Jurnal Teknologi

Full Paper

AUTONOMOUS EXPLORATION OF UNKNOWN ENVIRONMENTS USING ENTRANCE DETECTION, CLASSIFICATION, AND MULTI-TARGET ENTRANCE ORGANIZATION

Muqri Muazam Zainal[a], Kamarulzaman Kamarudin[a], A Hamid Adom[a], Wan Rahiman[b], Nasr Abdalmanan[a], M Aizat Abu Bakar[a], M Rizal Manan[a]

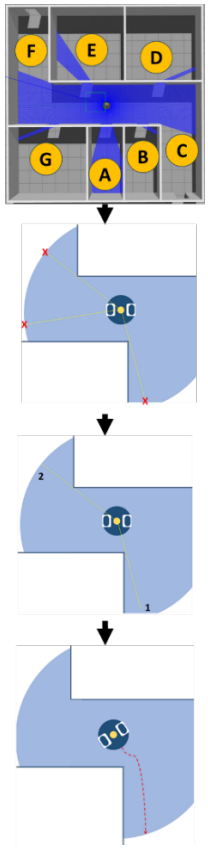
[a]Faculty of Electrical Engineering and Technology (Mechatronic Engineering Department), Universiti Malaysia Perlis, Perlis, Malaysia

[b]School of Electrical and Electronic Engineering, Universiti Sains Malaysia.

Article history
Received
11th February 2025
Received in revised form
28th Janaury 2026
Accepted
27th February 2026
Published
1st November 2026

*Corresponding author
kamarulzaman@unimap.edu.my

Graphical abstract



Abstract

Exploration in unknown environments remains a significant challenge in robotics and artificial intelligence, particularly in navigating dynamic (always changing) and complex terrains such as indoor areas with multiple rooms and obstacles. Traditional static target algorithms struggle with dynamic changes and lack the capability to verify detected entrances or junctions as actual free spaces. Additionally, these methods process one target at a time, increasing the risk of missing critical exploration areas. The need for advanced algorithms that adapt to dynamic entrance, evaluate detected entrance, sort multiple entrances, and efficiently explore based on importance is essential for advancing both research and practical applications. To address these challenges, this paper introduces the Multiple Target Evaluation Node (M-TEN) framework, inspired by Frontier Exploration. Using LiDAR data, M-TEN detects possible entrances, verifies true entrances with 99% accuracy, and optimally generates a sequence for multiple entrances. Performance was tested in a 10m x 10m simulation environment with two map configurations. The results demonstrate that M-TEN enables mobile robots to explore unknown environments efficiently, achieving 94.5% mapping coverage within 3.8 minutes compared to manually controlled and basic frontier exploration approach. This framework enhances exploration accuracy, reduces time, and ensures reliable target evaluation and sorting.

Keywords: Unknown environment, exploration, multiple entrance, evaluate entrance, optimum sequence.

Abstrak

Penerokaan kawasan yang tidak diketahui masih menjadi satu cabaran yang ketara dalam industri robotik dan kecerdasan buatan, khususnya penerokaan dalam kawasan yang mempunyai pintu masuk yang sentiasa berubah seperti dalam bangunan yang mempunyai bilik yang banyak. Pendekatan yang sedia ada hanya mampu proses satu destinasi dalam satu-satu masa dan tidak mampu untuk beradaptasi dalam situasi yang tidak diketahui dan berubah-ubah. Kekurangan ini perlu diatasi dengan rangka kerja yang mampu proses destinasi yang banyak, memberi justifikasi terhadap kewujudan destinasi tersebut, seterusnya menyusun lokasi destinasi tersebut dengan turutan yang optimum. Bagi mengatasi kekurangan ini, kajian ini memperkenalkan rangka kerja Penilaian Pelbagai Nod Sasaran (Pintu Masuk) (M-TEN) yang menggunakan data dari penderia LiDAR untuk mengenal pasti pintu masuk, dan memberi justifikasi terhadap kewujudan pintu masuk tersebut, seterusnya menyusun susunan pintu masuk yang optimum. Kajian dibuat menggunakan

DOI: https://doi.org/10.11113/jurnalteknologi.v89.24309

simulasi dalam peta bersaiz 10m x 10m dengan dua jenis susunan bilik. Hasil kajian menunjukkan rangka kerja M-TEN mampu memastikan robot untuk meneroka kawasan tersebut dengan effisien dan mencapai 94.5% ketepatan liputan peta, dengan masa 3.8 minit yang mana lebih pantas berbanding pengendalian manual. Hasil ini menunjukkan rangka kerja ini mampu untuk mengenal pasti pelbagai pintu masuk, meningkatkan ketepatan penerokaan, menyingkatkan masa, justifikasi kewujudan pintu masuk yang boleh dipercayai, dan susunan destinasi yang lebih optimum.

Kata kunci: Kawasan tidak diketahui, penerokaan, pelbagai pintu masuk, justifikasi pintu masuk, penyusunan optimum

© 2026 Penerbit UTM Press. All rights reserved

1.0 INTRODUCTION

Exploration in unknown environments is a critical challenge in the fields of robotics, autonomous systems, and artificial intelligence, emphasizing the need for a framework capable of self-navigation and decision-making [1]. In situation such as disaster response and exploration in dangerous gas leaking accident area which led to human intervention often seen impractical and dangerous that may affect the human health condition and complication [2]. The application of an autonomous system in these impractical environments for humans addresses efficient, reliable, and adaptive exploration in an unknown environment without human intervention [3]. Autonomous systems can navigate, map, and make real-time decisions in such environments, significantly enhancing efficiency and safety [4]. However, the traditional approaches for integration of autonomous capability often rely on static target algorithms, which are not well-suited to dynamically changing environments [5].

This limitation poses significant challenges, particularly when an autonomous agent must navigate through complex, unknown terrains such as indoor areas with multiple rooms, corridors, and obstacles [6]. Frontier Exploration is one of the traditional methods, capable of detecting the possible entrance or junction in a static environment, but unable to evaluate whether these detected entrances are actual entrances or false entrances before exploring it [7]. Additionally, current advanced methods can only handle a single target at a time, making this existing method ineffective at processing multi-target simultaneously [8], [9]. This limitation increases the risk of the agent missing areas that need exploration. To address the need for multiple entrance adaptability in static environments, both in research and real-world applications, the advanced framework should be capable of detecting and evaluating entrances as true or false, prioritizing multiple entrances, and efficiently exploring each entrance based on its significance. [10].

In order to enhance the autonomous capability to fit unknown area exploration, several components need to be considered. Firstly, there should be a sensor and sensor data processing algorithm capable to detect and process multiple entrances. In the context of mobile robot exploration, entrances are defined as potential openings or free spaces that the mobile robot can reach within the observed area, distinct from obstacles or walls. Secondly, the mobile robot should be able to execute an effective and reliable approach to evaluate the observed entrances based on the sensor data and differentiate it as false-entrance and true-entrance. Third component is the need of entrance coordinate sequencer to ensure the mobile robot explore the area with optimum entrance sequence.

Currently there are at least two main types of sensors capable to gathering the surrounding environment information which are LiDAR sensor and Camera-Based sensor. LiDAR sensors offer superior robustness in challenging environments by providing precise 2D spatial information regardless of lighting conditions [11], [12]. Camera-based methods, on the other hand, such as RGB and RGB-D cameras, rely on visual cues to identify structural features like entrances [13], [14], [15], [16]. LiDAR sensors offer precise 2D mapping and depth perception for unknown area exploration, while camera-based sensors provide visual data but struggle with depth accuracy and low-light conditions, making LiDAR more reliable for exploration in unknown environments.

However, the raw data captured by LiDAR sensors is complex and requires processing algorithms to interpret and distinguish specific environmental features, such as entrances, obstacles, or walls [17]. To address this challenge, the existing method processes LiDAR data by implementing the combination of geometric heuristics, and facilitate the detection of potential openings or entrances by analysing dimensions, shape, and continuity of clusters [18]. Due to the lack of an evaluation component, existing methods are often inaccurate in determining entrance locations and may be unreliable in assessing true entrances [19]. Despite these challenges, the existing method can be improved by combining with classification and sorting algorithm.

In recent years, there are several types of classification algorithm introduced including supervised and unsupervised classifier. The unsupervised classifier focusing on clustering the data based on the centroid learning from testing data and decide the class based on the distance between each data toward all centroids [20]. One of the unsupervised classifiers is K-

DOI: https://doi.org/10.11113/jurnalteknologi.v89.24309

Means algorithm [21]. Other than that, supervised classifier focuses on training data-based clustering, relying on a large labelled dataset to ensure accurate classification based on predefined outcomes [22]. The supervised learning includes neural network classification and K-nearest neighbour (K-NN) [23], [24].

In comparison, supervised learning provides a slight advantage but requires a pre-labelled dataset, whereas unsupervised learning offers unbiased results, though its performance is limited with datasets that show only minor differences. However, there is no existing research that compares these two types of methods specifically on the suitability in entrance detection application [25]. These two types of classifier flexibility and reliability can be significantly enhanced by tuning it to focus specifically on features such as the different distance between angles from LiDAR data and spatial distance. The suitability of each classifier will be studied in this paper to be implemented in the proposed framework [26].

Next, the sorting algorithm determines the optimal order for exploring entrances within an unexplored area. Heuristic-based methods use weighted scoring functions to prioritize entrances by assigning numerical importance to factors like proximity, safety, and accessibility [27]. Another method using Multi-criteria decision-making (MCDM) frameworks, such as the Analytic Hierarchy Process (AHP), provides a more systematic approach by comparing specific criteria and calculating weighted scores, incorporating both qualitative and quantitative factors [28]. Although effective for unknown area exploration, these methods are constrained by predefined weights, reducing its adaptability in unknown environment exploration. Additionally, AHP method can be biased due to non-foundation judgement and may lead to false decisions in complex scenarios.

Based on the literature, numerous studies have introduced a method for entrance detection, classifier algorithm and sorting algorithm. Each method has its own advantages in entrance detection application and the disadvantages and suitability can be improvised to fit entrance detection application. Building on these advancements, this study aims to propose an improvised framework that leverages LiDAR data and advanced classification algorithms to detect, evaluate, and prioritize entrances, addressing redundant exploration and improving mission efficiency.

This paper contributes to the ongoing development of autonomous framework capable of operating in unpredictable and unknown environments. The findings of this study could have significant implications for applications in search and rescue missions, environmental monitoring, and autonomous navigation in both indoor and outdoor settings. The contribution of this study includes the following three point:

1. The algorithm to detect possible entrance based on data from LiDAR sensor.
2. An optimized entrance evaluation algorithm to classify true or false entrance and reduce exploring time.
3. Establish an entrance sequence algorithm to optimize sequence exploration in unknown area.

2.0 METHODOLOGY

The method proposed in this paper composed of three distinct components. The first component focuses on identifying potential entrances within an unknown environment to be explored by the mobile robot. The second component evaluates and compares three different classification algorithms to enhance decision-making capabilities, ensuring the reliable detection and evaluation of entrances or junctions in the environment. Finally, the third component introduces a multi-target organizer, which systematically arranges and prioritizes the list of identified entrance based on its importance and the complexity of the paths required to reach each entrance.

Building on the proposed methodology, each component plays a critical role in addressing the challenges of exploration and entrance detection in unknown environments. The target acquisition method lays the foundation by accurately identifying potential points of interest, minimizing the likelihood of missing crucial entrances. The classification algorithms add an essential layer of validation, reducing false positives and ensuring that resources are not wasted on exploring non-viable targets. Meanwhile, the multi-target organizer optimizes the exploration strategy by enabling efficient navigation through the environment, ensuring that high-priority targets are addressed first while accounting for path complexity. Together, these components create a cohesive system that balances accuracy, efficiency, and adaptability in exploration tasks. The block diagram for the whole framework called Multi-Target Entrance Evaluation Node (M – TEN) shown in Figure 1.

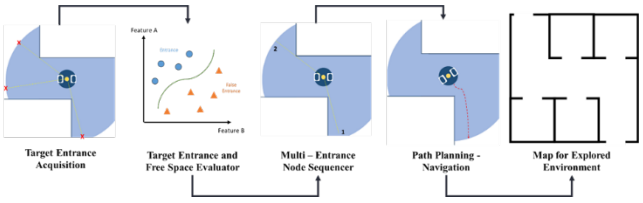


Figure 1 M-TEN Framework

2.1 First Component: Target Entrance Acquisition Algorithm

The first component, the target entrance acquisition method, serves as the cornerstone of the system by leveraging LiDAR data to detect potential entrances in an unknown environment. This process involves

DOI: https://doi.org/10.11113/jurnalteknologi.v89.24309

analysing point cloud data to identify distinct features such as gaps, openings, or other spatial characteristics that indicate an entrance or junction. This paper proposes a method by utilizing distance and angle data from LiDAR to identify the sudden change in distance between angle to define as a possible entry, free space, or junction inspired from Frontier Exploration algorithm [29].

This method utilizes a Sliding Window Filter to compare the distances between the current and previous angles, detecting any sudden changes. The window, with a size of 1-1 will be move to each angle start from 0° until 359°. All the different distance readings from the window will be recorded and plotted in a map representing each angle. The specific equation for this method is shown in Equation 1.

$$L_{new}(\theta) = \sum_{i=0}^{\theta=359} [L\,(\theta+1) - L\,(\theta)] \quad (1)$$

$$L = Distance\ from\ LiDaR$$
$$\theta = Angle\ in\ degree$$

The coordinate in 2D cartesian plane (x and y) for each angle with significant changes in distance will be extracted and defined as target coordinate for the mobile robot to explore during exploration task. By integrating geometric heuristics, the method enhances the accuracy of entrance identification, setting a robust foundation for subsequent evaluation. The detailed illustration of the first component is shown in Figure 2.

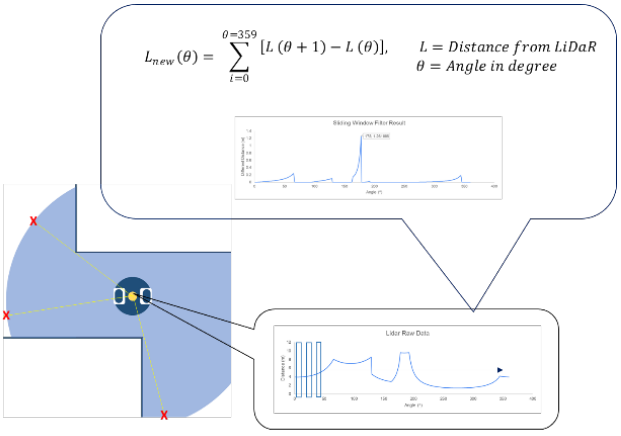


Figure 2 Target Entrance Acquisition Algorithm Illustration

2.2 Second Component: Target Entrance and Free-Space Evaluator Algorithm

This second step focuses on classification algorithms to validate the existence of detected entrances. In this component, this paper compares three type of classifier algorithm which include K-NN, K-Means and Neural Network. Each classifier will be fed the same data with two features: Difference in distance between current angle with previous angle and Spatial distance from each detected entrance to another detected entrance. The principle behind the selection of feature is:

1. The big sudden change between current angle with previous angle indicate that particular angle may serve as possible entrance.
2. The closer the spatial distance between each detected possible entrance to another detected entrance indicate that there may exist false-entrance or redundancy of detected entrance.

The data for each selected feature was recorded through the simulation from Target Entrance Acquisition Algorithm reading. The simulation setup to collect the trained data is by using two different maps shown in Figure 3, with 14 randomize position of the mobile robot to collect various data pattern.

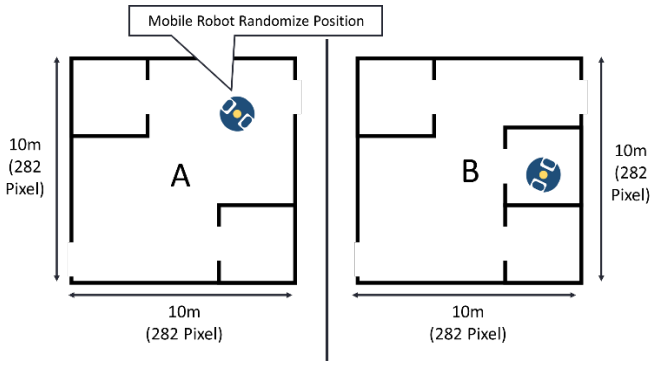


Figure 3 Environment setup to collect training data

The number of data specified for training purpose for all three-classifier algorithm is 5040 data for each feature. Then, another 500 data recorded from Target Entrance Algorithm will be utilized to test the accuracy of each algorithm in detecting possible or free space after training phase. The optimum classifier with high accuracy classifier to evaluate the existence of possible entrances or free – space. Each classifier working principle use in this application shown in Figure 4.

DOI: https://doi.org/10.11113/jurnalteknologi.v89.24309

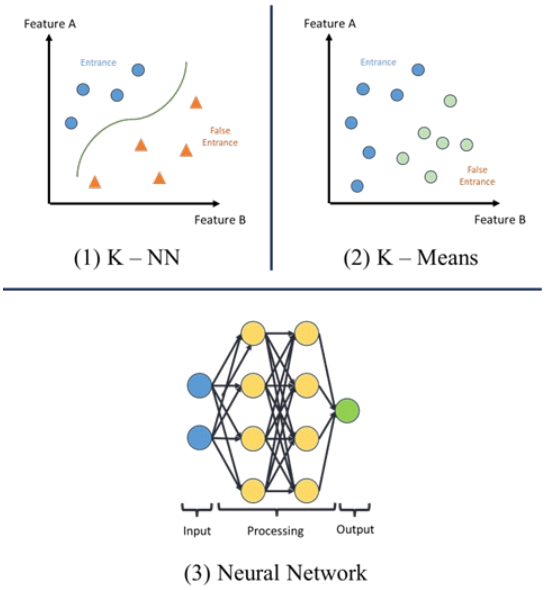


Figure 4 Three Classifier compared in this paper

By comparing multiple classification approaches including K – NN, K – Means, Neural Network is to ensures that the algorithm in this second component achieves optimal decision-making performance. Each algorithm is evaluated on its ability to handle noise, adapt to varied environmental conditions, and reduce false positives, contributing to a higher degree of reliability in real-world applications.

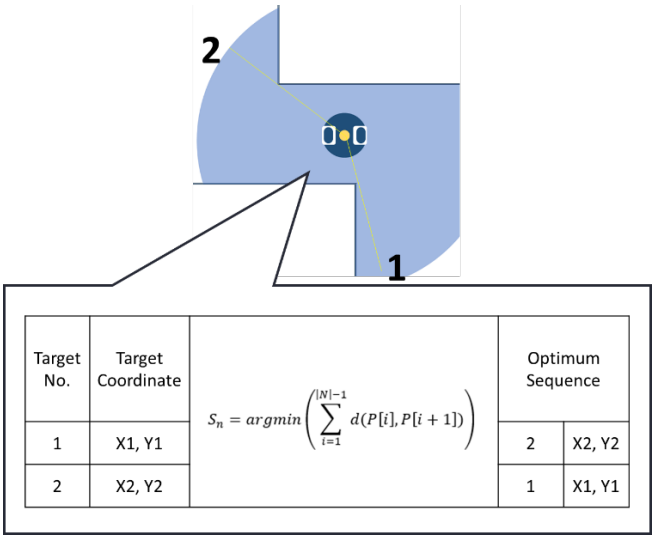
2.3 Third Component: Multi – Entrance Node Sequencer Algorithm

The third component, the multi – entrance node algorithm, complements the previous component by introducing a systematic approach to prioritize and sort all evaluated target entrance. After evaluation process from previous component, there is high possibility that there are multiple true entrances detected. In this case, each entrance may have different complexity and may affect the exploration time if the mobile robot goes to each entrance randomly [30].

The Multi – Entrance Node sequencer starts with identify the distance from current location of mobile robot to each detected entrance. Then, with the velocity data from the mobile robot, this algorithm will compute the best possible sequence based on the speed of the mobile robot and distance between mobile robot and each entrance. The resulting sequence is then fed to the mobile robot for exploration process. The list of sequence generated from the multiple entrances is computed based on Equation 2. Figure 5 shows full illustration of Multi – Entrance Node Sequencer algorithm.

$$S_n = argmin\left(\sum_{i=1}^{|N|-1} d(P[i], P[i+1])\right) \quad (2)$$

$S = Sequence\ of\ Node$
$argmin = Argument\ to\ select\ minimum\ value\ of\ distance$
$|N| = Total\ number\ of\ nodes$
$i = Initial\ number\ of\ node\ (First\ Entrance)$
$d(P[i], P[i+1]) = Distance\ between\ Current\ Node\ to\ Next\ Nodes$



Target No.	Target Coordinate	$S_n = argmin\left(\sum_{i=1}^{\|N\|-1} d(P[i], P[i+1])\right)$	Optimum Sequence	
1	X1, Y1		2	X2, Y2
2	X2, Y2		1	X1, Y1

Figure 5 Multi – Entrance Node Sequencer Algorithm Illustration

Based on Equation 2 and Figure 5, the algorithm employs a systematic approach to path planning and complexity assessment. Path complexity, measured by the number of obstacles encountered and total traversal time from start to target point, is evaluated for each potential route. The algorithm begins with the nearest entrance node and calculates the optimal sequence by progressing from the current nearest node to the next closest node. This process continues iteratively until a complete sequence of nodes is generated, covering all areas. The algorithm employs a greedy approach to determine the shortest path using the Euclidean Distance equation while considering the path complexity to ensure efficient navigation around obstacles. The output is a sequential list of nodes representing all evaluated entrances from component two, which is then provided to the mobile robot to follow each node in order until all have been visited

This algorithm act as complement for the whole framework propose in this paper and ensure that exploration is both efficient and strategic, taking into account factors such as the proximity of targets, the complexity of paths and the importance of each target. Together, these components create a unified framework for efficient and accurate entrance detection in unknown environments. The whole framework used in this study by combining each component is called Multi-Target Entrance Evaluation Node (M – TEN). The mobile robot will be simulated using

DOI: https://doi.org/10.11113/jurnalteknologi.v89.24309

Gazebo in ROS 2 platform. Nav2 tool used for path planning and SLAM to evaluate the quality of generated map compared to the floor plan. The setup map will be tested on 2 different maps as shown in Figure 6.

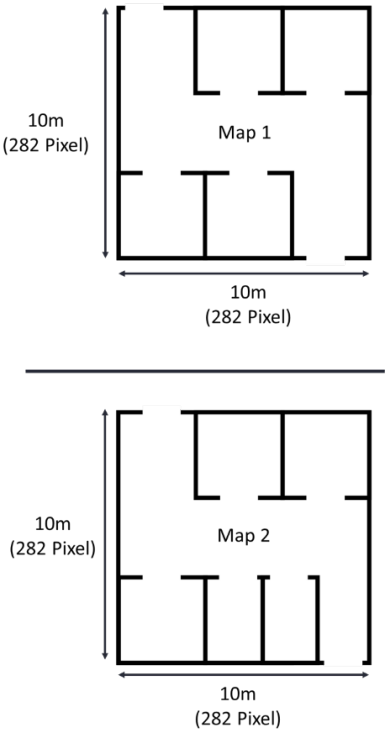


Figure 6 Setup Map for training and testing

3.0 RESULTS AND DISCUSSION

3.1 Target Entrance Acquisition Result

The target entrance acquisition algorithm was tested in two environments with different room position setup. In Map 1, the room was set to four room and expected entrance or free space detection is six. Different with Map 2 which the room was set to five rooms and referring to position of mobile robot at the middle of map, the expected entrance or free space is 7 entrances. The expected entrance detected by the algorithm was determine by analyse the original setup map and consider the positioning of room and its entrance. The result of target entrance acquisition algorithm shown in Figure 6.

The result presented the positioning of mobile robot as agent at the middle of map in both maps, the expected entrance, the raw LIDAR data and filtering result showing the significant changes in different distance between each angle from LIDAR data.

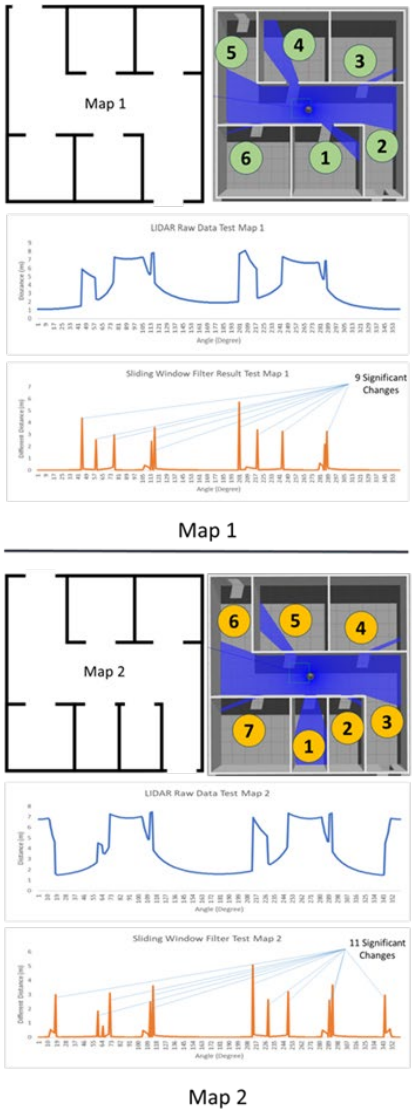


Figure 7 Result for Target Entrance Acquisition in Map 1 (Top) and Result for Target Entrance Acquisition in Map 2 (Bottom)

According to Figure 7, the results of the target acquisition algorithm demonstrate that the sliding window filter effectively identifies significant changes in LIDAR readings. The high-amplitude readings produced by the filtering process indicate sudden variations in distance between angles. This highlights the algorithm's capability to detect significant changes in raw LIDAR data, with each detected change representing a potential entrance or free space for the mobile robot to explore. The accuracy of detecting entrances based on these significant distance changes can be compared and evaluated in Target Entrance evaluator result. The number of detected entrance and true entrance shown Table 1.

DOI: https://doi.org/10.11113/jurnalteknologi.v89.24309

Table 1 Total of vehicles for each entrance

Setup Map	Detected Entrance	True Entrance
Map 1	9	6
Map 2	11	7

The difference between detected entrance and true entrance is due to the redundancy of detected entrance. As shown in Figure 7, Map 1 (on the top side) illustrates the LIDAR detecting two sudden changes in distance at entrance number 1, corresponding to two distinct angles. Despite the presence of two significant changes, these readings collectively indicate the presence of a single entrance.

This highlights the LIDAR sensitivity in detecting abrupt transitions, which can occur when scanning structural boundaries or entry points within the environment. The redundancy in detecting entrances may increase the time required for the mobile robot to explore the same room from different locations. To mitigate this redundancy, each detected location must be verified by the second component which are target entrance and free space evaluator.

3.2 Target Entrance and Free Space Evaluator Algorithm result

The results of the target entrance acquisition algorithm, as presented in the previous subsection, clearly highlight the need for further filtering of the detected entrances through a classifier algorithm. This additional step is essential to ensure the reliability of the identified entrances. To evaluate and refine the list of entrances, three different classifier algorithms were tested in this study. These classifiers were specifically assessed for its effectiveness in processing raw LIDAR data and accurately distinguishing between false entrances and true entrances. The data generated by the target entrance acquisition algorithm served as input for testing each classifier, enabling a comprehensive evaluation of each algorithm suitability for this task. The feature setup for the classification data is Different Distance between angle and Spatial Distance between reading for each angle of detected coordinate.

The Target Entrance and Free Space Evaluator algorithm test in this paper is three types with one unsupervised learning which are K-Means classifier, and two supervised learning classifier which are K-NN and Neural Network algorithm. Each classifier used same 5040 dataset, trained and then save in one trained model for testing purpose. The result for training is shown in Figure 8.

Figure 8 illustrates the performance of three classifiers including K-NN, Neural Network, and K-Means trained on the same dataset of 5040 samples. Despite using identical training data, the classification results varied due to differences in each algorithm underlying algorithms. Among the three, the K-NN classifier demonstrated the most promising performance, effectively distinguishing true entrances from false entrances (obstacles or redundant detections).

The K-NN classification (centre graph) utilized two neighbours to classify data points, resulting in a balanced and clear distinction between yellow (true entrances) and purple (false entrances). In contrast, the K-Means results (left graph) were skewed towards the yellow class, indicating an overclassification of data points as true entrances, with fewer points identified as false entrances. Lastly, the Neural Network results (right graph) showed some classification ability but leaned heavily toward the yellow class, classifying most data points as true entrances with only a small portion identified as false entrances (purple). This suggests a bias in the Neural Network's classification approach compared to K-NN's more balanced results. In this training result, each classifier reflects its working principle and has been tested on exploration data on Map 1 and Map 2. The testing result was shown in Figure 9 and Figure 10.

Figure 9 presents the testing results based on exploration data from Map 1. The dataset, collected from LIDAR's initial scan for first interpretation, comprises 360 data points. Each classifier including K-Means, K-NN, and Neural Network was tested to evaluate each ability to distinguish true entrances from false entrances (obstacles or redundant detections).

According to Figure 9, K-Means results (left graph) identified four true entrances (red areas), with the remaining data classified as false entrances. This outcome aligns with the training data, where K-Means tended to classify the true entrance class slightly toward the right side of the graph. The classifier appears to set a threshold between true and false entrances at an absolute difference of approximately 2 to 3.

The K-NN results (middle graph) showed a more balanced classification, successfully detecting six true entrances (red areas), with the remainder classified as false entrances. This performance is consistent with the training results, where the K-NN model was designed to classify evenly across classes without bias. As a result, K-NN produced stable and reliable classifications of true and false entrances. Finally, the Neural Network results (right graph) mirrored those of K-NN, accurately identifying six true entrances while maintaining even classification between classes.

Based on the Map 1 exploration data, the testing results indicate that K-NN and Neural Network classifiers are more effective at identifying true entrances, with both detecting six true entrances as observed in Map 1. In contrast, K-Means was only able to identify four true entrances, falling short of the actual count of six. Each classifier then testing for Map 2 as result in Figure 10.

Figure 10 illustrates the performance of the three classifiers on Map 2 exploration data. The K-Means algorithm (left graph) detected up to four true entrances, showing similar performance to its results on Map 1. This outcome aligns with the training model, where K-Means tended to classify more heavily toward the right side of the graph.

DOI: https://doi.org/10.11113/jurnalteknologi.v89.24309

In contrast, the K-NN classifier (middle graph) demonstrated the ability to classify data evenly, identifying seven true entrances, with the remainder classified as false entrances. The Neural Network classifier (right graph) detected six true entrances. While K-NN and Neural Network performed similarly on Map 1, each algorithm performance diverged on Map 2. This difference is attributable to each algorithm respective training models which the Neural Network exhibited a stronger focus on the right-side class, whereas K-NN maintained a balanced classification approach.

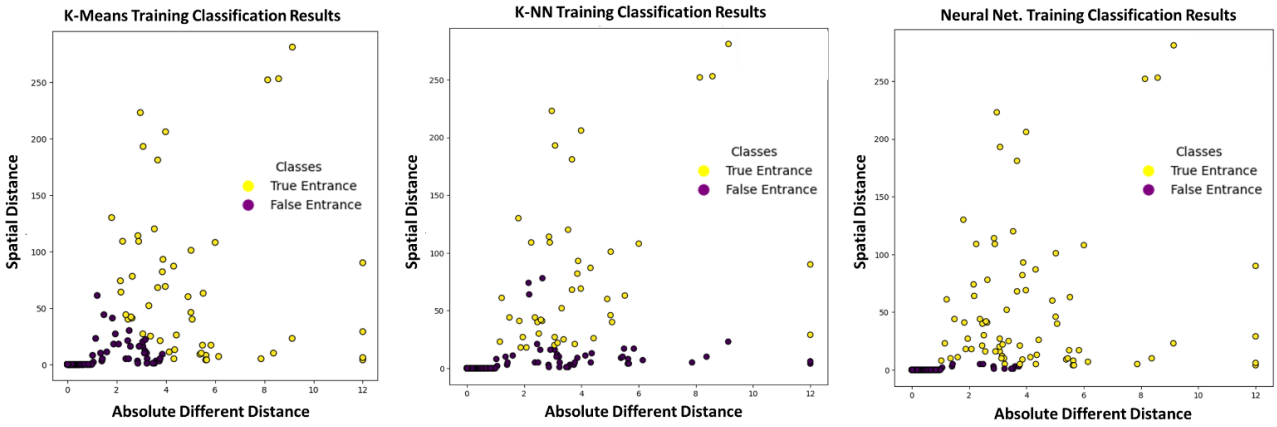


Figure 8 Classifier Algorithm Training for K-Means (Left), K-NN (Middle), Neural Network (Right) Result

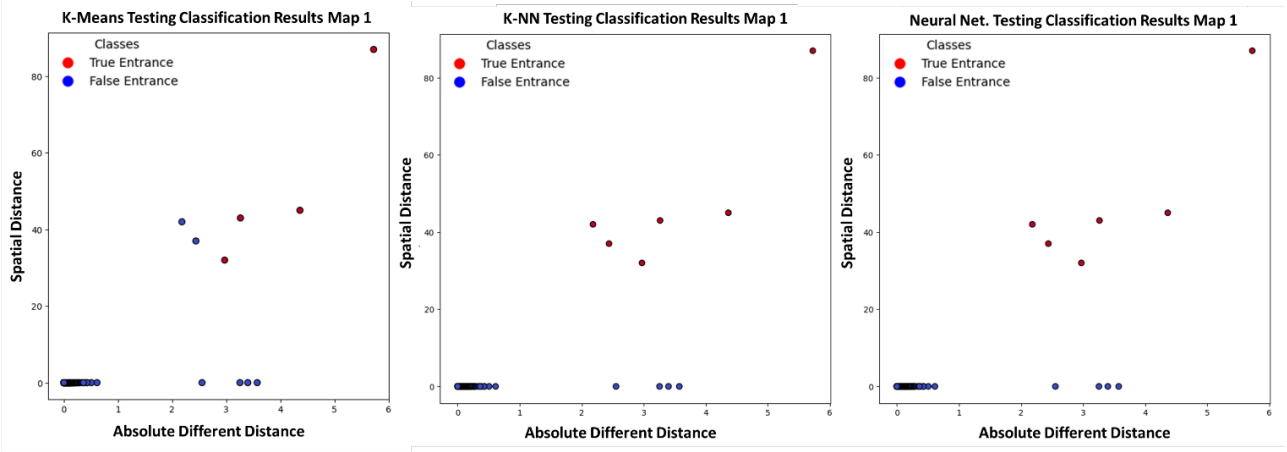


Figure 9 Classifier Algorithm Testing for K-Means (Left), K-NN (Middle), Neural Network (Right) in Map 1 Result

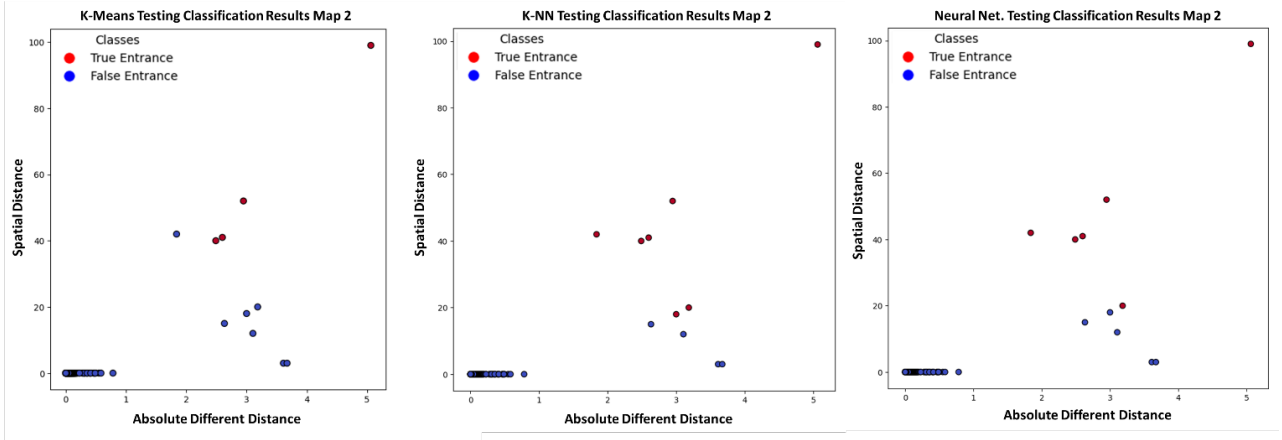


Figure 10 Classifier Algorithm Testing for K-Means (Left), K-NN (Middle), Neural Network (Right) in Map 2 Result

DOI: https://doi.org/10.11113/jurnalteknologi.v89.24309

Overall, based on the exploration data from Maps 1 and 2, K-NN achieved the highest accuracy compared to Neural Network and K-Means classifiers. On Map 1, both K-NN and Neural Network successfully detected all six true entrances, matching the actual map data. However, on Map 2, K-NN was the only classifier to accurately identify all seven true entrances. In contrast, Neural Network misclassified one true entrance as a false entrance, detecting six entrances in total, and K-Means identified only four true entrances out of seven.

These results reflect the differences in the classifiers' training models. K-NN consistently maintained a balanced classification, while K-Means tended to favour the right side of the dataset, and Neural Network exhibited a stronger bias toward the right-side class. Table 2 provides a detailed comparison of accuracy, precision, recall, and F1-score for each algorithm.

Table 2 Performance Metric for testing result for both Map

Algorithm	Average Performance			
	Accuracy	Precision	Recall	F1
K- Means	98.3	98.87	98.53	98.72
K-NN	100	100	100	100
Neural Network	99.46	99.66	99.45	99.59

Based on Table 2, the K-NN classifier algorithm achieved the highest performance, with 100% accuracy, precision, recall, and F1-score for both Map 1 and Map 2 testing results. This was followed by the Neural Network classifier, which averaged 99% across all metrics, and the K-Means classifier, which averaged 98% for all metrics in the testing results for both maps. These results demonstrate that K-NN is the most suitable algorithm for the Target Entrance and Free Space Evaluator Algorithm, providing high reliability in detecting true entrances. The classification data generated by K-NN is well-organized for further processing by subsequent components.

3.3 Multi – Entrance Node Sequencer Algorithm result

The results in the previous sections for the Target Entrance and Free Space Evaluator Algorithm highlight multiple possible entrances detected and evaluated using three types of classifier algorithms. Based on the evaluation of these entrances, the third component was implemented to prioritize and sort the entrances based on each entrance weightage and the distance between the mobile robot and the target. In this study, the maximum velocity of the mobile robot was set as a constant to maintain consistent exploration capabilities for each entrance. To sort the entrances, the default sequence was initially arranged based on the LIDAR sensor's angular readings, starting from 0° to 359°.

The classification result from component two is then processed using Equation 2 to generate an optimized node sequence for the mobile robot to explore. This node sequence ensures the robot explores the area in an optimal time based on the predefined weightage. Each entrance was declared as shown in Figure 11. Table 3 shows the optimum sequence for the mobile robot to reach all entrance for Map 1 and Map 2. The time taken to reach each node was compute based on constant max velocity around **0.14 m/s**.

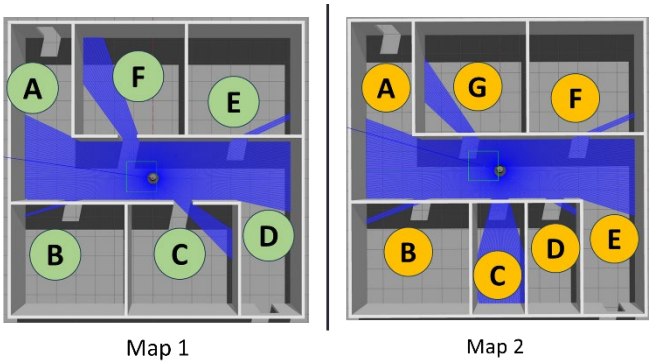


Figure 11 Declared Entranced

Table 3 Optimum node sequence to explore both map

Map	Current Node	Next Node	Distance (m)	Time (s)	Cum. Time (min)
1	C	D	3.891217	27.79	0.46
	D	E	4.795545	34.25	1.03
	E	F	10.658852	76.13	2.30
	F	A	5.559180	39.71	2.96
	A	B	5.135649	36.68	3.57
Total Time Taken to explore Map 1					**3.57**
2	C	D	3.486859	24.91	0.41
	D	E	3.000161	21.43	0.77
	E	F	5.017677	35.84	1.37
	F	G	11.146765	79.70	2.69
	G	A	4.319614	30.85	3.21
	A	B	5.119441	36.57	3.82
Total Time Taken to explore Map 2					**3.82**

Based on Table 3, the most optimal sequence for the mobile robot to explore Map 1 is C > D > E > F > A > B, completed in 3.58 minutes. Similarly, for Map 2, the optimal sequence is C > D > E > F > G > A > B, completed in 3.82 minutes. The time required for each sequence is calculated based on a fixed maximum velocity of 0.14 m/s. For Map 1, the longest segment in the sequence is between node E to node F, covering the longest distance around 10.65 metre and requiring approximately 76 seconds to complete. This makes it the most time-consuming part of the optimal sequence for exploring the entire Map 1. Meanwhile, for Map 2, the longest segment is between node F to node G,

DOI: https://doi.org/10.11113/jurnalteknologi.v89.24309

spanning a distance of around 11 meters and taking approximately 79 seconds to complete.

As shown in the Table 3, the Multi-Entrance Node Sequencer algorithm effectively determines the optimal sequence compared to other node sequencing methods. The results are then compared with simulation model times for the mobile robot to explore each map to evaluate the accuracy of the estimated times generated by the Multi-Entrance Node Sequencer algorithm. The comparison results are presented in Table 4.

Table 4 Performance Metric for testing result for both Map

Map	Est. Time (min)	Sim. Time (Min)	Accuracy (%)
1	3.576243	3.5869	99.7%
2	3.820299	3.9586	96.5%

According to Table 4, the mobile robot takes approximately 3.58 minutes to explore the entirety of Map 1 and 3.96 minutes to explore Map 2. Based on these simulation times, the Multi-Entrance Node Sequencer algorithm demonstrates an accuracy of 99.7% for Map 1 and 96.5% for Map 2 in estimating exploration time. The slight differences in accuracy between the algorithm's estimated times and the simulation times may be influenced by factors such as inertia, friction, and the weight of the mobile robot, which were not accounted for in the algorithm's calculations. These results indicate that the sequence generated by the algorithm, along with its estimated times, exhibits a high degree of accuracy. Given this level of precision, the algorithm is deemed reliable for generating exploration sequences for the mobile robot in various environments and maps.

3.4 Performance for the M – TEN Framework

The M-TEN framework is a combination of first, second and third component develop in this paper. The M-TEN framework start with entrance node acquisition, then evaluate the existing of the node. The evaluated entrance node then being sorted and optimize sequence was generated. The sequence of node then feed to the mobile robot to explore one by one node until the whole map is explored. The mapping progress was using SLAM feature and path planning using Nav2 plugin in ROS software. The performance of M-TEN framework has been evaluated based on these characteristics:

1. The quality of SLAM mapping compared to real map, reflect the performance of the proposed algorithm in exploring each part of the map.
2. The accuracy, time taken, and path trajectory to complete the exploration comparing between traditional frontier exploration, manual method, and M-TEN framework.

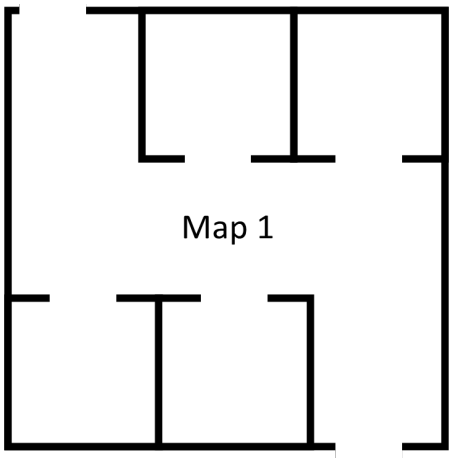
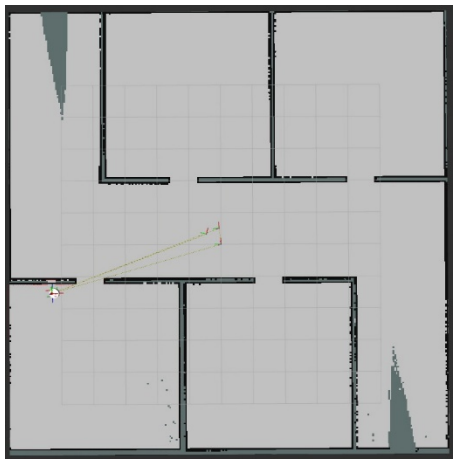
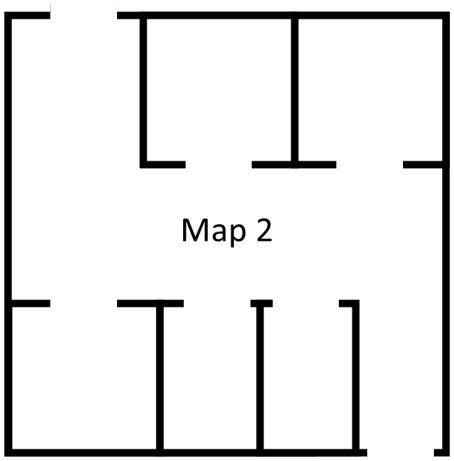
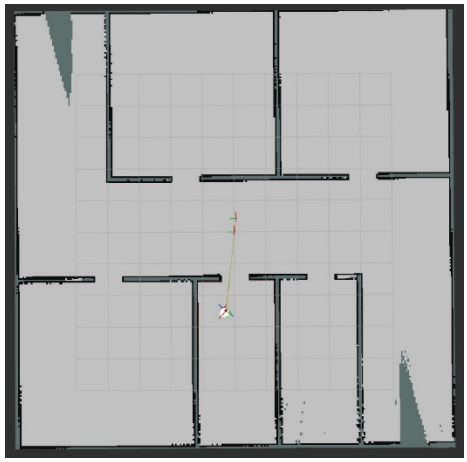
The first evaluated characteristic is the quality of SLAM mapping by using M-TEN algorithm compared to the real map. The SLAM mapping result is shown in Table 5. The mapping gathered the information of point cloud coordinate and illustrate the coverage every time the mobile robot explored new entrance inside the map. The performance evaluated based on the map accuracy, the coverage percentage compared to the real map.

Based on Table 5, the M-TEN framework was able to generate the SLAM mapping for both Map 1 and Map 2. The accuracy and coverage for generated map for Map 1 is 97% for both aspects. However, the accuracy for M-TEN SLAM result for Map 2 decreasing to 92% and coverage remain same with Map 1. The M-TEN framework demonstrated robust performance in environmental mapping and navigation, achieving an impressive 97% coverage in automatic exploration. While Map 2 showed slightly decreased accuracy compared to Map 1 due to localization setup misalignment, the framework maintained high reliability in mapping critical elements.

According to the SLAM results in Table 5, the framework accurately captured and positioned obstacles, entrances, and walls, effectively reflecting the real environment. The combination of Nav and SLAM capabilities enabled the framework to autonomously explore and map the entire space, making it particularly valuable for applications requiring automatic maximum coverage. The generated maps' reliability and high coverage rate validate M-TEN suitability for future applications, despite minor accuracy variations between different mapping scenarios.

DOI: https://doi.org/10.11113/jurnalteknologi.v89.24309

Table 5 Performance evaluation in term of quality of generated map

Real Map	SLAM Result	Accuracy (%)	Coverage (%)
		97%	97%
		92%	97%

The second aspect to evaluate performance of M-TEN is evaluated based on the total time taken to complete all three component (entrance node acquisition, evaluation, and sorting sequence) and also exploring the whole area. In this last evaluation part, the performance will be compared between manual exploration (manual control and refer to real map) and M-TEN framework exploration the whole map autonomously. Based on previous conducted research, the data for Traditional Frontier Exploration was used to be compared in this evaluation [31]. The bar chart in Figure 12 shows the performance between manually control performance, Traditional Frontier Exploration (FE) and M-TEN framework exploration. The comparison includes with accuracy of map and time taken to explore the whole map.

DOI: https://doi.org/10.11113/jurnalteknologi.v89.24309

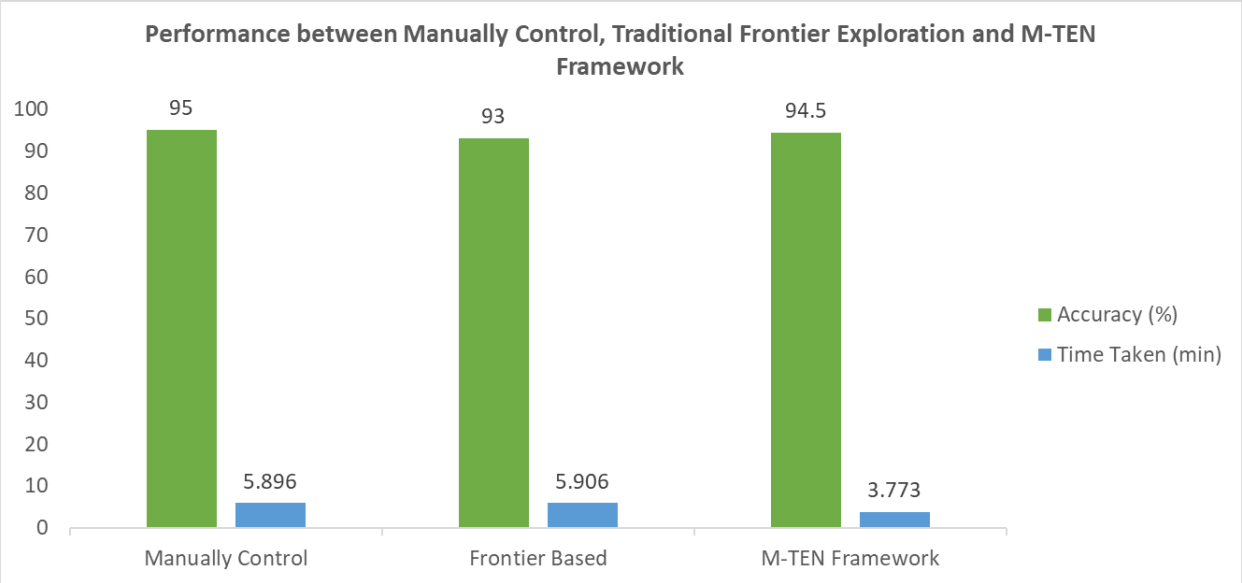


Figure 12 Performance comparing exploration using manual control and M-TEN Framework

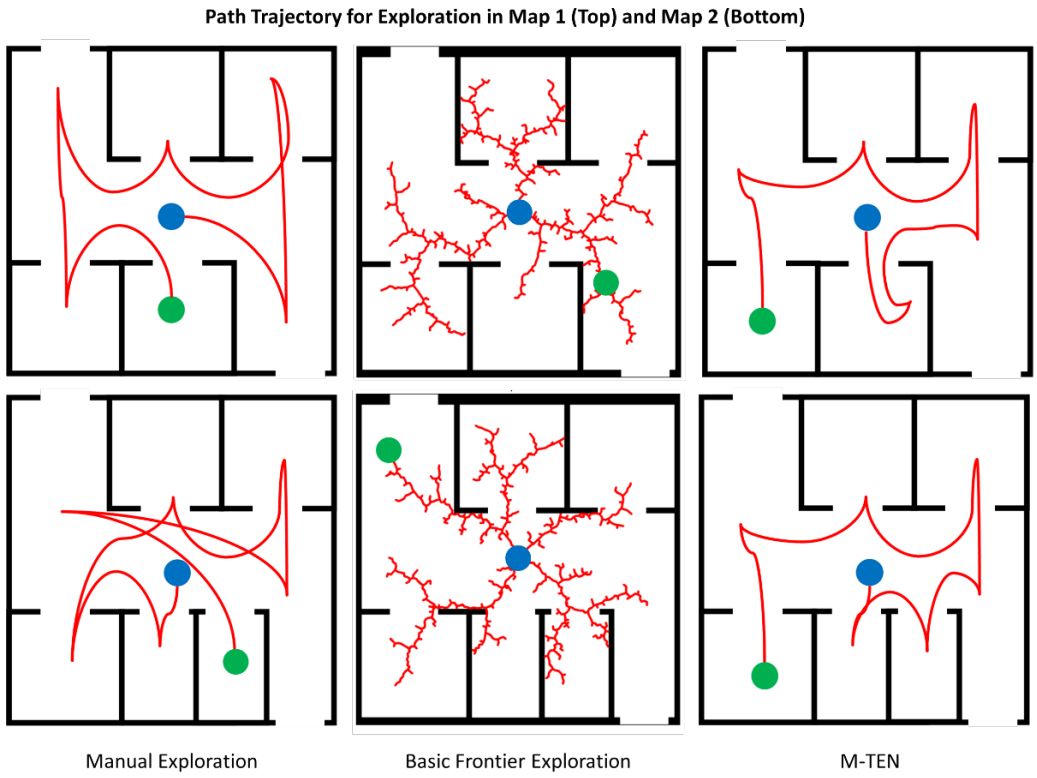


Figure 13 Path Trajectory result for exploration on both map (Blue Circle is Starting point, Green Circle is Final Point)

Based on bar chart in Figure 12, manually control have slightly higher accuracy at 95% followed by M-TEN Framework at 94.5%, and lastly Traditional FE at 93%. The traditional FE take the longest time to cover whole area at 5.9 minutes. Meanwhile, even the manually control has slightly higher accuracy in generated map compared to M-TEN framework, the manually control approach take slightly longer time to cover the whole map with 5.9 minute compared to the M-TEN framework with 3.8 minutes. The performance show

DOI: https://doi.org/10.11113/jurnalteknologi.v89.24309

that M-TEN with high accuracy at 94.5% take shorter time to complete the exploration.

To validate the findings presented in Figure 12, path trajectories were recorded to evaluate the exploration behaviour of each method across both maps, as illustrated in Figure 13. The manual exploration controlled by a human operator demonstrated inconsistent paths and unpredictable decision-making patterns. However, as shown in the top-left and bottom-left trajectories of Figure 13, the mobile robot achieved deeper penetration at each checkpoint under manual control, resulting in the highest mapping accuracy. In comparison, the basic Frontier Exploration method, while exploring the entire area, required longer paths and extended time periods. Moreover, its checkpoints remained primarily in surface areas, leading to lower mapping accuracy and increased completion time. The M-TEN framework, in contrast, demonstrated superior performance with consistent and optimal checkpoint coverage, resulting in both reduced exploration time and high mapping accuracy. This systematic approach of M-TEN proved more efficient than both manual control and basic Frontier Exploration methods.

This is because the M-TEN framework compute all possible entrance, evaluate it, and sort the optimum sequence to ensure the mobile robot can explore all entrance at effective and efficient path with shortest time. Meanwhile, manually control affected by operator decision making to explore which entrance first, and led to operator choose the sequence with longer time and inefficient path.

According to this performance evaluation for three aspect, M-TEN framework successfully outperform the traditional method by acquiring entrance, evaluated it, and sort the optimum sequence. The proposed framework capable to map at high accuracy, with shortest time, and generate reliable map for exploitation and future work.

4.0 CONCLUSION

This study proposed M-TEN, a novel framework designed to enhance mobile robot exploration in unknown environments by accurately detecting possible entrances, verifying their existence, and generating an optimal exploration sequence. In this work, an entrance is defined as a detectable free-space region identified using onboard sensors. The M-TEN framework integrates entrance acquisition, entrance existence evaluation, and multi-entrance sequencing to reduce redundant exploration actions and improve overall exploration efficiency.

To enable robust entrance detection in unknown areas, LiDAR sensing was integrated with a lightweight data filtering process to extract important spatial features. Based on the extracted features, a supervised classification approach was employed to distinguish real entrances from false detections caused by noise or repeated observations from different viewing angles. Among the evaluated classifiers, the K-nearest neighbour (K-NN) method demonstrated superior performance, achieving 100% accuracy and precision on the tested environments. This confirms the effectiveness of data-driven entrance evaluation in reducing redundant or misleading exploration decisions.

Furthermore, the detected entrances were modelled as nodes and organized using a sequencing algorithm that prioritizes exploration efficiency by minimizing travel distance and estimated traversal time. Experimental results show that this multi-entrance sequencing strategy enables the robot to systematically explore all detected entrances with fewer unnecessary movements. Overall system evaluation demonstrated that the M-TEN framework achieved an average map accuracy of 94.5% compared to ground truth and completed exploration tasks significantly faster than manual control, while maintaining consistent coverage.

Overall, this study contributes to develop framework capable to detect free-space define as entrance, then evaluate the existence of free-space and establish the sequencer algorithm to generate optimum sequence to be followed by mobile robot during exploration in unknown area. By explicitly addressing entrance redundancy and prioritization, the proposed M-TEN framework improves exploration efficiency and reduces unnecessary navigation actions compared to conventional approaches.

Several possible improvements can be considered for future development. These include the integration of neural network–based classifiers to enhance adaptability in complex and highly dynamic environments. In addition, the entrance sequencing module can be further optimized using learned cost models or pre-trained datasets to improve scalability when handling a large number of detected entrances. Most importantly, the proposed framework will be validated on real mobile robot platforms and tested in real-world environments to assess its robustness, tolerance to sensor noise, and practical feasibility beyond simulation-based evaluations.

Acknowledgement

This research is fully supported by The Fundamental Research Grant Scheme (FRGS) of the Ministry of Higher Education, under a grant number of FRGS/1/2022/TK08/UNIMAP/03/13 from the Ministry of Higher Education Malaysia. The authors fully acknowledged Ministry of Higher Education (MOHE) and Universiti Malaysia Perlis for the approved fund which makes this important research viable and effective.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Data availability

The data supporting the findings of this study are available from the corresponding author upon reasonable request.

Ethics declaration

Formal ethical approval was not obtained for this study. However, the authors confirm that the research was conducted in accordance with established ethical principles and good research practices, with due consideration given to participant rights, informed consent, confidentiality, and responsible data handling.

References

[1] M. Rybczak, N. Popowniak, and A. Lazarowska, "A Survey of Machine Learning Approaches for Mobile Robot Control," Jan. 01, 2024, *Multidisciplinary Digital Publishing Institute (MDPI)*. DOI: 10.3390/robotics13010012.

[2] C. Benson, I. C. Obasi, D. V. Akinwande, and C. Ile, "The impact of interventions on health, safety and environment in the process industry," *Heliyon*, vol. 10, no. 1, Jan. 2024, DOI: 10.1016/j.heliyon.2023.e23604.

[3] M. Andreoni, W. T. Lunardi, G. Lawton, and S. Thakkar, "Enhancing Autonomous System Security and Resilience With Generative AI: A Comprehensive Survey," *IEEE Access*, vol. 12, pp. 109470–109493, 2024, DOI: 10.1109/ACCESS.2024.3439363.

[4] A. Thakur and S. K. Mishra, "An in-depth evaluation of deep learning-enabled adaptive approaches for detecting obstacles using sensor-fused data in autonomous vehicles," Jul. 01, 2024, *Elsevier Ltd*. DOI: 10.1016/j.engappai.2024.108550.

[5] M. R. Wayahdi, S. H. N. Ginting, and D. Syahputra, "Greedy, A-Star, and Dijkstra's Algorithms in Finding Shortest Path," *International Journal of Advances in Data and Information Systems*, vol. 2, no. 1, pp. 45–52, Feb. 2021, DOI: 10.25008/ijadis.v2i1.1206.

[6] O. R. Chandra and W. Istiono, "A-star Optimization with Heap-sort Algorithm on NPC Character," *Indian J Sci Technol*, vol. 15, no. 35, pp. 1722–1731, 2022, DOI: 10.17485/IJST/v15i35.857.

[7] B. Yamauchi, "A Frontier-Based Approach for Autonomous Exploration," 1997.

[8] N. Thakur, E. Bhattacharjee, R. Jain, B. Acharya, and Y. C. Hu, "Deep learning-based parking occupancy detection framework using ResNet and VGG-16," *Multimed Tools Appl*, vol. 83, no. 1, pp. 1941–1964, Jan. 2024, DOI: 10.1007/s11042-023-15654-w.

[9] M. Antonazzi, M. Luperto, N. A. Borghese, and N. Basilico, "Development and Adaptation of Robotic Vision in the Real-World: the Challenge of Door Detection," Jan. 2024, [Online]. Available: http://arxiv.org/abs/2401.17996

[10] Akos Ferenc Kungl, "Robust learning algorithms for spiking and rate-based neural networks," 2020.

[11] I. Maksymova, C. Steger, and N. Druml, "Review of LiDAR Sensor Data Acquisition and Compression for Automotive Applications," MDPI AG, Dec. 2018, p. 852. DOI: 10.3390/proceedings2130852.

[12] N. D. Quadros, "Unlocking the Characteristics of Bathymetric LiDAR Sensors," 2013. [Online]. Available: www.lidarnews.com

[13] B. Quintana, S. A. Prieto, A. Adán, and F. Bosché, "Door detection in 3D coloured point clouds of indoor environments," *Autom Constr*, vol. 85, pp. 146–166, Jan. 2018, DOI: 10.1016/j.autcon.2017.10.016.

[14] A. C. Murillo, J. Košecká, J. J. Guerrero, and C. Sagüés, "Visual door detection integrating appearance and shape cues," *Rob Auton Syst*, vol. 56, no. 6, pp. 512–521, Jun. 2008, DOI: 10.1016/j.robot.2008.03.003.

[15] T. A. Pham, M. Delalandre, S. Barrat, and J. Y. Ramel, "Accurate junction detection and characterization in line-drawing images," *Pattern Recognit*, vol. 47, no. 1, pp. 282–295, Jan. 2014, DOI: 10.1016/j.patcog.2013.06.027.

[16] R. Mu˜noz *et al.*, "Door-detection using computer vision and fuzzy logic," Granada, 2004.

[17] H. Brehier, A. Breloy, C. Ren, and G. Ginolhac, "Through the Wall Radar Imaging via Kronecker-structured Huber-type RPCA," *Signal Processing*, vol. 214, Jan. 2024, DOI: 10.1016/j.sigpro.2023.109228.

[18] Y. Wang, X. Shao, W. Yao, B. Zhang, G. Sun, and L. Wu, "EHSP: An Efficient Heuristic Sampling-Based Planner for Autonomous UAV Exploration," *IEEE Sens J*, 2024, DOI: 10.1109/JSEN.2024.3462494.

[19] L. A. Fagundes, A. G. Caldeira, M. B. Quemelli, F. N. Martins, and A. S. Brandão, "Analytical Formalism for Data Representation and Object Detection with 2D LiDAR: Application in Mobile Robotics," *Sensors*, vol. 24, no. 7, Apr. 2024, DOI: 10.3390/s24072284.

[20] B. Rolf *et al.*, "A review on unsupervised learning algorithms and applications in supply chain management," 2024, *Taylor and Francis Ltd.* DOI: 10.1080/00207543.2024.2390968.

[21] Q. Zhou and B. Sun, "Adaptive K-means clustering based under-sampling methods to solve the class imbalance problem," *Data Inf Manag*, vol. 8, no. 3, Sep. 2024, DOI: 10.1016/j.dim.2023.100064.

[22] X. Li, M. Wei, and S. Chen, "PointSmile: Point Self-supervised Learning via Curriculum Mutual Information," Jan. 2023, DOI: 10.1007/s11432-023-4085-9.

[23] N. R. Sasmita *et al.*, "Decision Tree versus k-NN: A Performance Comparison for Air Quality Classification in Indonesia," *Infolitika Journal of Data Science*, vol. 2, no. 1, pp. 9–16, May 2024, DOI: 10.60084/ijds.v2i1.179.

[24] P. K. Sarswat, R. S. Singh, and S. V. S. H. Pathapati, "Real time electronic-waste classification algorithms using the computer vision based on Convolutional Neural Network (CNN): Enhanced environmental incentives," *Resour Conserv Recycl*, vol. 207, Aug. 2024, DOI: 10.1016/j.resconrec.2024.107651.

[25] J. Gui *et al.*, "A Survey on Self-supervised Learning: Algorithms, Applications, and Future Trends," *IEEE Trans Pattern Anal Mach Intell*, 2024, DOI: 10.1109/TPAMI.2024.3415112.

[26] M. Balestra *et al.*, "LiDAR Data Fusion to Improve Forest Attribute Estimates: A Review," *Current Forestry Reports*, vol. 10, no. 4, pp. 281–297, Aug. 2024, DOI: 10.1007/s40725-024-00223-7.

[27] S. O. Azarkasb and S. H. Khasteh, "Eligibility traces in an autonomous soccer robot with obstacle avoidance and navigation policy," *Appl Soft Comput*, vol. 164, Oct. 2024, DOI: 10.1016/j.asoc.2024.111889.

[28] B. Kizielewicz, T. Tomczyk, M. Gandor, and W. Sałabun, "Subjective weight determination methods in multi-criteria decision-making: a systematic review," in *Procedia Computer Science*, Elsevier B.V., 2024, pp. 5396–5407. DOI: 10.1016/j.procs.2024.09.673.

[29] C. Wei, J. Wu, Y. Xia, and Z. Ji, "Fast autonomous exploration with sparse topological graphs in large-scale environments," *Int J Intell Robot Appl*, vol. 8, no. 1, pp. 111–121, Mar. 2024, DOI: 10.1007/s41315-023-00318-7.

[30] T. Zhao, X. Lin, F. He, and H. Dai, "Robotic Sorting Systems: Robot Management and Layout Design Optimization," Apr. 2024, [Online]. Available: http://arxiv.org/abs/2404.04832

[31] K. Karur, N. Sharma, C. Dharmatti, and J. E. Siegel, "A Survey of Path Planning Algorithms for Mobile Robots," *Vehicles*, vol. 3, no. 3, pp. 448–468, Sep. 2021, DOI: 10.3390/vehicles3030027.