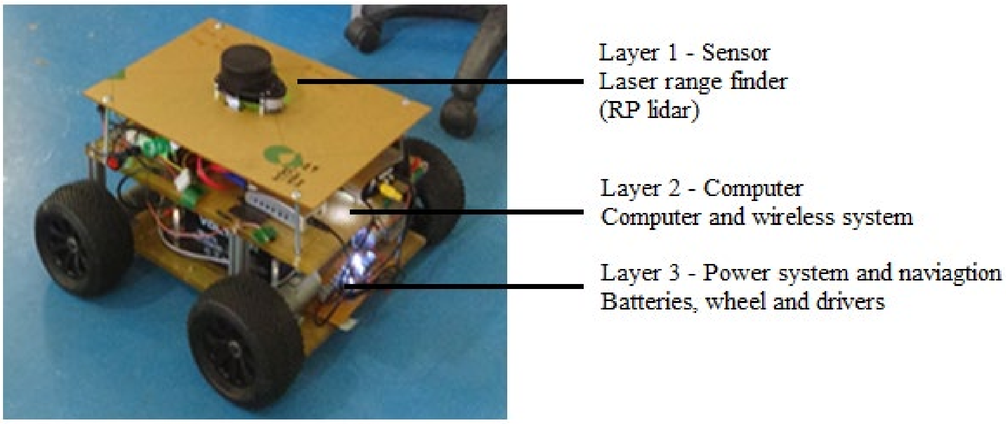


Fig. 1 *The mobile robot with RP lidar*

2.2 The Improved Scan Matching Algorithm

Development of improved K-nearest neighbors (KNN) scan matching algorithm is presented in two sub-sections: i. description of distance parameters for the algorithm, and, ii. the localization algorithm development. The details of the works are described in this section.

2.2.1 Distance Parameters for Localization Algorithm

This study employs three commonly used distance parameters, namely, Euclidean [20], Manhattan [21], and Mahalanobis [22], within the KNN scan matching algorithm. These parameters, which use different methods for measuring dissimilarity, produce varying values even for identical inputs. Their selection is crucial for determining the algorithm's classification accuracy and overall performance. By using these different metrics, the study thoroughly investigates their impact on classification outcomes and evaluates the KNN method's effectiveness.

The Euclidean distance calculates the straight-line distance between samples (x_m, y_m) and (x_n, y_n). Mahalanobis distance provides a normalized measure by evaluating distance while accounting for sample variances (s_m and s_n). Manhattan distance measures distance by adding the absolute differences between sample coordinates (x_m, y_m) and (x_n, y_n).

The best performance is defined as the closest distance between the computed location and reference points. Choosing the proper distance parameter based on data attributes and issue requirements can have a considerable impact on KNN classification performance and accuracy.

Euclidean distance:

$$euc_{m,n} = \sqrt{(x_m - x_n)^2 + (y_m - y_n)^2} \tag{1}$$

Mahalanobis distance:

$$mah_{m,n} = \sqrt{\frac{(x_m - x_n)^2 + (y_m - y_n)^2}{{s_m}^2 + {s_n}^2}} \tag{2}$$

Manhattan distance:

$$man_{m,n} = |x_m - x_n| + |y_m - y_n| \tag{3}$$

Where m and n are a number of new samples and references, respectively. x is the data sample at x-axis, y is the data sample at the y-axis, and s is the standard deviation.

2.2.2 Development of KNN Scan Matching Localization Algorithm

The development of the localization algorithm incorporating KNN and scan matching was carried out in four distinct stages within this study. The four stages are continuity works, which are performed sequentially from Stages 1 to 4. Figure 2 shows two block diagrams of algorithms: i. the typical KNN scan matching, and ii. the improved KNN scan matching. The flow and progression of the algorithm's stages are illustrated in Figure 3 to 6, providing a visual representation of the sequential steps employed in the localization algorithm incorporating KNN and scan matching as well as the specific of mobile robot coordinate.

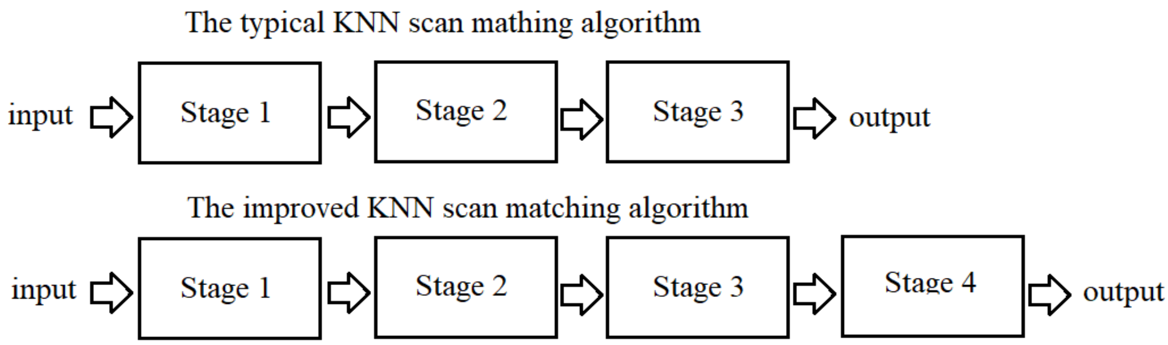


Fig. 2 *The block diagrams of a typical scan matching algorithm utilizing conventional KNN with three stages and the proposed improved version with four stages*

Stage 1 focuses on input preparation, beginning with the acquisition of scanning environment data from the RP lidar. This device captures distance (d) and angle (θ) measurements, which are then converted into a two-dimensional coordinate system (x,y). Following this conversion, both raw and smooth filtering processes are applied to refine the data. Specifically, only data from new samples and local map references with similar angles are processed, enhancing the efficiency of subsequent stages. The filtered data forms the output of this stage.

Stage 2 encompasses the scan matching process, wherein the distance parameters played a critical role. Each tested distance parameter was integrated into the algorithm, facilitating the alignment of the scans with local maps, which is in this study 10 local maps have been initialized in the previous work. The initial local map data is known as reference data. The results obtained from the scan matching process were organized based on the corresponding local maps, contributing to the subsequent stages. The input of this stage is the filtered data from Stage 1, and the output is the distance data, and they are arranged according to the local maps.

In Stage 3, the results of the scan matching process underwent classification using the KNN method, enabling the association of each scan with the most suitable map reference. To be specific, the input is the distance data that has been arranged according to the 10 local maps. Data in each local map are summed up, and the least value is decided as the result of area. This classification step further refined the localization accuracy by leveraging the collective information from the local maps. Also, in this stage, there is a rotation and translation process. This process will be turned on when the scanning data does not obtain 90% accuracy when matching to all local maps reference data. It starts by rotating the scan in 2^o steps. At every step, the accuracy with respect to the reference maps is computed. It will stop when the accuracy is 90%. At this angle, the mobile robot finds its location as well as orientation to the references. Figure 7 demonstrates the rotation process.

Stage 3 procedures are also known as the final stage of scan matching, as shown in [7]. The limitation of this step is that it does not determine the mobile robot's position, leading to a lack of information. This study addresses this issue by introducing Stage 4.

Lastly, Stage 4 involves mobile robot coordinate determination. This stage utilizes the information obtained from the previous stages to precisely determine the mobile robot's position by a 2D coordinate (x and y) within the environment. As the previous stage has provided the result for the area of the local map, then, the coordinate of the mobile robot can be determined by the horizontal plane (x-axis) and followed by the vertical plane (y-axis). For the x coordinate determination, the environment is divided into two sides, left and right as the mobile robot position is in the middle of the environment.

To determine the mobile robot's coordinates, the difference between corresponding values from the input and reference data is calculated. This difference indicates the robot's position relative to the environment: a positive difference suggests the robot is on the left side, while a negative difference indicates the right side. Three conditions are used to finalize the x-coordinate: if the difference matches the input value, the x-coordinate remains unchanged; if the difference is positive, it is added to the input value; otherwise, it is subtracted from the input value. For the y coordinate, the output from Stage 3, which carries the decided local map, is taken to be the input. The input data will be added with the reference data of the local map to get the y coordinate.

The definitions of essential variables are as follows. In Stage 1, two sets of variables are involved, $s(s_i, s\theta_i)$ representing scanning data and $Rn(r_i, r\theta_i)$ representing reference data, each containing distance and degree values. Next, $s\theta_j$ and $r\theta_j$ represents the degree values in radian. Additionally, sx_j, sy_j, rx_j and ry_j denote variables for both scanning and reference data in the x and y coordinates. Subsequently, $s(sx_k, sy_k)$ and $Rn(rx_k, ry_k)$ represent filtered data derived from the scanning and reference datasets, respectively. In Stage 2, the variables $euc_{m,n}$, $man_{m,n}$ and $mah_{m,n}$ represent the distance values for Euclidean, Manhattan, and Mahalanobis distances. Here, m and n correspond to the local map and the specific data number. Next in Stage 4, x_r and y_r are mobile robot coordinate of x and y. dx_i is the difference between scanning data, sx_i, and reference data, rx_i. The ith is a group of scanning data and reference data.

Stage 1: Input preparation

1. Get current input from the scanning, $S(s_i, s\theta_i)$.
2. Acquire the reference input from the database, $Rn(r_i, r\theta_i)$, where n is the number of local maps.
3. Compare angle between the input, $s\theta_i$ and reference, $r\theta_i$.
   If $s\theta_i = r\theta_i$, then, convert to radian from degree using (4) and (5).
   $$s\theta_i \text{ rad} = \frac{\pi}{180} \times s\theta_i \quad (4)$$
   $$r\theta_i \text{ rad} = \frac{\pi}{180} \times r\theta_i \quad (5)$$
   Else, remove $s\theta_i$ and $r\theta_i$ from input array.
4. Convert the current input, $S(s_i, s\theta_i)$ and reference, $Rn(r_i, r\theta_i)$ into two dimensions, $S(sx_i, sy_i)$ and $Rn(rx_i, ry_i)$ using (6), (7), (8), and (9), respectively.
   $$sx_i = s_i \cos(s\theta_i) \quad (6)$$
   $$sy_i = s_i \sin(s\theta_i) \quad (7)$$
   $$rx_i = r_i \cos(r\theta_i) \quad (8)$$
   $$ry_i = r_i \sin(r\theta_i) \quad (9)$$
5. Run raw filter to remove unwanted data, such as overshot and black data.
   For raw filter, a threshold is set up within a range of 0m to 6m. This range is determined by the laser range finder range. Any data falls in the range, it is consisted as valid data. Otherwise, it is invalid and considered as 0. Raw filter is conducted using (10) and (11).
   For $S(sx_i, sy_i)$
   $$S(sx_i, sy_i) = \begin{cases} 0, & if\ S(sx_i, sy_i) < 0\ and\ S(sx_i, sy_i) > 6m \\ S(sx_i, sy_i), & else \end{cases} \quad (10)$$
   End
   For $Rn(rx_i, ry_i)$
   $$Rn(rx_i, ry_i) = \begin{cases} 0, & if\ Rn(rx_i, ry_i) < 0\ and\ Rn(rx_i, ry_i) > 6m \\ Rn(rx_i, ry_i), & else \end{cases} \quad (11)$$
   End
6. End

Fig. 3 *The Stage 1 localization algorithm, input preparation*

Stage 2: Scan matching process

1. Compute distances values, $euc_{m,n}$, $man_{m,n}$ and $mah_{m,n}$ from $s(sx_k, sy_k)$ and $Rn(rx_k, ry_k)$ using (1), (2) and (3). Where m is the number of local maps and n is number of input data.
2. Arrange the distance values according to the local maps.
3. End.

Fig. 4 *The Stage 2 of localization algorithm, scan matching process*

Stage 3: Area classification using KNN

1. Sum up the distance values according to the local maps.
   for m = 1 to 10
   sum_euc_localmap(m) = $\sum_{n=1}^{N} euc_n$ (12)
   sum_man_localmap(m) = $\sum_{n=1}^{N} man_n$ (13)
   sum_mah_localmap(m) = $\sum_{n=1}^{N} mah_n$ (14)
   end
2. Determine the least_values of local maps summation.
3. Compute the percentage of accuracy of the classified local map.
   if percentage of accuracy < 90%
   Result = Go to 4 (rotation and translation);
   elseif percentage of accuracy ≥ 90%
   Result = Go to 5 (classification result);
   end
4. Rotation and translation
   for i = 1 to max(i)
   $s\theta_i = s\theta_i + 2^\circ$; (15)
   end
   Go to Stage 1;
5. Classification result using rule-based decision
   if least_value = localmap1
   Final_result = Local Map 1;
   elseif least_value = localmap2
   Final_result = Local Map 2;
   ⋮
   ⋮
   elseif least_value = localmap10
   Final_result = Local Map 10;
   else
   Final_result = undefined;
   end
6. End

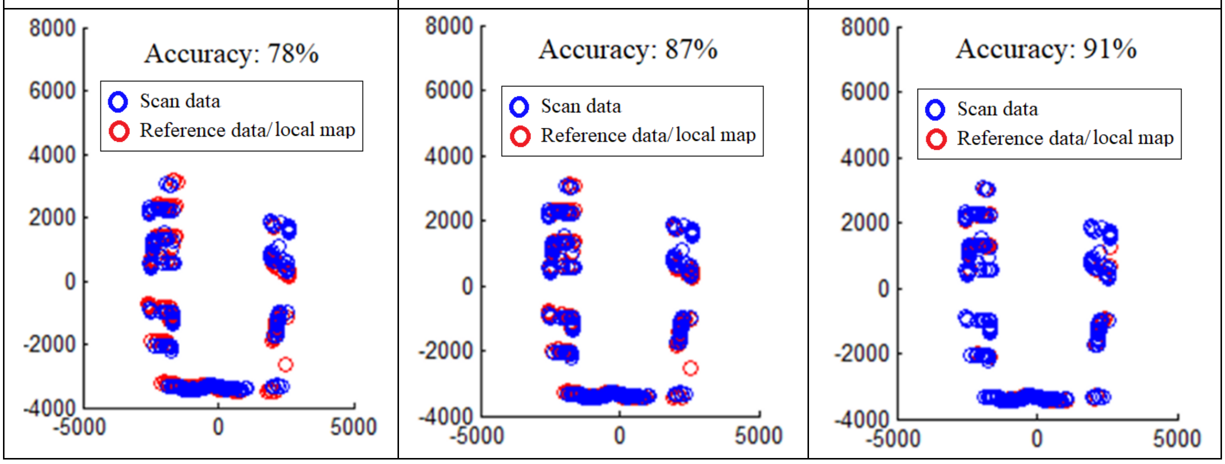
Fig. 5 *The Stage 3 of localization algorithm, area classification using KNN*

Stage 4: Mobile robot coordinates determination

1. For coordinate x_r (cm)
   Divide the environment into two sides, right (+) and left (-)
   for i = 1 to 10
   $dx_i = sx_i - rx_i$ (16)
   $dx_r = \text{mean}(dx_i)$
   if $dx_r = 0$
   $x_r = sx_i$ (17)
   elseif $dx_r = +ve$
   $x_r = sx_i + dx_r$ (18)
   elseif $dx_r = -ve$
   $x_r = sx_i - dx_r$ (19)
   end
   end
2. For coordinate y_r (cm)
   $y_r = sy_i + ry_i$ (20)
   ry_i is obtained from Final_result of Stage 3.
   if Final_result = localmap1
   $y_r = sy_i + 1000$ (21)
   elseif Final_result = localmap2
   $y_r = sy_i + 2000$ (22)
   ⋮
   ⋮
   elseif Final_result = localmap10
   $y_r = sy_i + 10000$ (23)
   end
3. Mobile robot coordinates, (x_r, y_r)
4. End

Fig. 6 *The Stage 4 of localization algorithm, mobile robot coordinate determination*

Initial scanning	2nd step of rotation (+2°)	3rd step of rotation (+4°)



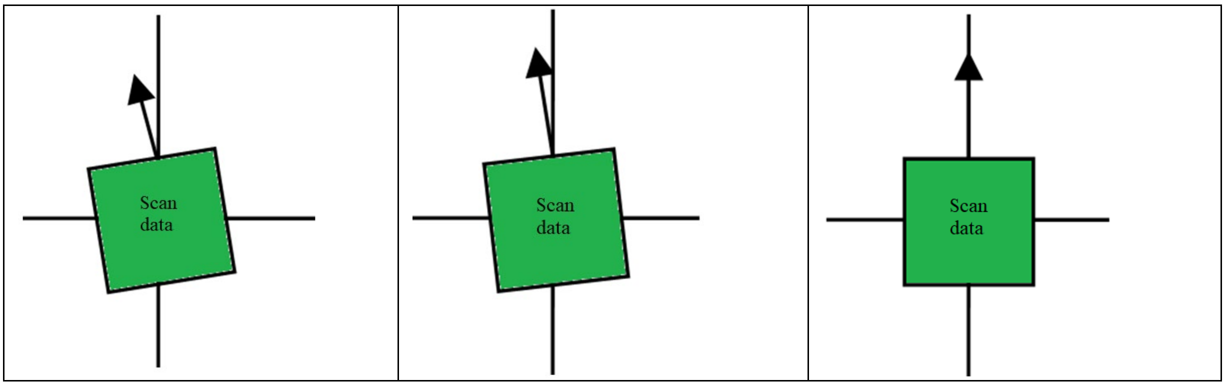


Fig. 7 *An example of the rotation and translation process in Stage 3, with initial scanning, 2°, and 4° rotation steps. The red circle in the top figures refers to the reference map and the blue circles refer to the scanned data. As the scanned data is rotated by 2-degree steps, its error from the reference map is calculated*

2.3 Development of Reference Maps

To implement the localization algorithm effectively, constructing a reference or prior map is essential to ensure accuracy. Initially, the mobile robot performs an exploratory maneuver to scan the entire environment, including all surrounding walls or borders. For instance, using a laser rangefinder in a room requires the robot to cover every wall for a comprehensive scan. This is addressed through area segmentation [24-25], with the scan results forming the reference map [26].

To enhance accuracy and reliability, each node is scanned 20 times to increase data points and improve consistency. Statistical measures like standard deviation, average, median, and mode are then used to assess this consistency, ensuring the local maps serve as reliable references for new scans. Once the local maps are created, they are merged based on their node positions to form the reference map. This process includes filtering the data with moving averages to produce a unified local map that accurately represents the environment in a 2D view, capturing obstacles and walls.

The study was conducted in three different environments to validate the theory and the newly developed localization algorithm. As illustrated in Figure 8, the research room, highlighted within the red box, with dimensions of 4m (W) x 12m (L), is used as the primary environment for demonstrating the workflow in this paper. This environment, free of moving obstacles and equipped with fixed equipment, offers a clear and consistent space for data collection and testing.

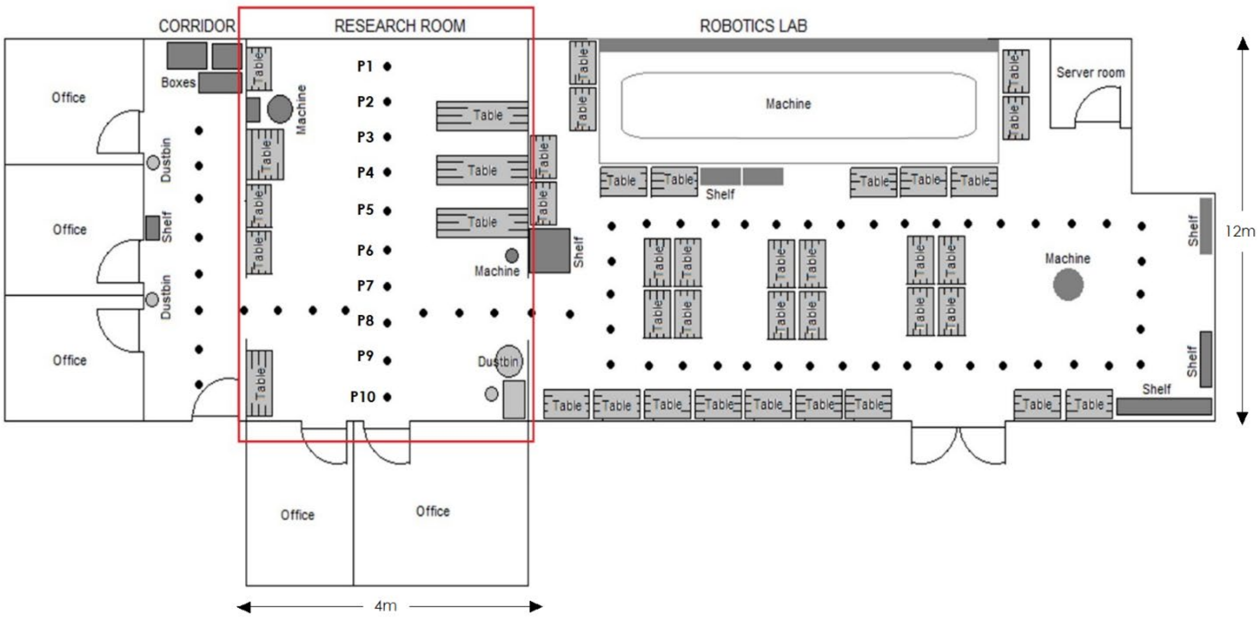


Fig. 8 *The illustration of the studied environment. The red box shows the area used for the demonstration work of the proposed algorithm*

As there is no predefined formula for determining the optimal number of segments, initial testing was conducted to explore various segmentations. The research room was divided into several segment configurations: two, four, 10, 16, and 32 segments. Each segment consists of a scanning node representing the mobile robot's location during the scanning process and corresponds to a distinct class in the classification stage. Through this testing, it was found that segmenting the environment into 10 segments provided the most significant difference. Therefore, this study employs a 10-segment configuration to demonstrate the workflow and evaluate the localization algorithm's performance.

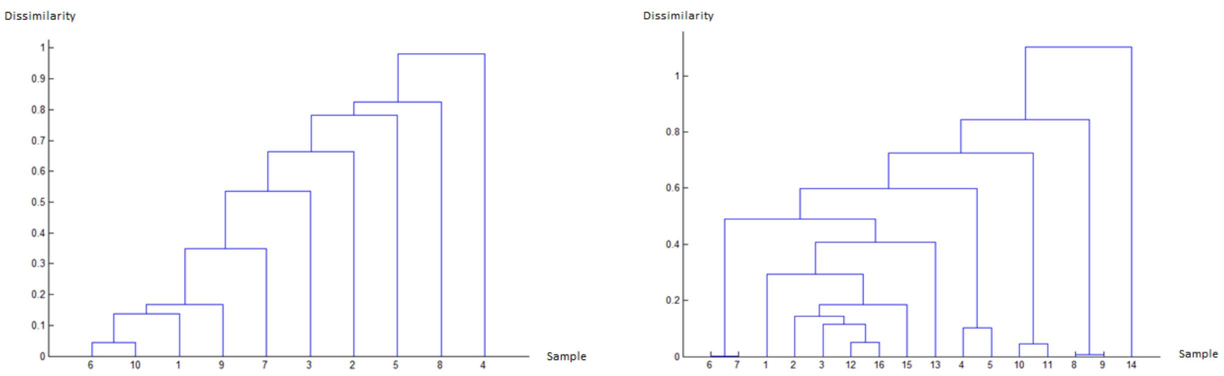
In the segmentation process, scanning points were strategically placed at the center of the environment. Figure 8 depicts these scanning nodes corresponding to the 10 segments within the environment, while Table 1 details the segment representations and their respective robot coordinates.

To further validate the segmentation, Multivariate Analysis of Variance (MANOVA) and dendrogram analysis were applied to assess the dissimilarity between environment segments [27-28]. The MANOVA results confirmed that all segmentations successfully passed the dissimilarity test, with p-values approaching zero, indicating that the segments are different and distinguishable. The dendrogram analysis further revealed that the 10-segment configuration provided a significant difference compared to other segmentations.

The comparison of dendrograms focused on the significance of differences and balanced distribution. To illustrate this more clearly, dendrograms with 10 and 16 segments, as shown in Figure 9, are used as comparison examples. This figure shows the 10-segment dendrogram demonstrates a gradual and balanced increase in dissimilarity, with clusters forming incrementally and a more uniform distribution of differences among samples. In contrast, the 16-segment dendrogram reveals a more uneven structure, with certain samples, such as those labeled 14 and 9, exhibiting significant dissimilarity. Therefore, the 10-segment dendrogram provides a clearer and more consistent representation of sample relationships, making it the better choice for capturing subtle variations and the overall structure in the data. Then, may lead to more accurate classification.

Table 1 *The positions of the scanning points for the environment with 10 segments*

The scanning node	Representation of segment	Representation of mobile robot coordinates in the environment (x, y) (cm)
P1	Segment 1	(0, 100)
P2	Segment 2	(0, 200)
P3	Segment 3	(0, 300)
P4	Segment 4	(0, 400)
P5	Segment 5	(0, 500)
P6	Segment 6	(0, 600)
P7	Segment 7	(0, 700)
P8	Segment 8	(0, 800)
P9	Segment 9	(0, 900)
P10	Segment 10	(0, 1000)



(a) (b)

Fig. 9 *Dendogram results, (a) 10 segments of environment; and (b) 16 segments of environment*

Int. Journal of Integrated Engineering Vol. 18 No. 3 (2026) p. 311-326

2.4 Testing

The testing aimed to evaluate the real-time navigation capabilities of the mobile robot in a controlled environment. A target location was set within the laboratory, and the mobile robot was positioned at a random starting point. The robot's task was to autonomously navigate to the target location using real-time data provided by the RP Lidar, which continuously scanned the environment. This lidar data was processed to update the environment map, allowing the robot to effectively avoid obstacles and make informed decisions based on the most up-to-date information, thereby demonstrating its navigation capabilities. For the evaluation results, as the robot moved towards the target, the monitoring system provided a path view, local map classification, and real-time coordinate updates, ensuring comprehensive feedback on the robot's navigation performance.

For this testing, the focus was on evaluating two localization algorithms: (i) the conventional KNN scan matching algorithm, and (ii) a newly improved KNN scan matching with coordinate algorithm. As stated in the previous section, the conventional scan matching algorithm was limited to Stages 1 to 3, as illustrated in Fig. 3 to 5, while the improved KNN scan matching with coordinate algorithm extended to Stage 4, as shown in Fig. 3 to 6. The conventional KNN scan matching algorithm concentrated on segment classification, providing information about the general area of the mobile robot's position. In contrast, the improved KNN scan matching with coordinate algorithm offered precise coordinates for the mobile robot's position.

Two type of data were used in this testing, i. reference data, and ii. non-reference data. Reference data, which consists of scanning data already included in the database, allowed the robot to operate in a familiar environment, enabling the assessment of the algorithm's ability to correctly classify samples and establish benchmarks for accurate localization. Non-reference data, representing new scanning environments not included in the database, was used to evaluate the algorithm's performance in generalizing to unseen data, ensuring that the KNN model could adapt effectively beyond the pre-existing environment data.

The first test aimed to evaluate the performance of conventional localization algorithms. This test involved 10 reference locations, with 10 trials conducted per location, resulting in a total of 100 input samples. The algorithm's performance was measured as a percentage of accuracy, calculated by dividing the number of correct classifications by the total number of samples. In the second test, the mobile robot's coordinates were determined, and the distance between the Stage 4 results and the actual coordinates was calculated. To thoroughly assess the algorithm's precision, various distance ranges were prepared, specifically ±1cm, ±2cm, ±3cm, ±4cm, ±5cm, and ±10cm. Similar to the first test, 100 samples were used for each distance range to evaluate the precision of the mobile robot's positioning. Finally, a comprehensive test was conducted to evaluate the complete process from Stage 1 to Stage 4, utilizing the best-performing distance parameter.

This test focused on analyzing the results of local map classification from Stage 3, the mobile robot's coordinates from Stage 4, and the corresponding distance ranges. This rigorous testing process confirmed the algorithm's ability to accurately determine the mobile robot's position in the room, ensuring reliable localization in both known and unknown environments.

3. Results and Discussion

In this section, the performance and results of the work are shown. It includes the work of construction of a reference map and assessment of its consistency and the performance of improved KNN scan matching localization results using reference and non-reference data.

3.1 Reference Map Development Results

In the methodology section, the work on the construction of a reference map involved multiple scanning to provide multiple data at one point, and then, to evaluate the consistency of the scanning data. This work presents that the laser range finder collected approximately 360 data points during one scanning rotation, which offers sample data for analysis. Even with some occasional blank values in the laser range finder data, these issues are less critical compared to other range sensors, like ultrasonic sensors.

Table 2 shows standard deviation values based on the scanning data for local maps at various angles. The majority of scanning data exhibit small standard deviation values, typically less than 10, which is considered small due to the wide scanning range of the laser range finder, spanning from zero up to 6m. The high value of standard deviation is due to the overshot scanning results and the blank values. This can be improved using filters to remove unwanted values. Here, the blank and high values of standard deviation are shaded as shown in Table 2. Next, Table 3 shows the standard deviation values after the scanning data went through the filtering process. The result after filtering shows the standard deviation values have improved by showing the decreasing of its values. The improved values are shaded in blue color.

On the contrary, the scanning data can be harnessed to generate a detailed 2D representation of the local map, as showcased in Figure 10. Once these local maps are seamlessly assembled, they can be integrated to create a comprehensive global map. The merging process is carried out based on their corresponding node positions, leading to the creation of the finalized reference map, as presented in Figure 11. Additionally, the scanning data is illustrated before and after the filtering process. This comprehensive global map impeccably portrays the actual environmental layout, encompassing obstacles and walls, in 2D view.

Table 2 *Standard deviation results (σ in cm) based on 20 times of scanning for segment 1 (local map 1) to segment 10 (local map 10). The blanks and high values of standard deviation are shaded*

Local Map	Angles (°)										
	30	60	90	120	150	180	210	240	270	300	330
LM1	12.14	1.41	1.50	1.24	3.89	0.55	2.12	12.58	0.86	2.21	35.02
LM2	26.86	14.45	4.03	3.88	22.17	1.28	4.09	8.04	0.52	2.41	10.50
LM3	15.13	12.48	5.53	6.81	11.97	10.57	9.70	7.41	1.46	25.78	22.76
LM4	-	11.37	11.90	13.15	-	38.49	22.81	20.57	6.15	2.25	0.36
LM5	12.64	-	3.70	57.86	17.27	29.90	55.95	8.46	1.08	24.60	30.58
LM6	19.97	-	3.70	29.69	17.27	29.90	55.95	8.46	1.09	24.67	30.58
LM7	9.27	3.34	0.40	-	3.68	-	-	6.82	7.07	7.50	25.50
LM8	1.51	13.00	8.66	2.70	8.09	-	-	16.63	15.85	3.47	31.15
LM9	1.78	4.15	0.77	5.98	10.05	-	-	13.40	3.31	1.05	11.91
LM10	7.57	20.62	1.20	11.46	9.00	-	9.93	9.53	3.39	266.30	13.54

Table 3 *Standard deviation results (σ in cm) after filtering for segment 1 (local map 1) to segment 10 (local map 10). The blanks and high values of standard deviation are shaded in orange, and the improved values are shaded in blue*

Local Map	Angles (°)										
	30	60	90	120	150	180	210	240	270	300	330
LM1	10.89	1.32	1.45	1.20	3.79	0.54	2.06	12.24	0.83	2.14	2.64
LM2	10.45	8.95	4.03	3.88	3.22	1.28	4.09	8.04	0.52	2.41	10.50
LM3	1.13	11.80	5.53	6.81	11.97	10.57	9.70	7.41	1.46	11.51	1.94
LM4	-	11.38	11.90	13.15	-	5.93	2.39	12.08	6.15	2.26	0.36
LM5	12.64	-	3.71	2.54	11.44	4.00	2.78	8.46	1.09	13.50	5.53
LM6	2.81	-	3.71	2.33	8.69	5.50	14.15	8.46	1.09	11.97	17.32
LM7	9.28	3.34	0.40	-	3.68	-	-	6.82	7.08	7.50	13.49
LM8	1.52	13.03	8.67	2.70	8.09	-	-	16.64	15.86	3.48	7.93
LM9	1.78	4.15	0.77	5.98	10.05	-	-	13.40	3.31	1.05	11.91
LM10	7.57	10.03	1.20	11.47	9.01	-	9.93	9.53	3.40	14.82	13.54

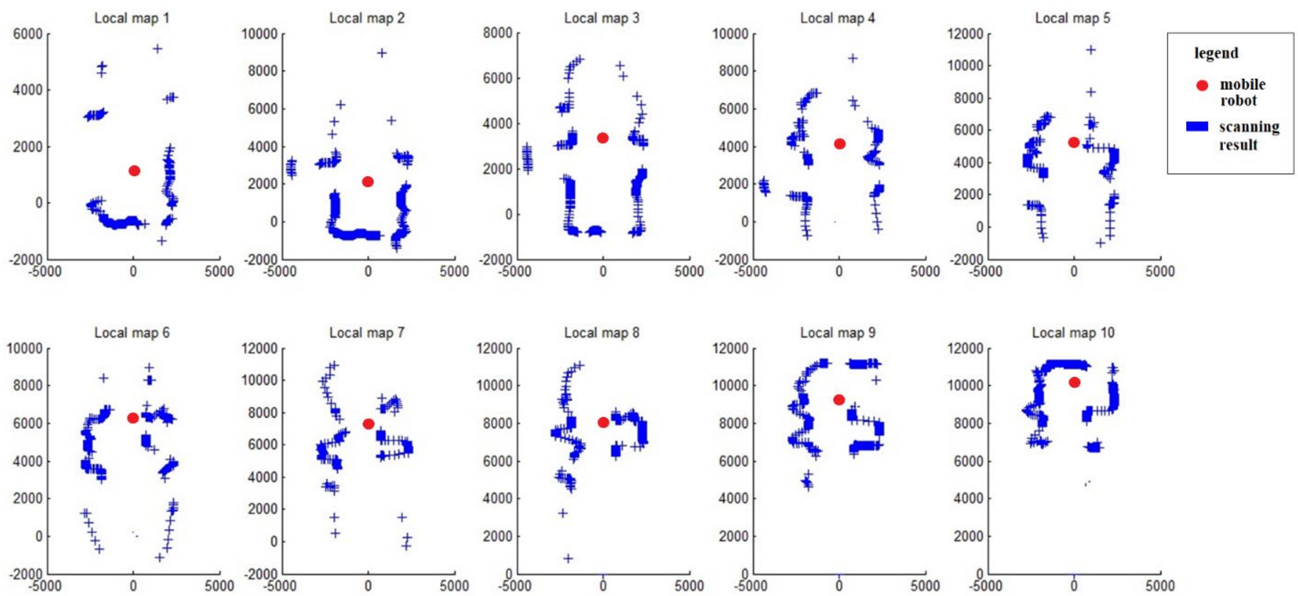
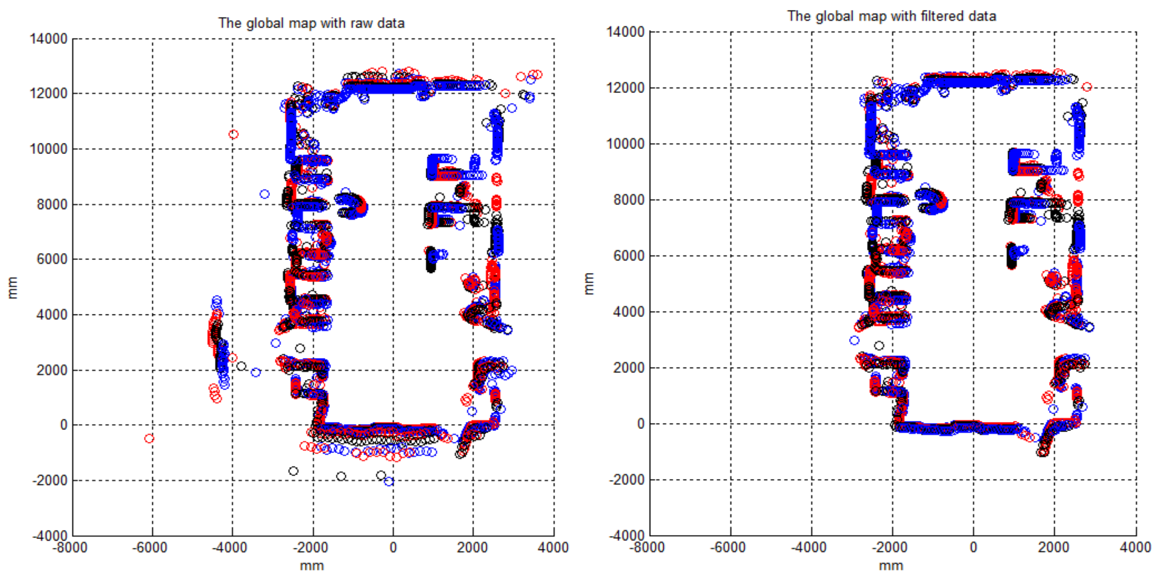


Fig. 10 *2D Plot of local map 1 to local map 10 determination*



(a) **(b)**

Fig. 11 *(a) Reference map with raw data; and (b) Reference map with filtered data*

3.2 Classification Results of KNN Scan Matching Algorithm

The results shown here are to demonstrate the performance of the conventional and the improved algorithm are presented in Table 4 and 5, respectively. It is also expected that the results of reference samples will be higher than non-reference samples, which reference samples are almost similar to the KNN database. These results are performed based on the accuracy of the classification, and the computation is shown in (24).

$$\%\ accuracy = \frac{number\ of\ correct\ sample}{total\ sample} \times 100\% \tag{24}$$

3.2.1 Results of Stage 1 to 3

Stage 1 to 3 is also known as the conventional KNN scan matching localization algorithm. Table 4 shows the local map accuracy to the reference map. It is based on 100 trials for all local maps. For the reference sample classification, the scan matching models of the Euclidean, Mahalanobis, and Manhattan distances obtained the correct classification of 100%, 100%, and 79%, respectively. While, for non-reference sample classification, the accuracy results for Euclidean, Mahalonobis, and Manhanttan are 100%, 100%, and 21%, respectively.

Table 4 *The classification results of the distance parameters based on 100 samples, that are obtained from the reference and non-reference samples*

Distance parameters	Reference sample (% accuracy)	Non-reference sample (% accuracy)
Euclidean	100	100
Mahalanobis	100	100
Manhattan	79	21

For reference sample classification, the high accuracy rates of 100% for both the Euclidean and Mahalanobis distance models are indicative of the effectiveness of these metrics in determining the closeness of scanned data to the reference dataset. In contrast, the Manhattan distance model achieved an accuracy rate of 79% for reference sample classification. This slightly lower accuracy rate implies that the Manhattan distance might not consistently identify the closest data points from the reference dataset. The 79% classification correctness indicates that there might be cases where Manhattan distance does not yield the most accurate results, possibly due to its sensitivity to variations in both x and y coordinates. The local maps that were incorrectly classified by the Manhattan distance metric include local maps 1 (2 incorrect), 2 (1 incorrect), 3 (4 incorrect), 4 (3 incorrect), 5 (2 incorrect), 6 (3 incorrect), 7 (2 incorrect), 8 (2 incorrect), and 9 (2 incorrect).

In the context of non-reference sample classification, the results highlight the superiority of the Euclidean and Mahalanobis distance models. Both achieved 100% accuracy, indicating that these distance metrics are exceptionally effective at classifying samples against a reference dataset even when the data samples did not match the reference set. This finding demonstrates the robustness of Euclidean and Mahalanobis distances in handling data points that may not have direct counterparts in the reference dataset. On the other hand, the Manhattan distance model displayed a 21% accuracy rate for non-reference sample classification. This outcome suggests that the Manhattan distance may not be as well-suited for classifying samples when they deviate from the reference dataset. It appears to struggle in scenarios where the data points follow grid-like or orthogonal patterns, which is a limitation that needs consideration.

To discuss in detail, several factors could contribute to these misclassifications. One possibility is the presence of overlapping or intersecting features, as seen in local maps 3 and 4, and in maps 7 and 8. Additionally, sparse data could be a contributing factor, where certain maps have fewer data points compared to others, limiting the classifier's ability to accurately identify patterns. This issue is particularly apparent in local maps 7 and 8.

3.2.2 Results of Stage 1 to 4

Here, the performance of the improved KNN scan matching localization algorithm is presented and discussed. The results are summarized in Table 5. The distance between the algorithm results and its real coordinates was calculated using equation (25). Various distance ranges or thresholds, ±1cm, ±2cm, ±3cm, ±4cm, ±5cm, and ±10cm, were used to determine the proportion of samples that fell within each range, thereby assessing the algorithm's precision. A total of 100 samples were analyzed for each distance threshold.

$$d = \sqrt{(x_t - x_r)^2 + (y_t - y_r)^2} \tag{25}$$

where (x_t, y_t) are the real coordinates of the mobile robot and (x_r, y_r) are the results of the algorithm.

Table 5 *Accuracy results based on the distance range of ±1, ±2, ±3, ±4, ±5 and ±10. Accuracy 90% or better is highlighted in bold*

Distance parameters (cm)	Reference sample (% accuracy)						Non-reference sample (% accuracy)					
	± 1	± 2	± 3	± 4	± 5	± 10	± 1	± 2	± 3	± 4	± 5	± 10
Euclidean	**98**	**99**	**99**	**99**	**99**	**100**	64	70	80	**93**	**100**	**100**
Mahalanobis	**100**	**100**	**90**	74	41	26	82	75	70	66	54	54
Manhanttan	0	0	2	43	67	79	0	0	0	0	0	21

Int. Journal of Integrated Engineering Vol. 18 No. 3 (2026) p. 311-326

The Euclidean distance consistently achieved high accuracy, ranging from 98% to 100% across all distance ranges, demonstrating its robustness for both small and wide distances. In contrast, the Mahalanobis distance excelled in shorter ranges, achieving 100% accuracy at ±1cm and ±2cm, but its performance significantly declined in broader ranges, dropping to just 41% and 26% at ±5cm and ±10cm, respectively. This drop is likely due to the Mahalanobis equation's reliance on standard deviation, which can result in less precise matching at wider distances. The Manhattan distance, however, performed poorly overall, failing to achieve more than 79% accuracy even at the broadest range.

For non-reference samples, Euclidean distance again demonstrated superior performance, achieving 100% accuracy at ±5cm and ±10cm ranges, though its accuracy declined at narrower ranges, reaching 64% at ±1cm. The Mahalanobis distance followed a reverse trend, improving as the range narrowed, with a maximum accuracy of 82% at ±1cm. However, it never reached 100% accuracy in any range. The Manhattan distance performed worst overall, achieving only 21% accuracy even at the widest range, indicating its limited applicability in the tested scenarios.

The selection of the suitable distance parameter is critical for optimizing the performance of the KNN scan matching localization algorithm. The Euclidean distance has emerged as the most suitable parameter, providing consistent and robust performance across varying conditions. Its straightforward calculation and adaptability to both small and wide ranges make it an ideal choice for mobile robot localization tasks.

In contrast, while the Mahalanobis distance exhibited strong performance in short-range scenarios, its accuracy sharply decreased as the distance range increased. This suggests that Mahalanobis distance may be more appropriate in applications requiring high precision over short distances but is less effective in scenarios demanding broader environmental awareness. The Manhattan distance showed the least potential for practical application in mobile robot localization, particularly in non-reference scenarios, where its accuracy was consistently low. This underperformance highlights the limitations of Manhattan distance in handling the complexity of localization tasks where precision and adaptability are essential.

3.2.3 The Final Performance

Table 6 presents the results of the improved localization algorithm, using the Euclidean distance parameter based on its demonstrated effectiveness in previous tests. The table includes 30 instances of the algorithm's performance, showing the mobile robot's coordinates (x_r, y_r) and the calculated coordinates (x_t, y_t) along with the difference value, *d,* between the real and estimated positions for each axis. Additionally, the local map classification from Stage 3 is indicated.

The results show that most of the difference values are under 3cm, with several instances where the error is zero, indicating a close match between estimated and real positions. When comparing these results to the standard algorithm's performance (as seen in Table 4), the improved algorithm demonstrates significant accuracy gains, confirming that the selection of Euclidean distance plays a crucial role in enhancing mobile robot localization accuracy.

In summary, the improved algorithm, relying on the Euclidean distance parameter, exhibits outstanding accuracy as reflected in local map accuracy, the minimal difference values, even down to zero. It demonstrates a high degree of robustness to variations in the mobile robot's orientation, further solidifying its potential for precise and reliable mobile robot localization across diverse environments.

Table 6 *Results of coordinate determination (Stage 4)*

No.	Real position (cm) LM	Real position (cm) (x_t, y_t)	Result (cm) LM	Result (cm) (x_r, y_r)	d (cm)	No.	Real position (cm) LM	Real position (cm) (x_t, y_t)	Result (cm) LM	Result (cm) (x_r, y_r)	d (cm)
1			1	(1.3,102.0)	2.4	16			6	(0,600.0)	0.0
2	1	(0,100)	1	(6.7,102.2)	2.3	17	6	(0,600)	6	(-0.3,600.1)	0.1
3			1	(9.4,102.7)	2.9	18			6	(0,600.0)	0.0
4			2	(0.7,200.3)	0.3	19			7	(-2,701.8)	1.8
5	2	(0,200)	2	(-0.7,200.1)	0.1	20	7	(0,700)	7	(0.2,700.1)	0.1
6			2	(-2,200.8)	0.9	21			7	(-0.2,700.1)	0.1
7			3	(0.3,300.2)	0.2	22			8	(1,800.2)	0.2
8	3	(0,300)	3	(0,300.1)	0.1	23	8	(0,800)	8	(0,800.0)	0.0
9			3	(0,300.0)	0.0	24			8	(-0.6,800.1)	0.1
10			4	(2.1,400.3)	0.4	25			9	(-0.6,900.1)	1.5
11	4	(0,400)	4	(0.1,400.1)	0.1	26	9	(0,900)	9	(0.1,901.3)	0.1
12			4	(0,400.1)	0.1	27			9	(1.3,901.4)	1.3
13			5	(0,500.0)	0.0	28			10	(0,1000.0)	0.0
14	5	(0,500)	5	(-1.1,501.3)	1.3	29	10	(0, 1000)	10	(0,1000.0)	0.0
15			5	(0.3,500.1)	0.1	30			10	(0.2,1000.2)	0.2

*LM = local map

4. Conclusion

Finally, this study considerably improved mobile robot localization by developing and applying the KNN scan matching algorithm. Unlike classic KNN scan matching, which simply provides an area estimate, the proposed approach exactly calculates the mobile robot's coordinates in the environment, significantly improving localization accuracy. Furthermore, the work demonstrates the efficacy of using a single laser range finder for mobile robot localization. Among several distance parameters, the Euclidean distance consistently exceeds the Mahalanobis and Manhattan distances in KNN classification. The revised algorithm is highly accurate, reaching 93% accuracy within a 4 cm range across 100 test samples and consistently determining coordinates within 3 cm for 30 samples.

As a recommendation for future work, the algorithm will be tested in dynamic environments that include both static and moving objects. The approach can be implemented in a manner that the algorithm updates the reference map in predetermined intervals. This will allow it to perform in an environment in which the objects are dynamic. This extension would provide valuable insights into the algorithm's robustness and effectiveness in real-world scenarios. Overall, this study paves the way for further advancements in mobile robot localization algorithms and their application in various dynamic environments.

Acknowledgement

We would like to express our appreciation for the support provided by all parties directly or indirectly involved in this research. We also thank our colleagues and collaborators for their support and assistance throughout the research process.

Conflict of Interest

Authors declare that there is no conflict of interests regarding the publication of the paper.

Author Contribution

The authors confirm contribution to the paper as follows: ***study conception and design:*** *Markom, M.A., Adom, A.H., Mohd Muslim Tan, E.S.;* ***data collection:*** *Markom, M.A., Markom, A.M;* ***analysis and interpretation of results:*** *Markom, M.A., Zakaria, A., Adom, A.H.;* ***draft manuscript preparation:*** *Markom, M.A., Adom, A.H., Markom, A.M. All authors reviewed the results and approved the final version of the manuscript.*

References

[1] Husbands, P. (2021). Robots: What Everyone Needs to Know®. Oxford University Press.

[2] Nikitha, M. A., Sai Swetha, B. S., Mantripragada, K. H., & Jayapandian, N. (2022, February). The future warfare with multidomain applications of artificial intelligence: research perspective. In Proceedings of Second International Conference on Sustainable Expert Systems: ICSES 2021 (pp. 329-341). Singapore: Springer Nature Singapore. DOI: 10.1007/978-981-16-7657-4_28.

[3] Dehant, V., Blanc, M., Mackwell, S., Soderlund, K.M., Beck, P., Bunce, E.J., Charnoz, S., Foing, B.H., Filice, V., Fletcher, L.N., Forget, F., Griton, L., Hammel, H.B., Honing, D., Imamura, T., Jackman, C.M., Kaspi, Y., Korablev, O.I., Leconte, J., Lellouch, E., Marty, B., Mangold, N., Michel, P., Morbidelli, A., Mousis, O., Prieto-Ballesteros, O., Spohn, T., Schmidt, J., Sterken, V.J., Tosi, N., Vandaele, A.C., Vernazza, P., Vazan, A., & Westall, F. (2022). Planetary Exploration Horizon 2061 Report, Chapter 3: From science questions to Solar System exploration. DOI: 10.48550/arXiv.2211.04474

[4] Aguzzi, Jacopo, Flogel, Sascha, Marini, Simone et al (2022) Developing technological synergies between deep-sea and space research. Elementa: Science of the Anthropocene, 10 (1). 00064. ISSN 2325-1026

[5] Escobar-Naranjo, J., Caiza, G., Ayala, P., Jordan, E., Garcia, C. A., & Garcia, M. V. (2023). Autonomous Navigation of Robots: Optimization with DQN. Applied Sciences, 13(12), 7202. DOI: 10.3390/ app13127202

[6] Jiang, Z., Zhu, J., Lin, Z., Li, Z., & Guo, R. (2020). 3D mapping of outdoor environments by scan matching and motion averaging. Neurocomputing, 372, 17-32. DOI: 10.1016/j.neucom.2019.09.022

[7] Cheein, F. A., Torres-Torriti, M., & Rosell-Polo, J. R. (2019). Usability analysis of scan matching techniques for localization of field machinery in avocado groves. Computers and Electronics in Agriculture, 162, 941-950. DOI: 10.1016/j.compag.2019.05.024

[8] Zhu, J., & Xu, L. (2019, November). Design and Implementation of ROS-Based Autonomous Mobile Robot Positioning and Navigation System. In 2019 18th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES) (pp. 214-217). IEEE. DOI: 10.1109/DCABES48411 .2019.00060

[9] Ilias, B., Shukor, S. A. A., Adom, A. H., Ibrahim, M. F., & Yaacob, S. (2016, May). A novel indoor mobile robot mapping with USB-16 ultrasonic sensor bank and NWA optimization algorithm. In 2016 IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE) (pp. 189-194). IEEE. DOI: 10.1109/ISCAIE.2016.7575061

[10] Wang, H., Lin, Y., Wang, Z., Yao, Y., Zhang, Y., & Wu, L. (2017). Validation of a low-cost 2D laser scanner in development of a more-affordable mobile terrestrial proximal sensing system for 3D plant structure phenotyping in indoor environment. Computers and Electronics in Agriculture, 140, 180-189. DOI: 10.1016/j.compag.2017.06.002

[11] Filotheou, A. (2022). Correspondenceless scan-to-map-scan matching of homoriented 2D scans for mobile robot localization. Robotics and Autonomous Systems, 149, 103957. DOI: 10.1016/j.robot.2021.103957

[12] Konecny, J., Kromer, P., Prauzek, M., & Musilek, P. (2019). Scan matching by cross-correlation and differential evolution. Electronics, 8(8), 856. DOI: 10.3390/electronics8080856

[13] Li, X., Du, S., Li, G., & Li, H. (2019). Integrate point-cloud segmentation with 3D LiDAR scan-matching for mobile robot localization and mapping. Sensors, 20(1), 237. DOI: 10.3390/ s20010237

[14] Filotheou, A., Tsardoulias, E., Dimitriou, A., Symeonidis, A., & Petrou, L. (2020). Pose selection and feedback methods in tandem combinations of particle filters with scan-matching for 2D mobile robot localization. Journal of Intelligent & Robotic Systems, 100, 925-944. DOI: 10.1007/s10846-020-01253-6

[15] Jiang, Z., Zhu, J., Lin, Z., Li, Z., & Guo, R. (2020). 3D mapping of outdoor environments by scan matching and motion averaging. Neurocomputing, 372, 17-32. DOI: 10.1016/j.neucom.2019.09.022

[16] Niewola, A., & Podsędkowski, L. (2020). PSD–probabilistic algorithm for mobile robot 6D localization without natural and artificial landmarks based on 2.5 D map and a new type of laser scanner in GPS-denied scenarios. Mechatronics, 65, 102308. DOI: 10.1016/j.mechatronics.2019.102308

[17] Chen, P., Shi, W., Fan, W., Xiang, H., & Bao, S. (2021). RectMatch: A novel scan matching method using the rectangle-flattening representation for mobile LiDAR systems. ISPRS Journal of Photogrammetry and Remote Sensing, 180, 191-208. DOI: 10.1016/j.isprsjprs.2021.08.011

[18] Abanay, A., Masmoudi, L., & El Ansari, M. (2022). A calibration method of 2D LIDAR-Visual sensors embedded on an agricultural robot. Optik, 249, 168254. DOI: 10.1016/ j.ijleo.2021.168254

[19] Koide, K., Miura, J., & Menegatti, E. (2020). Monocular person tracking and identification with on-line deep feature selection for person following robots. Robotics and Autonomous Systems, 124, 103348. DOI: 10.1016/j.robot. 2019.103348

[20] Atiyah, H. A., & Hassan, M. Y. (2021). Outdoor localization in mobile robot with 3D LiDAR based on principal component analysis and K-Nearest neighbors algorithm. Engineering and Technology Journal, 39(06), 965-976. DOI: 10.30684/etj.v39i6.2032

[21] Keung, K. L., Lee, C. K., & Ji, P. (2021). Data-driven order correlation pattern and storage location assignment in robotic mobile fulfillment and process automation system. Advanced Engineering Informatics, 50, 101369. DOI: 10.1016/j.aei.2021.101369

[22] Wong, B., Marquette, W., Bykov, N., Paine, T. M., & Banerjee, A. G. (2023). Human-assisted robotic detection of foreign object debris inside confined spaces of marine vessels using probabilistic mapping. Robotics and Autonomous Systems, 161, 104349. DOI: 10.1016/j.robot.2022.104349

[23] Gómez, J., Aycard, O., & Baber, J. (2023). Efficient Detection and Tracking of Human Using 3D LiDAR Sensor. Sensors, 23(10), 4720. DOI: 10.3390/s23104720

[24] Tan, Y., Li, S., Liu, H., Chen, P., & Zhou, Z. (2021). Automatic inspection data collection of building surface based on BIM and UAV. Automation in Construction, 131, 103881. DOI: 10.1016/j.autcon.2021.103881

[25] Zhu, K., & Zhang, T. (2021). Deep reinforcement learning based mobile robot navigation: A review. Tsinghua Science and Technology, 26(5), 674-691. DOI: 10.26599/TST.2021.9010012

[26] Rezende, A. M., Júnior, G. P., Fernandes, R., Miranda, V. R., Azpúrua, H., Pessin, G., & Freitas, G. M. (2020, August). Indoor localization and navigation control strategies for a mobile robot designed to inspect confined environments. In 2020 IEEE 16th International Conference on Automation Science and Engineering (CASE) (pp. 1427-1433). IEEE. DOI: 10.1109/CASE48305.2020.9217005

[27] Cebollada, S., Payá, L., Flores, M., Peidró, A., & Reinoso, O. (2021). A state-of-the-art review on mobile robotics tasks using artificial intelligence and visual data. Expert Systems with Applications, 167, 114195. DOI: 10.1016/j.eswa.2020.114195

[28] Kasap, M., Yılmaz, M., Çinar, E., & Yazıcı, A. (2023). Unsupervised dissimilarity-based fault detection method for autonomous mobile robots. Autonomous Robots, 47(8), 1503-1518. DOI: 10.1007/s10514-023-10144-2