

The Effect of Artificial Neural Network Towards the Number of Particles of Rao-Blackwellized Particle Filter using Laser Distance Sensor

Amirul Jamaludin¹, Norhidayah Mohamad Yatim², Zarina Mohd Noh³

Centre for Telecommunication Research & Innovation (CeTRI)-Fakulti Kejuruteraan Elektronik & Kejuruteraan Komptuer (FKEKK), Universiti Teknikal Melaka (UTeM) Malaysia, Melaka, Malaysia

Abstract—Rao-Blackwellized particle filter (RBPF) algorithm aims to solve the Simultaneous Localization and Mapping (SLAM) problem. The performance of RBPF is based on the number of particles. The higher the number of particles, the better the performance of RBPF. However, higher number of particles required high memory and computational cost. Nevertheless, the number of particles can be reduced by using high-end sensor. By using high-end sensor, high performance of RBPF can be achieved and reduced the number of particles. But the development of the robot came at a high cost. A robot can be equipped with low-cost sensor in order to reduce the overall cost of the robot. However, low-cost sensor presented challenges of creating good map accuracy due to the low accuracy of the sensor measurement. For that reason, RBPF is integrated with artificial neural network (ANN) to interpret noisy sensor measurements and achieved better accuracy in SLAM. In this paper, RBPF integrated with ANN is experimented by using Turtlebot3 in real-world experiment. The experiment is evaluated by comparing the resulting maps estimated by RBPF with ANN and RBPF without ANN. The results show that RBPF with ANN has increased the performance of SLAM by 25.17% and achieved 10 out of 10 trials of closed loop map by using only 30 particles compared to RBPF without ANN that needs 400 particles to achieve closed loop map. In conclusion, it shows that, SLAM performance can be improved by integrating RBPF algorithm with ANN and reduces the number of particles.

Keywords—SLAM; occupancy grid map; Rao-Blackwellized particle filter; artificial neural network; laser distance sensor

I. INTRODUCTION

Simultaneous localization and mapping (SLAM) plays an important role in robotics, and particularly in a mobile robot system. SLAM's primary objective is to jointly measure the robot's position as well as map the surrounding of the robot [1]–[5]. The essential of the SLAM algorithm is to map the unknown environment and at the same time exploring the environment. Then, the resulting maps can be used for various applications such as autonomous navigation and search and rescue.

SLAM algorithm commonly used the occupancy grid map (OGM) as a map representation. The maps generate precise metric maps that are close to the detail environmental representations [6]–[8]. This maps require an accurate position of the robot, which makes it a difficult process because of the lack of efficiency of odometric system. The map generated

using raw odometry data only as the position of the robot cannot be sustain and suffers a serious error due to the dead reckoning of odometry system [9]. But these problem can be improved by using Rao-Blackwellized particle filter (RBPF) which can compute more accurate position of the robot. The map is more accurate and globally consistent using RBPF approach. The accuracy of RBPF algorithm is based on the number of particles that required high memory and computational cost [10]–[13]. The higher the number of particles, the more accurate the robot's position and map construction. Therefore it required high memory usage and computational cost to solve the SLAM problem with occupancy grid.

In addition, to achieve high accuracy of occupancy grid map, many notable research have carried out SLAM using high-end sensor to achieve high accuracy map [11], [14]–[18]. Nevertheless, the development of the robot came at a high cost. Therefore, a low-cost sensor could be incorporated to the robot as an alternative to reduce the cost. However, low-cost sensors presented challenges of creating good map accuracy. This is due to the low measurement accuracy of the sensors, which can affect map construction. This problem can be solved by integrating the SLAM technique which is RBPF algorithm with an artificial neural network (ANN) while using low-cost sensors. ANN has been used with occupancy grid maps to interpret noisy sensor measurements and achieved better accuracy of the map [5], [19], [20]. Furthermore, [19] and [5] also shows the number of particles can be reduced with better accuracy of the sensor measurement. Therefore, ANN integrated with RBPF algorithm can reduce the computational cost of the particle filter and achieve better accuracy of OGM. Hence, in this paper, ANN integrated with RBPF algorithm is introduced to improve the sensor measurement and enhance the SLAM technique by using only low-cost sensor.

This paper aims to reduce the number of particles consumption by improving the measurement accuracy of a low-cost laser distance sensor (LDS) and subsequently improve the performance of the SLAM algorithm by incorporating ANN technique with RBPF algorithm. The organization of this paper is as follows: Section 2 reviews the past studies related to the RBPF, sensor measurements accuracy and ANN. Section 3 describes the methodology of the ANN model training and RBPF algorithm framework integrated with ANN. Section 4 analyzes the performance of ANN model after the training and

reports the results of the RBPF algorithm after integrating with ANN. Lastly, section 5 concludes the finding of this paper.

II. RELATED WORK

The Rao-Blackwellized Particle Filter, also referred to RBPF is employed in the grid-based SLAM algorithm. RBPF is a version of particle filter-based SLAM which is an effective implementation of particle filter-based SLAM. By using Gaussian substructures in the model, RBPF improves the efficiency of the particle filter algorithm [21]. In this approach, RBPF approximates the belief distribution of the robot's pose, while each particle keeps an individual map of the environment, as depicted in Fig. 1. The main contribution of this method is reducing the number of particles used [21]–[24]. This method uses only the most recent observation as well as the most accurate proposal distribution. This can greatly reduce the robot pose uncertainty during filter's prediction step. In addition, the resampling operation is performed selectively in order to address the particle depletion problem. For example in [25], to achieve high accuracy of grid-based SLAM, the classical particle filter required 10000 particles while in [9], RBPF only need 100 particles to achieve high accuracy of SLAM. One of the ROS package implement RBPF algorithm is Gmapping package [26]–[32]. However, since RBPF is based on particle filter-based SLAM, the performance of the algorithm relies on the number of particles. The higher the number of particles, the better is the SLAM performance. This is due to the computation of RBPF itself is based on probability distribution. The higher the number of particles, the higher is the probability to estimate the correct poses and map. As a result, better accuracy of SLAM can be achieved. As in [10], the author shows that with higher number of particles, the map constructed is more accurate than the lower number of particles.

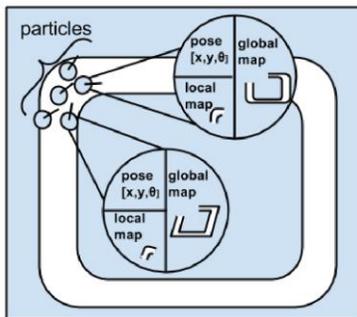


Fig. 1. Each particles contains a hypothesis of robot pose and maintain its own map [10]

A consistent grid-based SLAM algorithm capable to perform loop closure in an unstructured environment. But it requires high precision of the laser scanner to obtain the best hypothesis to build the grid map of the environment. The high precision sensor measurement also can reduce the number of particle usage. For example in [30] and [32], both paper used gmapping package and used the default number of particles which is 30. In [30] the authors successfully achieve high accuracy SLAM by using Hokuyo urg-04lx-ug01 2D lidar which is in high-cost sensor category. While in [32], the map produced is not very accurate since the author only used RGB-D sensor Kinect of Xbox 360 which is low-cost sensor to

perform the SLAM task. But as mention before, low-accuracy SLAM can be mitigated by increasing number of particles. In [10], the author increased the number of particles to 500 to achieve better accuracy SLAM since they used low-cost sensor. However, higher number of particles might suffer from forbidding memory burden and higher computational cost. This problem can be overcome by integrating the SLAM technique with an artificial neural network (ANN) while using low-cost sensors [5], [19], [20], [33]. The noisy dataset from the sensor of the mobile robot are used to train the ANN learner. Afterwards, the ANN model is then applied in the RBPF SLAM algorithm to execute the SLAM task.

In this paper, there are two strategies of dataset that have been reviewed to train the ANN network. Firstly, the training network using the position of each of the grid cells of OGM [5], [19], [33]. Secondly, by using the distance from sensor to obstacles [20]. Firstly, in [19], the author collect the dataset from the front six infrared sensor of E-puck mobile robot as depicted in Fig. 2. Then, the dataset is used as the input layer of the ANN training as shown in ANN configuration in Fig. 3. This ANN is trained to estimate the occupancy probability value x, y position of the cell. The same architecture of ANN is used in [5], [33]. The authors used the front four sensors' measurements of Khepera III robot that are closest to the cell position as the input of ANN. The same target OGM output of the ANN model which is x and y position of the OGM's cell (C_{xy}) are used to train the input as shown in Fig. 3.

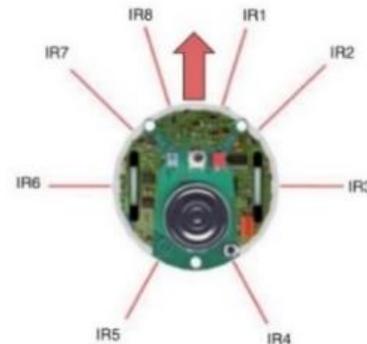


Fig. 2. E-puck mobile robot is equipped with eight infrared sensors [19]

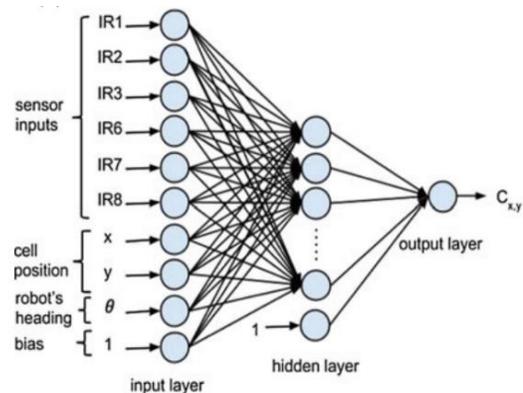


Fig. 3. A nine-input ANN consisting of infrared sensors, cell position and heading of the robot [19]

While in [20], the dataset is collected based on the reading of the sensor values that contain the estimated distance to obstacle. Then this dataset is trained using ANN to correct the distance measurement to the obstacles by using the real distance as the target output of the ANN. The grid map for the observed region is then built using this corrected measurement. To compute the grid map estimation in this region, only two values are taken into account: 0 for open space and 1 for occupied space (i.e., objects in the space). In the context of computation time, estimation of ANN by using distance to the obstacles is considered faster [20] than estimation by using each of the grid cells [5], [19], [33]. This is due to the estimation by distance does not evaluate cell by cell and only taking value 0 and 1 as mention above. While in the latter approach, the estimation of the map computes each cell of the region by calculating x and y cell's occupancy value of the OGM. Cell by cell evaluation can cause slow computation time especially in large environment that required many cells to build the OGM. Hence, real time implementation is not feasible. Therefore, ANN based on the distance is implemented in this paper since it is faster and more suitable for real-time implementation due to the computation does not require for each grid cell but for the entire detected region. The presented methods of ANN training are summarized in Table I.

TABLE I. COMPARISON OF ANN TRAINING METHOD

| ANN training method | Input/Output | Pros/Cons |
|---------------------------|--|---|
| Based on cell-by-cell OGM | Input: Multiple sensor measurement value, cell position (x, y) and robot heading, θ Target output: Occupancy probability value | Pros: Exploit adjacent sensor measurements and less depending on accuracy of the sensor Cons: slow computation |
| Based on distance | Input: Sensor measurement value Target output: Actual distance | Pros: Fast computation Cons: more depending on accuracy of the sensor |

III. METHODOLOGY

For the methodology phase, the operation is divided into four phases which is sensor data collection, ANN training, ANN model integrated with RBPF algorithm and lastly, evaluation of the resulting map. Firstly, the LDS data points from the laser distance sensor LDS-01 are collected. The data points are collected from the real-world sensor. These data are collected between the range measurement of 0.12 to 3.5m as shown in Fig. 4(a). At each interval of 0.1m, 2000 measurements were collected. Since LDS-01 sensor is capable to sense 360 degree of the surrounding, the data are taken only at 0 degree. This is because only the measurement at 0 degrees is perpendicular to the wall as shown in Fig. 4(b). The measurements at other degrees facing the wall would not be precisely equal to the ground truth but at longer range as they are slightly slanted towards the wall. Then, these data were used as the input of the ANN training. The actual distance between the mobile robot and the wall was used as the reference or target output.

After the data collection, these data are trained to build ANN model in the second phase. This ANN training employs a multilayer feed-forward network. The architecture of the

network is made up of an input layer, a hidden layer and a single neuron for output layer as depicted in Fig. 5. In this paper, a tangent sigmoidal activation function and a linear transfer function were used for the hidden and output layers, respectively.

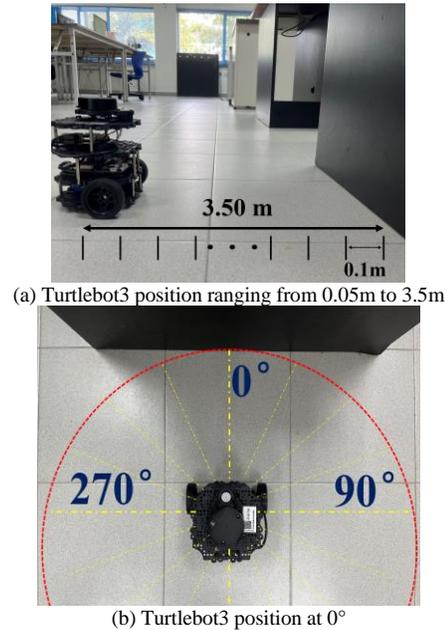


Fig. 4. Position of the turtlebot3 to the wall in real-world experiment

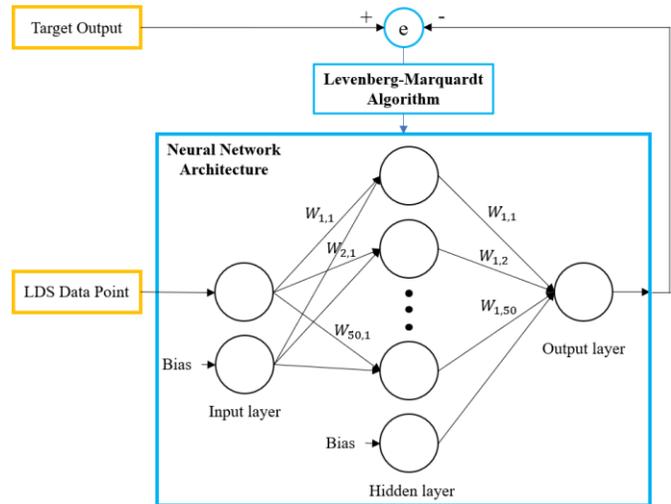


Fig. 5. Architecture of ANN model training

After the ANN structure is established, the next step is training the model. A set of input and reference data pairs as well as a training rule are presented during the training procedure. As mention in the first phase, the input is the LDS data point that have been collected and the reference data is the actual distance of the sensor to the wall. The ANN generates its own output from the input and compares it to the reference data. To ensure the difference of the output is small as possible to the reference value, the interconnection weights, W between the nodes are determined by using training rule. The challenge with learning is finding the optimal weights W combination

and attempts to minimize the mean-square-error (MSE) between the reference value and the predicted output. The use of a backpropagation algorithm is the most widely used training rule for error minimization. An improved backpropagation algorithm, the Levenberg Marquardt (LM) algorithm, come out to be a faster and more efficient approach for training medium-sized feedforward neural networks [34]–[37]. Therefore, in this paper, the LM algorithm is employed. The output of the network is also computed based on the number of neurons in the hidden layer. The number of neurons in the hidden layer is determined by the response of the output that resulted in the smallest MSE error. At the end of the training phase, 50 number of neurons is selected as the number of neurons which has managed to achieve MSE value of 9.55×10^{-5} .

After the desired ANN model is obtained, the next phase is to integrate the model with Rao-Blackwellized particle filter (RBPF) algorithm as shown in Fig. 6. The RBPF algorithm implemented in Gmapping package of Robot Operating System (ROS) is used in this paper. RBPF requires odometry data as well as sensor observations to solve grid-based SLAM problem. The new sensor observation that has been improved using ANN is integrated in this phase. The main idea of RBPF is to use odometry data, $u_{1:t-1}$ and observations, $z_{1:t}$ to estimate the trajectories of robot, $x_{1:t}$ and map, m . This joint posterior, which is written as $p(x_{1:t}, m|z_{1:t}, u_{1:t-1})$ can be factored as follows:

$$p(x_{1:t}, m|z_{1:t}, u_{1:t-1}) = p(m|z_{1:t}, x_{1:t}) \cdot p(x_{1:t}|z_{1:t}, u_{1:t-1}) \quad (1)$$

The computations are made simpler by factorization, which enables the process to be executed in two steps. First, odometry and observation data can be used to estimate the robot's trajectory, $p(x_{1:t}|z_{1:t}, u_{1:t-1})$. Once the $x_{1:t}$ and $z_{1:t}$ are known, the map $p(m|z_{1:t}, x_{1:t})$ can be computed.

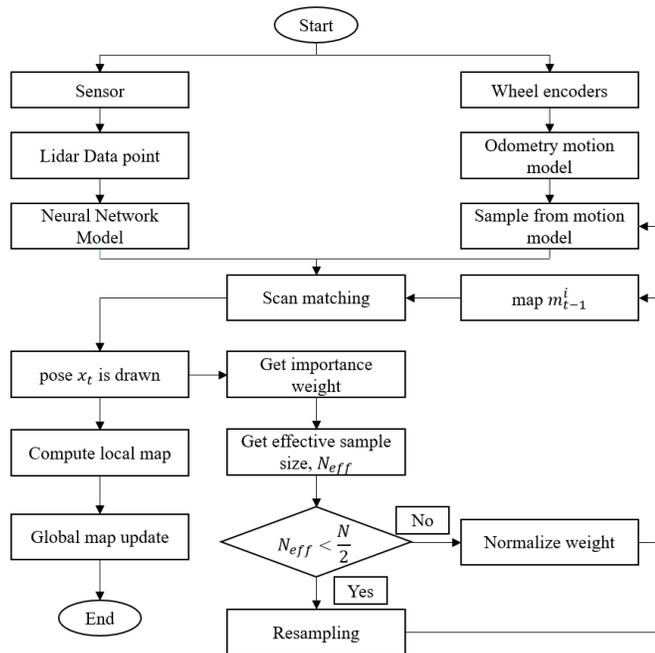


Fig. 6. RBPF algorithm integrated with ANN flow chart

The estimation is more complex with respect to $p(x_{1:t}|z_{1:t}, u_{1:t-1})$. A particle filter is used for this purpose where any particle reflects the robot's potential trajectory. Based on the potential trajectory, an individual map for each particles can be computed. Then, the full map is built by the corresponding particles. The sampling importance filter (SIR) is commonly used in this algorithm. The SIR function is used to select the particles with the highest probability and the output of the algorithm is the the associated map. In order to improve the algorithm, the Rao-Blackwellized SIR filter uses the most recent sensor observations and odometry readings where available. Specifically, in RBPF, the approximation of the trajectory $p(x_{1:t}|z_{1:t}, u_{1:t-1})$ was remodeled to $p(x_{1:t}|m_{t-1}^i, x_{t-1}^i, z_{1:t}, u_{1:t-1})$ [31]. Scan matching is used to match the observations to the map that has been created so far, optimize observation probabilities, and provide information about the most likely poses of the robot. In this step, accuracy of the sensor measurement is important to get better observation for scan matching step and obtain better pose estimation.

After that, particles are resampled according to their weight. Particles with higher weight will be most likely to be resampled for the next generation. All particles have the same weight after resampling. To avoid the resampling from removing good particles, a careful resampling step is taken. Hence, a selective resampling technique which is effective samples size, N_{eff} is proposed which is to decide when to perform a resampling step and is described as:

$$N_{eff} = \frac{1}{\sum_{i=1}^N (\omega^i)^2} \quad (2)$$

where ω^i represents the normalized weight of i -th particle. The weights of the samples are approximately equal if they are close to the target distribution. As the samples deviate from the target distribution, their weights variance increases and N_{eff} decreases. Every time N_{eff} falls below $N/2$, the resampling procedure is initiated (where N is the number of particles used in the filter). This greatly reduces the risk of replacing useful particles, as resampling is performed only when necessary and the cumulative number of such operations is reduced.

Last phase is to evaluate the performance of RBPF integrated with ANN in real-world experiment. The RBPF with ANN will be compared to RBPF without ANN. The maps that have been obtained from both algorithms respectively will be evaluated with ground truth map using number of inliers evaluation. To evaluate the resulting maps quantitatively, similarities between both maps are measured using number of inliers that are obtained from RANSAC algorithm. RANSAC algorithm can calculate the similarity point (inliers) between resulting maps and ground truth map. The higher the number of inliers, the better the similarity of the resulting map and ground truth map and the better the performance of the map constructed. Ground truth map is obtained by using RBPF algorithm with high number of particles. This is due to the high number of particles can obtained high accuracy of map. In this paper, 1000 particles is used to obtain the ground truth map. Additionally, the robot explored the environment in smaller loops. This process is repeated until a satisfactory ground truth map is obtained. By exploring in smaller loops, the

accumulated error of robot's state estimate is kept to the minimum and robot's trajectory maintain on the correct path and did not diverged. Turtlebot3 robot platform is used in this paper to navigate the environment as shown in Fig. 7. The experiment was conducted at Faculty of Electronics and Computer Engineering Universiti Teknikal Malaysia Melaka. The robot was set to explore wing A of the faculty within the red rectangle by following the path marked using red arrow. The size of this real-world environment is approximately 43 x 16 meters. The Turtlebot3 explored the environment using the teleop operation. The average speed of the robot was about 0.12 m/s and simultaneously recorded by the Rosbag tool. Rosbag tool allows recording and playing back the data of the robot that have been recorded. Hence, the recorded data which is the same data can be applied to other algorithms. This way, RBPF algorithm integrated with ANN and without ANN can be evaluated in equal condition. After that, the RBPF with ANN algorithm was observed if there is any improvement after the ANN integration.



Fig. 7. Layout of wing A of faculty of electronics and computer engineering

IV. RESULT

The result is divided into two sections. In the first section, ANN model that have been trained is analyzed by comparing the sensor measurement with ANN and without ANN. Afterwards, in the second section, the RBPF algorithm after integrating with ANN is analyzed by comparing the resulting maps with the ground truth map.

A. Performance of ANN

The training of the ANN model has managed to achieve MSE value of 9.55×10^{-5} . 50 number of neurons are selected as it has achieved the lowest MSE value. The synaptic weights in each layer are computed and adjusted according to the lowest mean-squared error (MSE). The training automatically stops when generalization stops improving, as indicated by the lowest MSE of the training samples. Finally, weight and bias vectors for this ANN training were determined after several attempts and ANN model is established.

Afterwards, the ANN model that have been established is analyzed. The performance of the model is analyzed based on the comparison of the sensor measurement before (without ANN) and after the ANN model (with ANN) is used. The data of both sensor measurement for every interval of 0.1m were averaged and plotted in the graph respectively as shown in Fig. 8. The graph shows that the sensor reading without ANN

(green line) is slightly different from the actual distance (red line). The difference of the sensor measurement without ANN and actual distance can be observed in more detail in Fig. 9. The graph shows that the error of sensor measurement without ANN to the actual distance is consistently higher. It is also observed that when the range of the distance increases, the distribution of sensors measurements is wider. Fig. 10 shows the histogram of the LDS sensor measurement at 3 meter distance is wider than the histogram of sensor measurement at 1 meter distance as depicted in Fig. 11. Therefore, the standard deviation of the sensor measurements also increases as the distance increase as shown in Fig. 13.

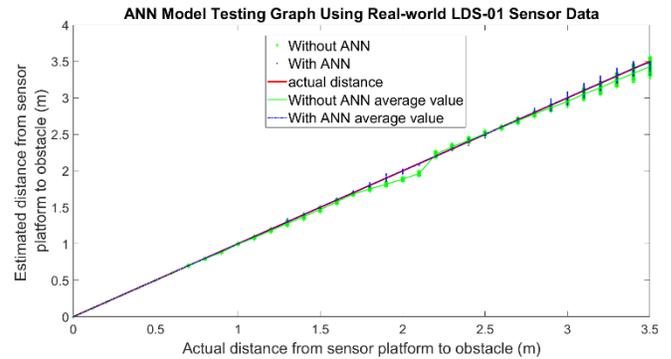


Fig. 8. Graph illustration for the data of ANN model testing using real-world LDS-01 sensor measurements data

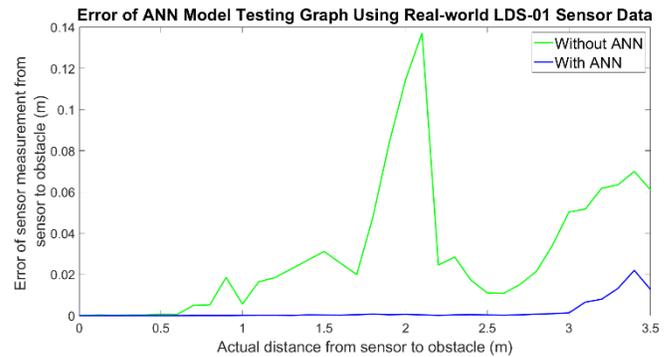


Fig. 9. Comparison of error of sensor measurement without ANN (green) and with ANN (blue)

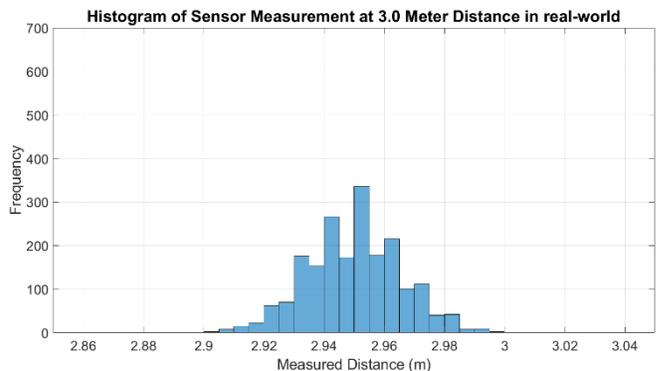


Fig. 10. Histogram of LDS sensor measurement at 3 meter distance

After the LDS sensor data has been applied to the ANN model that have been trained, the value of the data with ANN

(blue line) and the actual distance value (red line) has minimal difference such that the blue and red line are in the same axis as shown in Fig. 8. The error of the sensor measurement also decreased remarkably as shown in Fig. 9. It shows that the non-linearity and the error of the sensor readings are significantly reduced by using ANN. In addition, the histogram of sensor measurement at 3 meter distance using ANN is narrower than the histogram without ANN as shown in Fig. 12. The standard deviation of the sensor measurement with ANN (blue line) is also consistently lower than the standard deviation of the sensor measurement without ANN (green line) as shown in Fig. 13. From these results, it shows that there is a significant improvement of the accuracy of the LDS sensor measurements.

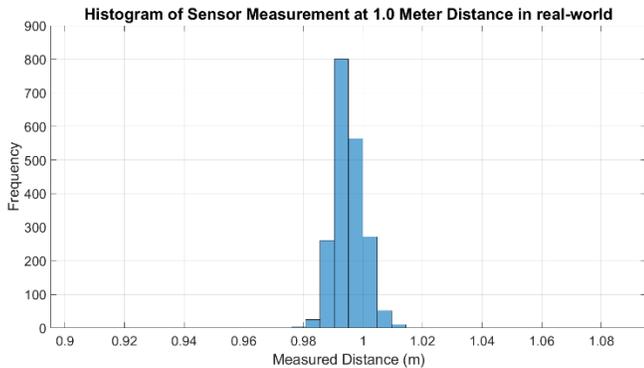


Fig. 11. Histogram of LDS sensor measurement at 1 meter distance (without ANN)

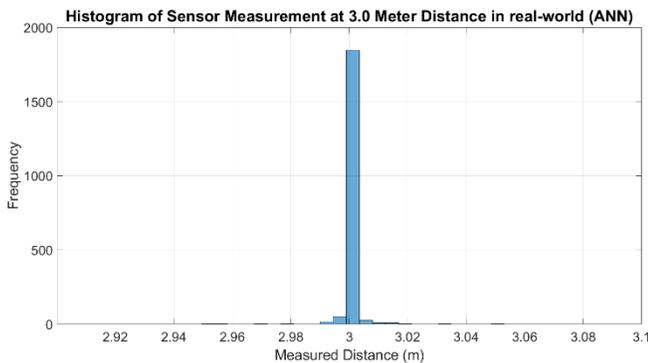


Fig. 12. Histogram of LDS sensor measurement at 3 meter distance (with ANN)

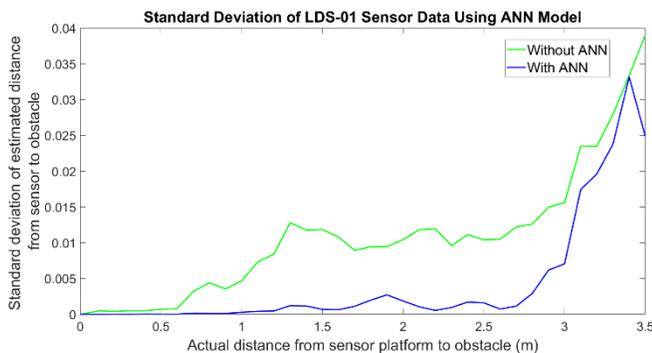


Fig. 13. Standard deviation of LDS-01 sensor data using ANN model before and after testing

B. Performance of RBPF Algorithm

In this experiment, the number of particles of Gmapping package which is 30 to 400 particles is used. The maps are obtained in the Occupancy Grid map (OGM) representation and saved in pgm format using map saver from map server package. The size of each grid cells of the map is set to 5 cm^2 . In OGM, the black cells are considered occupied, white cells are considered free cells and grey cells are the unknown region that has not been explored yet as shown in Fig. 14. The data that have been recorded is tested 10 times resulting 10 grid maps. Then, the maps that have been obtained using ANN and without ANN are evaluated using number of inliers evaluation to ensure for a consistent result. The dimension of the map is 1056×608 cells. Since, the size of the grid cell is 5 cm^2 , this make the size of the real environment in Fig. 14 is approximately $52.8\text{ m} \times 30.4\text{ m}$. Within this map, the explored area travelled by Turtlebot3 mobile robot is approximately $43\text{ m} \times 16\text{ m}$ as stated in Section III.

Primarily, the particles used for the first experiment is 30. From the resulting maps, it is observed that, RBPF algorithm that is not integrated with ANN cannot achieved closed loop condition out of 10 trials as shown in Fig. 15. Most of the resulting maps of RBPF algorithm without ANN show that the algorithm cannot recognize the revisited environment at the loop closure point (green circle) as shown in Fig. 15. This is due to the low accuracy of the LDS sensor measurement reading and consequently effects the RBPF algorithm. To achieve all closed loop condition without using ANN, further experiment is carried out by increasing number of particles. After several trials with increasing number of particles (100, 200 and 400), 400 particles used in the RBPF algorithm without ANN managed to attain loop condition for all trials which is 10 out of 10 as shown in figure 16. The detail results of RBPF with number of particles used are included in Table II. As shown in the table, for 100 and 200 particles, the resulting maps have achieved 4 and 6 closed loop condition out of 10 trials, respectively.

TABLE II. RESULTING MAP CONDITION ACCORDING TO NUMBER OF PARTICLES

| SLAM Algorithm | Number of particles | Closed loop map out of 10 trials | Average number of inliers |
|------------------|---------------------|----------------------------------|---------------------------|
| RBPF without ANN | 30 | 0/10 | 53 |
| | 100 | 4/10 | 47.9 |
| | 200 | 6/10 | 48.5 |
| RBPF with ANN | 400 | 10/10 | 46.9 |
| | 30 | 10/10 | 58 |
| | 100 | 10/10 | 58.3 |
| | 200 | 10/10 | 61.6 |
| | 400 | 10/10 | 67.8 |

It shows that by increasing the number of particles, the accuracy of the grid map can be improved. However, higher number of particles may suffer from limitation of memory capacity and higher computational cost.

Fig. 17 and 18 represent the computational consumption diagrams obtained by the system monitor of Ubuntu for 30 and 400 particles of RBPF algorithm, respectively. The specifications of the computer used are AMD Ryzen 5 3600 GPU 6-core processor with 16GB of RAM. The data is taken after one minute of the execution of one of the tests. It can be observed that RBPF with 400 particles consumed up to 100% of the most cores available during the lapse while 30 particles only consume around 30 to 40% of the cores available. This shows that, higher computational cost is needed to execute higher number of particles.

However, ANN can overcome this problem as mention before by improving the LDS sensor measurement. The ANN model integrated with RBPF algorithm (using 30 particles) is tested 10 times. From the results, it is observed that, RBPF algorithm integrated with ANN have achieved all closed loop map condition which is 10 times out of 10 trials by using only 30 particles. Sensor reading by using ANN has increased the measurement likelihood hence resulting a more accurate estimate in robot localization in the RBPF algorithm. As the localization error has decreased, the robot is able to recognize the revisited area. RBPF with ANN also has proven to achieve this condition with less number of particles more consistently. As a result, we can conclude that the particle consumption can be reduced by integrating ANN with the RBPF algorithm and overcome the computational cost suffers from the high number of particles remarkably. Further experiment by increasing number of particles (100, 200 and 400) is also tested by using ANN. The performance of resulting maps is increased and all resulting maps achieved closed loop condition. As mention in section III, the performance of the resulting maps is compared with the ground truth map (Fig. 14) by using number of inliers. The number of inliers is stated in Table I along with the number of particles.

Then, Fig. 19 plots the performance of the resulting maps by using the number of inliers (y-axis) vs the number of particles (x-axis). As shown in the graph, by using ANN integrated with RBPF, the performance of the resulting maps (red) is consistently higher than the resulting maps without ANN (blue). The performance using ANN is increasing with the number of particles while for RBPF without ANN, the performance is decreasing. We believe that the performance without ANN is decreasing because of the dimension of the resulting maps (without ANN) is less similar to the ground truth map. By using 400 particles (without ANN) better pose estimation is achieved compared to lesser particles. Due to that, closed-loop map condition is achieved when the robot explored the environment and encountered the initial position. Although closed loop map condition is achieved, the dimension of the resulting map is not consistent due to the nonlinearity error of the LDS sensor measurement. As shown in the graph, 30 and 200 particles have achieved higher number of inliers, while 100 and 400 particles achieved lower number of inliers. Due to that, the performance of the resulting maps is not consistent

along the number of particles. Fig. 20 shows one of the examples of the number of inliers point between resulting map (without ANN) and ground truth map by using 400 particles. The dimension of the resulting map (without ANN) is 15.7 m x 42.6 m while ground truth map is 15.9 m x 43 m. Since the dimension of the map is slightly smaller to the ground truth map, hence, only 42 number of inliers point is obtained.

C. Discussion

It is observed that when using ANN, the quality of the resulting maps is increasing with the number of particles as expected. This is due to the more accurate sensor measurements. Better sensor measurement improved the observation of the robot for scan matching step and obtained better pose estimation as mentioned in Section III. Due to that lower number particles can be used to achieve higher performance of the map building and closed loop condition. For this experiment, only 30 particles are used to accomplished the closed loop condition. Besides, better resulting map is achieved and more similar to the ground truth map by increasing the number of particles using ANN compared to without ANN. As shown in Fig. 21, the number of inliers point between the resulting maps (with ANN) and ground truth map is higher which is 77 inliers compared to without ANN which is 42 as shown in Fig. 20. This is due to the dimension of the map (with ANN) which is 15.95 m x 43.10 m is more similar to the ground truth map (15.9 m x 43 m) compared to without ANN (15.7 m x 42.6 m). Based on the average number of inliers, the overall performance of the resulting maps by using RBPF with ANN has increased by 25.17%. In conclusion, with better accuracy of the sensor measurement integrated with ANN reduces the number of particles consumption. Therefore, ANN can reduce the computational cost of the RBPF algorithm and achieve better accuracy of OGM by using only low-cost sensor.

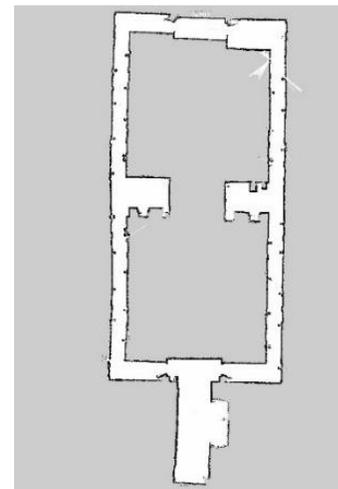


Fig. 14. Ground truth map of the real-world environment



Fig. 15. Non-closed loop map without ANN

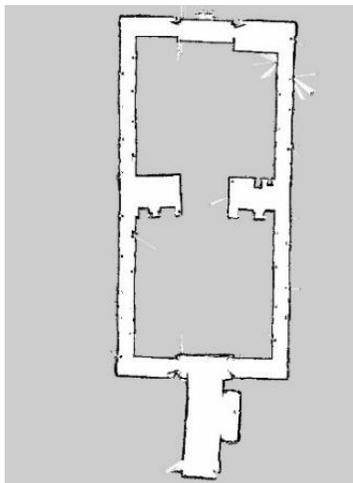


Fig. 16. Closed loop map with ANN

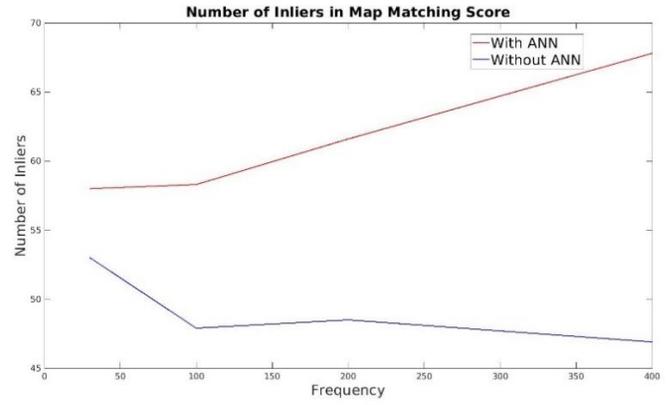


Fig. 19. Map performance by using number of inliers between resulting map and ground truth map

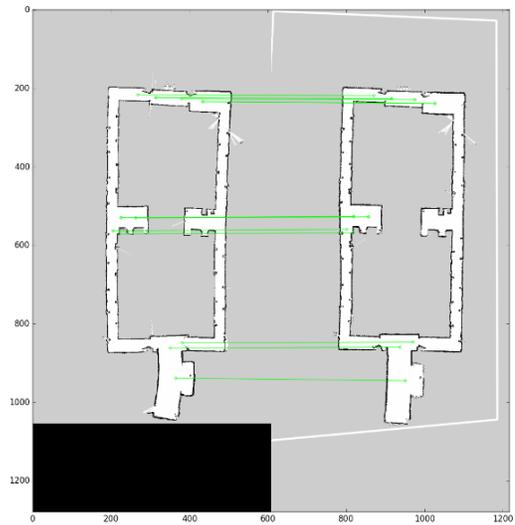


Fig. 20. 42 number of inliers points between resulting map (without ANN) and ground truth map

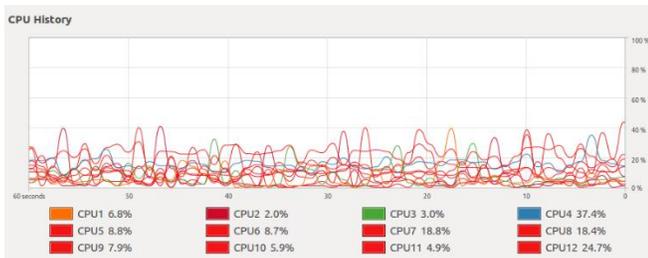


Fig. 17. CPU consumption in one minute of the RBPf using 30 particles.

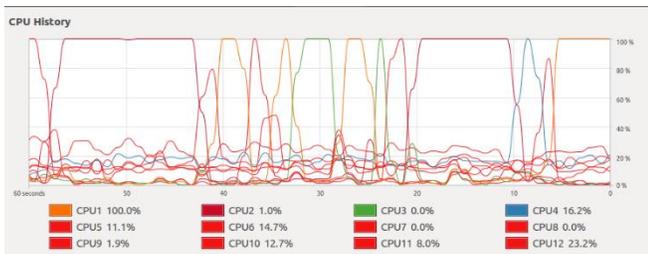


Fig. 18. CPU consumption in one minute of the RBPf using 800 particles

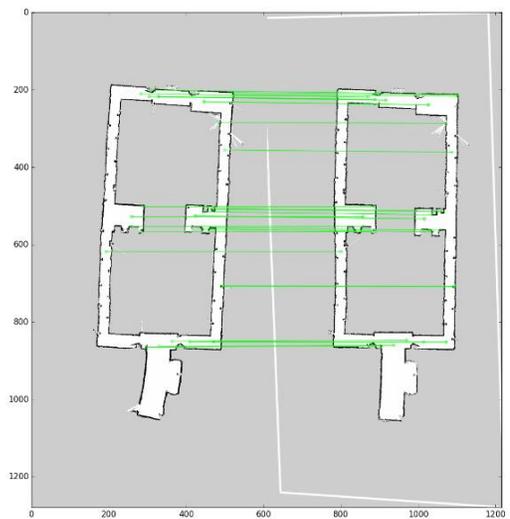


Fig. 21. 77 number of inliers points between resulting map (with ANN) and ground truth map

V. CONCLUSION

This paper presented the effects of the ANN towards the number of particles of RBPF algorithm using laser distance sensor LDS-01. An investigation has been made to observe the performance of the RBPF algorithm by using ANN and without ANN. The performance of the algorithm is calculated based on the accuracy of the map constructed by both algorithms. Based on the results, RBPF with ANN has achieved all closed loop condition of the resulting maps which is 10 times out of 10 trials by using only 30 particles compared to RBPF without ANN that needs 400 particles to achieved all closed loop condition. Furthermore, the number of inliers of the resulting maps by using RBPF with ANN is higher than RBPF without ANN consistently for the number of particles ranging from 30 to 400 particles and increase the performance by 25.17%. From the results, it can be concluded that using ANN improves the performance of the RBPF algorithm and reduces the number of particles consumption for mobile robot platform with low-cost sensors. For the future works is to test the SLAM algorithm integrated with ANN on cooperative robot. Cooperative robot SLAM has many advantages, including the ability to complete missions quicker and being resilient to the malfunction of any one of the robots. To obtain high accuracy of the cooperative SLAM, many researchers chose to adopt high-end sensor such as 2D SICK and Hokuyo Lidar as robot's perception. But this will cause staggering cost for cooperative robot system. Thus, the alternative is to implement low-cost sensors with limited sensing to perform cooperative SLAM.

ACKNOWLEDGMENT

We (the authors) would like to acknowledge and express our gratitude to Centre for Research and Innovation Management (CRIM), Centre for Telecommunication Research & Innovation (CeTRI), Fakulti Kejuruteraan Elektronik dan Kejuruteraan Komputer (FKEKK), Universiti Teknikal Malaysia Melaka (UTeM) and the Ministry of Higher Education Malaysia for the grant RACER/1/2019/TK04/UTEM//5 for providing us with the opportunity, facilities, and finances to conduct this study. We also would like to thank Mr. Hee Thien Tack, Mrs. Fatini Kamaruzaman and Mr. Mahadi Salleh for the technical support during this research.

REFERENCES

- [1] S. Wen, J. Chen, X. Lv, and Y. Tong, "Cooperative simultaneous localization and mapping algorithm based on distributed particle filter," *Int. J. Adv. Robot. Syst.*, vol. 16, no. 1, pp. 1–8, 2019.
- [2] I. Ullah, X. Su, X. Zhang, and D. Choi, "Simultaneous Localization and Mapping Based on Kalman Filter and Extended Kalman Filter," *Wirel. Commun. Mob. Comput.*, vol. 2020, 2020.
- [3] S. Saeedi, M. Trentini, H. Li, and M. Seto, "Multiple-robot Simultaneous Localization and Mapping - A Review 1 Introduction 2 Simultaneous Localization and Mapping: problem statement," *J. F. Robot.*, vol. 33, no. 1, pp. 3–46, 2016.
- [4] Y. Zhao, T. Wang, W. Qin, and X. Zhang, "Improved Rao-Blackwellised particle filter based on randomly weighted particle swarm optimization," *Comput. Electr. Eng.*, vol. 71, no. August, pp. 477–484, 2018.
- [5] N. M. Yatim, A. Jamaludin, and Z. M. Noh, "Comparison of Sampling Methods for Rao-blackwellized Particle Filter with Neural Network,"

- Symp. Intell. Manuf. Mechatronics. Symp. 2021. Lect. Notes Mech. Eng. B. Ser. (LNME)., pp. 60–75, 2021.
- [6] S. M. J. Alexandre, "MRSLAM – Multi-Robot Simultaneous Localization and Mapping," Master's Thesis, no. September, pp. 1–67, 2013.
- [7] M. P. De Melo, L. Cambuim, and E. Barros, "Occupancy Grid Map Estimation Based on Visual SLAM and Ground Segmentation," 2021 Lat. Am. Robot. Symp. 2021 Brazilian Symp. Robot. 2021 Work. Robot. Educ. LARS-SBR-WRE 2021, pp. 288–293, 2021.
- [8] R. Singh and K. S. Nagla, "Improved 2D laser grid mapping by solving mirror reflection uncertainty in SLAM," *Int. J. Intell. Unmanned Syst.*, vol. 6, no. 2, pp. 93–114, 2018.
- [9] D. Hähnel, W. Burgard, D. Fox, and S. Thrun, "An Efficient FastSLAM Algorithm for Generating Maps of Large-Scale Cyclic Environments from Raw Laser Range Measurements," *IEEE Int. Conf. Intell. Robot. Syst.*, vol. 1, no. November, pp. 206–211, 2003.
- [10] N. M. Yatim and N. Buniyamin, "Particle filter in simultaneous localization and mapping (Slam) using differential drive mobile robot," *J. Teknol.*, vol. 77, no. 20, pp. 91–97, 2015.
- [11] H. Jo, H. M. Cho, S. Jo, and E. Kim, "Efficient Grid-Based Rao-Blackwellized Particle Filter SLAM with Interparticle Map Sharing," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 2, pp. 714–724, 2018.
- [12] T. Zhang, C. Wang, Z. Yuan, and M. Zheng, "Improved grid mapping technology based on Rao-Blackwellized particle filters and the gradient descent algorithm," *Syst. Sci. Control Eng.*, vol. 7, no. 1, pp. 65–74, 2019.
- [13] H. Zhou, Z. Yao, C. Fan, S. Wang, and M. Lu, "Rao-Blackwellised particle filtering for low-cost encoder/INS/GNSS integrated vehicle navigation with wheel slipping," *IET Radar, Sonar Navig.*, vol. 13, no. 11, pp. 1890–1898, 2019.
- [14] U. Lidars, M. Torres-torriti, P. Nazate-burgos, F. Paredes-lizama, and J. Guevara, "Passive Landmark Geometry Optimization and Evaluation for Reliable Autonomous Navigation in Mining Tunnels Using 2D Lidars," *Sensors*, vol. 22, no. 8, 2022.
- [15] D. Wu, Y. Meng, K. Zhan, and F. Ma, "A LIDAR SLAM Based on Point-Line Features for Underground Mining Vehicle," *Proc. 2018 Chinese Autom. Congr. CAC 2018*, pp. 2879–2883, 2019.
- [16] G. S. Martins, D. Portugal, and R. P. Rocha, "mrgs: A Multi-Robot SLAM Framework for ROS with Efficient Information Sharing," *Robot Oper. Syst. (ROS), Stud. Comput. Intell.*, vol. 895, pp. 45–75, 2021.
- [17] F. Demim, A. Nemra, K. Louadj, M. Hamerlain, and A. Bazoula, "Cooperative SLAM for multiple UGVs navigation using SVSF filter," *Automatika*, vol. 58, no. 1, pp. 119–129, 2017.
- [18] S. Saeedi, L. Paull, M. Trentini, M. Seto, and H. Li, "Map merging for multiple robots using Hough peak matching," *Rob. Auton. Syst.*, vol. 62, no. 10, pp. 1408–1424, 2014.
- [19] N. M. Yatim and N. Buniyamin, "Development of Rao-Blackwellized Particle Filter (RBPF) SLAM Algorithm Using Low Proximity Infrared Sensors," 9th Int. Conf. Robot. Vision, Signal Process. Power Appl., no. September 2017, pp. 395–405, 2017.
- [20] G.-G. Kim, Heon-hui Ha, Yun-Su Jin, "A Study on the Environmental Map Building for a Mobile Robot Using Infrared Range-finder Sensors," *Intell. Robot. Syst. 2003. (IROS 2003). Proceedings. 2003 IEEE/RSJ Int. Conf.*, vol. 1, no. October 2003, pp. 711–716, 2003.
- [21] Y. Dai and M. Zhao, "Grey Wolf Resampling-Based Rao-Blackwellized Particle Filter for Mobile Robot Simultaneous Localization and Mapping," *J. Robot.*, vol. 2021, 2021.
- [22] G. Vallicrosa and P. Ridao, "H-SLAM: Rao-Blackwellized particle filter SLAM using Hilbert Maps," *Sensors (Switzerland)*, vol. 18, no. 5, 2018.
- [23] A. S. Aguiar, F. N. dos Santos, H. Sobreira, J. B. Cunha, and A. J. Sousa, "Particle filter refinement based on clustering procedures for high-dimensional localization and mapping systems," *Rob. Auton. Syst.*, vol. 137, p. 103725, 2021.
- [24] N. Akai, L. Y. Morales, and H. Murase, "Simultaneous pose and reliability estimation using convolutional neural network and Rao-Blackwellized particle filter," *Adv. Robot.*, vol. 32, no. 17, pp. 930–944, 2018.

- [25] S. Thrun, D. Fox, W. Burgard, and F. Dellaert, "Robust Monte Carlo localization for mobile robots," *Artif. Intell.*, vol. 128, no. 1–2, pp. 99–141, 2001.
- [26] C. S. Gurel, "Real-Time 2D and 3D Slam Using Rtab-Map , Gmapping , and Cartographer Packages," no. August, 2018.
- [27] M. Islam, T. Ahmed, A. T. Bin Nuruddin, M. Islam, and S. Siddique, "Autonomous Intruder Detection Using a ROS-Based Multi-Robot System Equipped with 2D-LiDAR Sensors," 2020 IEEE Int. Symp. Safety, Secur. Rescue Robot. SSR 2020, pp. 326–333, 2020.
- [28] B. Hampton, A. Al-hourani, B. Ristic, and B. Moran, "RFS-SLAM Robot: An Experimental Platform for RFS Based Occupancy-Grid SLAM," 2017 20th Int. Conf. Inf. Fusion.
- [29] Y. Alborzi, B. S. Jalal, and E. Najafi, "ROS-based SLAM and Navigation for a Gazebo-Simulated Autonomous Quadrotor," 2020 21st Int. Conf. Res. Educ. Mechatronics, REM 2020, 2020.
- [30] R. Yagfarov, M. Ivanou, and I. Afanasyev, "Map Comparison of Lidar-based 2D SLAM Algorithms Using Precise Ground Truth," 2018 15th Int. Conf. Control. Autom. Robot. Vision, ICARCV 2018, pp. 1979–1983, 2018.
- [31] K. Kamarudin, S. M. Mamduh, A. Y. Md Shakaff, and A. Zakaria, "Performance analysis of the microsoft kinect sensor for 2D simultaneous localization and mapping (SLAM) techniques," *Sensors (Switzerland)*, vol. 14, no. 12, pp. 23365–23387, 2014.
- [32] M. Rojas-Fernandez, D. Mujica-Vargas, M. Matuz-Cruz, and D. Lopez-Borreguero, "Performance comparison of 2D SLAM techniques available in ROS using a differential drive robot," 2018 28th Int. Conf. Electron. Commun. Comput. CONIELECOMP 2018, vol. 2018-Janua, pp. 50–58, 2018.
- [33] N. A. Yatim, N. Buniyamin, Z. M. Noh, and N. A. Othman, "Occupancy grid map algorithm with neural network using array of infrared sensors," *J. Phys. Conf. Ser.*, vol. 1502, no. 1, 2020.
- [34] S. Mammadli, "Financial time series prediction using artificial neural network based on Levenberg-Marquardt algorithm," *Procedia Comput. Sci.*, vol. 120, pp. 602–607, 2017.
- [35] M. I. Ibrahimy, M. R. Ahsan, and O. O. Khalifa, "Design and optimization of levenberg-marquardt based neural network classifier for EMG signals to identify hand motions," *Meas. Sci. Rev.*, vol. 13, no. 3, pp. 142–151, 2013.
- [36] H. Ullah et al., "Levenberg-Marquardt Backpropagation for Numerical Treatment of Micropolar Flow in a Porous Channel with Mass Injection," *Complexity*, vol. 2021, 2021.
- [37] P. Malik, A. Gehlot, R. Singh, L. R. Gupta, and A. K. Thakur, "A Review on ANN Based Model for Solar Radiation and Wind Speed Prediction with Real-Time Data," *Arch. Comput. Methods Eng.*, no. January, 2022.