IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 9 August 2024, accepted 11 September 2024, date of publication 19 September 2024, date of current version 1 October 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3463876

RESEARCH ARTICLE

Door-Density-Aware Path Planning

CHARAN VIKRAM[1], M. K. DOJITHA M. MIRIHAGALLA[1,2], MATTHEW S. K. YEO[1], ZIMOU ZENG[1], CHARAN SATYA CHANDRA SAIRAM BORUSU[1], M. A. VIRAJ J. MUTHUGALA[1], (Senior Member, IEEE), AND MOHAN RAJESH ELARA[1]

[1]Engineering Product Development Pillar, Singapore University of Technology and Design, Singapore 487372
[2]Department of Electronic and Telecommunication Engineering, University of Moratuwa, Moratuwa 10400, Sri Lanka

Corresponding author: M. A. Viraj J. Muthugala (viraj_jagathpriya@sutd.edu.sg)

This research is supported by the National Robotics Programme under its National Robotics Programme (NRP) BAU, Ermine III: Deployable Reconfigurable Robots, Award No. M22NBK0054 and also supported by A*STAR under its "RIE2025 IAF-PP Advanced ROS2-native Platform Technologies for Cross-sectorial Robotics Adoption (M21K1a0104)" programme.

ABSTRACT Doors are part of the building infrastructure that mobile robots have to pass through to reach zones on the other side. If robots were to clear these obstacles, they would require human assistance, advanced end-effectors, and complex control systems, making it challenging for robots. Therefore, a robot deployed in an environment should be capable of minimizing the passing through doors as well as path distance to improve overall efficiency. This paper proposes a novel Door-Density-Aware (DDA) path planning method. A vision-based door-detecting framework based on YOLOv8 has been developed to tag the door locations in a robot's navigation map. The proposed DDA path planner uses a door-tagged map to plan an efficient path considering the cost of moving through doors and the path distance. Genetic Algorithm (GA) and Gray Wolf Optimization (GWO) have been considered for solving this optimization problem. According to the experimental results, the proposed method can effectively detect and tag doors in the navigation map and plan efficient paths. In summary, the proposed DDA path planner with GA outperformed other approaches, achieving cost reductions of 66%, 34%, 49%, and 60% compared to random selection, DDA with GWO, GA minimizing only distance, and GWO minimizing only distance, respectively.

INDEX TERMS Door density, path planning, optimization, vision-based detection.

I. INTRODUCTION

Mobile service robots are being implemented in service and maintenance industries to mitigate the increasing shortage and rising costs of trained manpower [1]. Menial and repetitive tasks such as cleaning, inspection maintenance and elderly care have been delegated to mobile service robots over the years with the improvements in robotic technology and the need of quick responses for such tasks [2].

With mobile service robots, navigation is a crucial feature for the above applications and work tasks. Robots would work best when they can have accessible space within their work environment [3]. However, most of the existing built environments do not consider robotic deployments in their design. The doors that separate rooms, while a trivial infrastructure for people to open and access the other side, are a pain point for robotic deployments [4]. This is especially evident when the robots do not have the requisite end-effectors or equipment to be able to open doors [5], or would require human aid to open doors, which prevents fully autonomous work.

The typical method would be to implement such manipulators onto the robot, or to upgrade the entire building infrastructure to a smart building for the service robots to wirelessly communicate and open the doors. These methods are often costly and unfeasible, especially if the service robot is small in footprint, or do not have the advanced systems for deftly controlling end-effectors needed for opening doors [6].

The 'Design for Robot' (DfR) methodology is a growing field to create environments more conducive to robot operation by changing architectural elements to enhance robot inclusivity [7]. DfR employs a multidisciplinary approach, combining robotics, architecture, and human-robot research application to design spaces that accommodate for both

The associate editor coordinating the review of this manuscript and approving it for publication was Nikhil Padhi.

humans and robots [8], [9], [10]. In addition, DfR approaches have been utilized to assessing the robot-inclusivity of building infrastructure. The work [11] proposed an automated robotic system to evaluate the robot-inclusivity of an environment based on the lighting condition which effects the performance of a robot. There has also been research conducted on automated door detection for indoor environments [12], [13], [14]. However, a notable research gap exists in assessing how these artificial barriers of doors and gated entryways would impact robot inclusivity, and no autonomous system has been developed yet to quantify such a factor in building infrastructure for determining how fit the environment is for deploying the mobile service robots.

On the other hand, path planning in robotics frequently employs algorithms like A* and Dijkstra for point-to-point navigation [15]. These methods can find efficient paths while avoiding the obstacles during point-to-point navigation. The traditional path planning methods often aim to minimize travel energy and distance, as well as reduce the number of turns [16], [17]. The Traveling Salesman Problem (TSP) is a well-known approach used to minimize energy consumption when covering a set of waypoints [18]. By solving the TSP, a robot can determine the optimal sequence for navigating through multiple waypoints minimizing the energy. Metaheuristics, such as Genetic Algorithms (GA) and ant colony optimization, are often utilized to address the complexities of TSP, providing efficient solutions for waypoint navigation and energy optimization in robotic path planning [19], [20]. However, path planning that takes into account the density of doors within an environment has not been explored until now.

This paper proposes a novel Door-Density-Aware (DDA) path planning method. The proposed approach integrates door density into the path planning process, potentially leading to more efficient and contextually appropriate navigation strategies for robots in environments with varying door densities. The major scientific contributions of this paper include the development of:

- A method to autonomously generate door-density maps of building infrastructure for navigation.
- A Door-Density-Aware (DDA) path planning approach based on metaheuristics.

Section II delves into the methods used for automating door-density mapping. Section III presents the proposed DDA path planning approach. Section IV details the experiment used for validating the proposed system. Section V then concludes the paper, along with future works to be conducted for this research work.

II. DOOR-DENSITY MAP GENERATION

A. ROBOT PLATFORM

The Meerkat audit robot platform was designed to conduct mapping and auditing of sites using the DfR and robot inclusive parameters. The robot is shown in Fig. 1. The robot supports multiple sensors and has attachment points for additional sensor types to be mounted for other parameter tracking purposes. The typical sensor setup involves a SICK 2D LiDAR, VectorNav Inertial Measurement Unit (IMU), and a RealSense camera. An onboard computer is used for processing. This enables the robot to perform auditing operations for given spatial sites, and data collection and storage, along with analysis for the various robot inclusivity parameters.

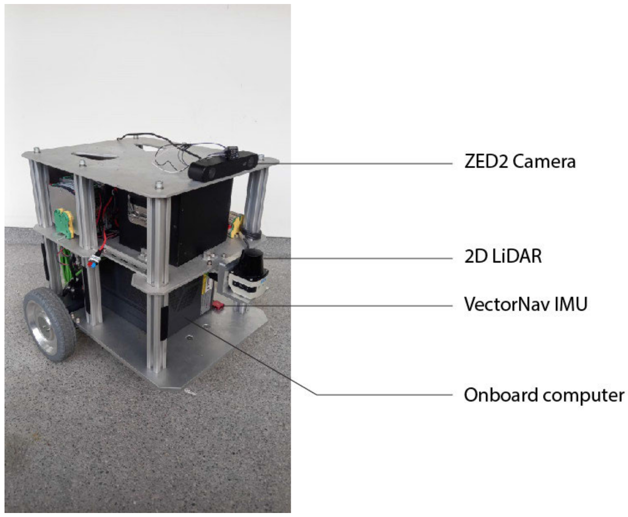


FIGURE 1. Meerkat audit robot and its main sensors.

B. DOOR DETECTION FRAMEWORK

In this research, we focus on the detection of door handles as a proxy for detection, rather than entire doors due to the limited field of view available to robotic visual systems when the robots are close to the doors. If the robot attempts to detect an object from far away, the accuracy of locating the object would reduce. Typically, only the door handle is visible to the robot when it is close to them (see Fig. 2). By accurately identifying door handles, the robot can effectively infer the locations of doors within the environment. In addition to that, use of door handles as a proxy for doors enables the identification of already opened doors. Furthermore, a rough estimation of the door location is sufficient for the path planning strategy proposed in this paper.

Many vision-based artificial intelligence detection models have been developed to cater to the demand in diverse applications. Faster Region-based Convolutional Network (Faster R-CNN) [21], Single-Shot Detector (SSD) [22], and You Only Look Once (YOLO) [23] are some examples. YOLO, particularly its eighth iteration (YOLOv8), stands out due to its remarkable speed and efficiency without compromising accuracy [24]. Its lightweight architecture and fast inference time make it ideal for integration into robotic systems, enabling effective object detection and tracking. Thus, YOLOv8 is highly suitable for detecting doors in our work.

In order to enhance feature extraction, YOLOv8 has a new backbone network that uses cutting-edge methods including Cross-Stage Partial (CSP) connections to strengthen gradient flow and lower computing complexity. YOLOv8's head

FIGURE 2. Using door handle as a proxy for locating door location.

uses anchor-free detection to expedite detection and reduce computing cost, while the neck integrates Path Aggregation Network (PANet) for improved spatial information retention and multi-scale feature fusion.

C. GENERATING DOOR-TAGGED MAPS

The overall robot system architecture of the framework proposed for generating door-tagged maps is shown in Fig. 3. In the Meerkat's system, the LiDAR sensor scans the environment and stores the data using the 'rplidar' ROS package. Another ROS SLAM algorithm, 'gmapping' would use those scans along with IMU data to build a map while simultaneously tracking the robot's location within it. Another ROS package, 'AMCL' then uses this pre-built map and new lidar scans to estimate the robot's pose (position and orientation) with a particle filter, effectively localizing the robot within the environment. This combined approach allows the robot to navigate in an unknown environment. The 'rviz' ROS package can be separately used for visualization of the mapped zone.

A ZED2 depth camera was used for the detection of door handles. Depth information was used for gauging the distance from robot to door for tagging the door's location on the map. Upon detecting a door through the ZED2 camera up to a maximum distance of 1.2 meters from the robot, the depth gauging algorithm will trigger to gauge the current distance of the robot to the detected door, and the door's apparent location will be saved as coordinate points in the xy-plane with reference to the stored map of the site. These coordinates are saved under a separate file for the next step of door location tagging.

By using the coordinates of detected doors from the ZED2 camera, the doors would then be tagged upon the base map of the site. These location tags are then overlaid on the base map to reflect the positions of doors within the work environment. To avoid the tagging of the same door several time, the doors detected in close proximity is filtered out. This filtering was done by obtaining the centroid of the multiple locations of doors that overlap or are in contact with each other, and using that as the new center point. The formula for obtaining the aggregated centroid is given in (1), where $p_{x,y}$ is the center, and (x_k, y_k) is the center of k^{th} detection among all M number of detection of the same door.

$$p_{x,y} = \frac{\sum_{k=1}^{M} (x_k, y_k)}{M} \tag{1}$$

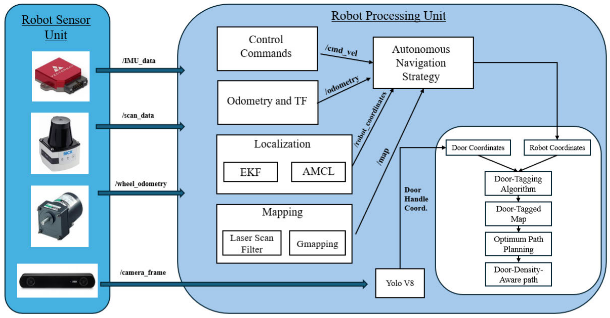


FIGURE 3. Overall robot architecture for the framework proposed for generating door-tagged maps.

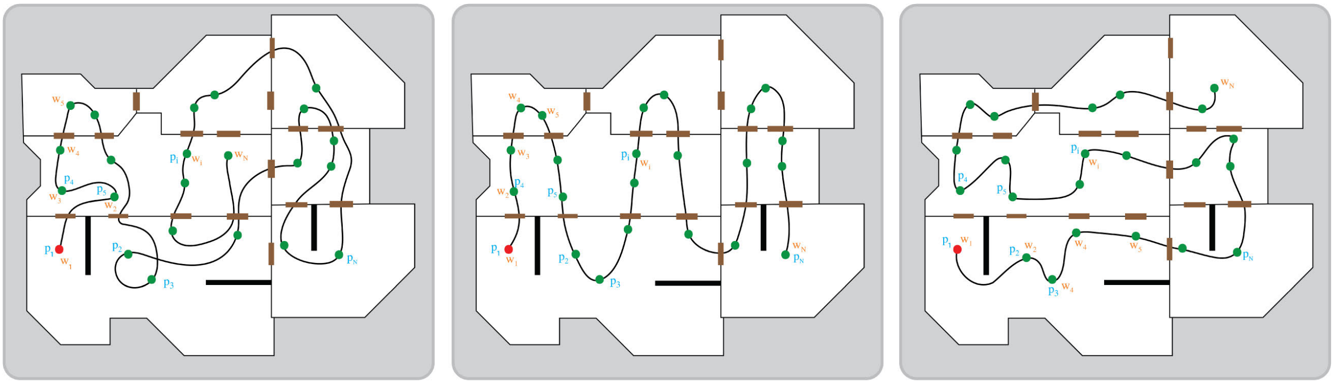
The overall process of generating door-tagged maps is explained in Algorithm 1.

III. DOOR-DENSITY-AWARE (DDA) PATH PLANNING

When performing activities such as inspections and patrolling by a robot, the robot often need to go to a set of designated goal points. All the designed goal points (p_i) should be covered when performing activities in the environment. These locations are referred to as waypoints (w_j). Three possible paths are depicted in Fig. 4 for an example scenario with N number of goal points to be visited. Fig. 4(a) shows a non-optimized path where the sequence of navigation is randomly selected. Fig. 4(b) shows an optimized path that passes through 11 doors. Fig. 4(c) shows an optimal path that passes through only 4 doors. Considering the energy usage of robots, the path in (c) is preferable.

Algorithm 1 Algorithm for Door Tagging

1) Initialize Mapping
   - Initialize 2D LiDAR
   - Perform site mapping
   - Save LiDAR map
2) Initialize Door detection run
   - Initialize door detection algorithm and dataset
   - If door is detected AND within 1.2 meters from robot's current location:
     – Create separate layer overlaid on site map with same origin
     – Tag location on separate layer
   - Else Standby for next detection
   - Save final location map
3) Combining maps
   - Align site map and location map
   - Remove door location aggregates for tagged door locations in close proximity (overlapping circles)
   - Replace door location tag aggregates with combined tag
4) Overlay final maps



(a) (b) (c)

FIGURE 4. Potential waypoint sequences for the auditing process. (a): Randomly selected sequence, (b): Shortest path distance, and (c): Path distance minimum and number of door passing minimum.

The optimal sequence of waypoints is represented as W_k, where k ranges from 1 to N, with N being the total number of waypoints, including the starting point. The traditional Traveling Salesman Problem (TSP) method uses Euclidean distances. However, in a typical environment, there are many obstacles. It is necessary to plan a path that avoids these obstacles. To achieve this, the A* (A-star) algorithm is used. These A* distances are then used instead of the Euclidean distances, unlike in the traditional TSP problem.

Assume $D(k, l)$ is the distance calculated using the A-star algorithm from point k to point l. The energy consumed by the robot for navigation is directly proportional to the distance it travels, assuming a flat, doorless workspace. The energy required for the movement of the robot can be expressed as: $E_{k,l} = MD(k, l)$, where M is a proportional constant.

If the robot needs to go through a door, additional overhead is required in terms of requiring human support or using a manipulator to open the door. This requirement should penalize the energy cost for a path that connects two locations through a door. In our solution, a penalty of E_0 is applied when the robot passes through a door. The energy cost should be updated as follows in (2).

$$\hat{E}_{k,l} = \begin{cases} MD(k, l) + nE_0 & \text{if path goes through } n \text{ doors} \\ MD(k, l) & \end{cases} \tag{2}$$

where E_0 is the additional effort required for passing through a door. Thus, the total energy required is calculated as in (3). The required additional effort is considered as 10 times than the average distance between waypoints in a particular case.

$$\text{Total cost, } C = \sum_{k=1}^{k=N-1} \hat{E}_{k,k+1} \tag{3}$$

It is now required to find an optimal solution that minimizes the above expression. Given N waypoints, there are $\frac{N(N-1)}{2}$ waypoint pairs. Calculating the energy usage between each pair and finding the optimal solution requires $(N-1)!$ comparisons, which becomes impractically large for large N. Therefore, it is essential to focus on metaheuritics optimization techniques. We selected the metaheuritics, Genetic Algorithm (GA) and Grey Wolf Optimizer (GWO) to solve this optimization problem due to their proven efficacy in solving complex optimization problems [25]. GA is renowned for its robust search capabilities and ability to explore a large solution space through mechanisms inspired by natural selection and genetics [26]. GWO, on the other hand, mimics the social hierarchy and hunting behavior of grey wolves, offering a strong balance between exploration and exploitation, which is crucial for finding optimal solutions [27].

A. GENETIC ALGORITHM (GA)

A Genetic Algorithm (GA) is inspired by natural selection, where the best individuals survive and reproduce based on principles of genetics and evolution [26]. This technique employs a population of individuals (representing waypoint sequences) working collectively to find the optimal solution for through genetic processes. The flow of the GA is outlined in Algorithm 2. The population size and the number of generations were selected as 100 and 2000, respectively observing the performance variations.

B. GRAY WOLF OPTIMIZATION (GWO)

Gray Wolf Optimization (GWO) is inspired by the social hierarchy and hunting strategy of gray wolves in nature. This technique uses a population of individuals (representing waypoint sequences) working together to find the optimal solution by demonstrating the cooperative behavior and leadership hierarchy of gray wolves [27]. The flow of the GWO is outlined in Algorithm 3. The population size and the number of generations were selected as 40 and 5000, respectively observing the performance variations.

IV. EXPERIMENTS

A. TRAINING AND TESTING OF DOOR DETETCION MODELS

A dataset comprising 933 images was used in this work, and basic augmentations were applied to expand the dataset.

Algorithm 2 Pseudocode for the GA

1) Initialize Parameters:
   - Generations
   - Population size
   - CrossOverRate = 0.5
2) Generate Population:
   - Generate initial population of random waypoint connections.
   - Sort the population by fitness scores.
3) Evolutionary Process:
   - For each generation:
     - Parent Selection: Select top individuals as parents and split into mothers and fathers.
     - Crossover: Combine parent segments to create offspring.
     - Mutation: Randomly swap two waypoints in offspring.
     - Population Replacement: Merge old and new populations, sort by fitness, and retain the top individuals.
4) Return Best Solution:
   - Output the best route for the waypoint sequence after all generations.

Algorithm 3 Pseudocode for the GWO

1) Initialize parameters:
   - MaxIterations
   - SearchAgents
   - lb = 0
   - ub = Number of waypoints - 1
2) Initialize Alpha, Beta, Delta scores to infinity
3) Initialize positions of SearchAgents randomly within boundaries [lb, ub]
4) For each iteration from 1 to MaxIterations:
   - For each SearchAgent:
     - Ensure the position is within [lb, ub]
     - Calculate fitness
     - Update Alpha, Beta, Delta based on fitness:
       * If fitness < Alpha_score: Update Alpha
       * Else if fitness < Beta_score: Update Beta
       * Else if fitness < Delta_score: Update Delta
5) Calculate the parameter 'a' as $2 - (\frac{\text{current iteration} \times 2}{\text{MaxIterations}})$
6) For each SearchAgent:
   - For each dimension (waypoint):
     - Update position based on Alpha, Beta, Delta positions
7) Return the best solution.
   - Output the best route for the waypoint sequence after all Iterations

These augmentations included horizontal flipping, 90-degree rotations, blurring, and the introduction of Gaussian noise with a standard deviation of 0.5. As a result, the dataset was inflated to 2,195 images. This augmented dataset was then split into training, testing, and validation sets, with proportions of 70 %, 15%, and 15%, respectively. The model training was carried out using an NVIDIA GeForce RTX 3080 GPU.

A performance comparison of the trained YOLOv8 model with other models, YOLOv5, SSD, and Faster-RCNN is given in Table 1. All the models were trained using the same dataset and the standard performance matrices precision, recall, F1-score, and mean Average Precision (mAP) have been used here. This comparison of trained detection models shows that YOLOv8 is the best choice for our application. With a precision of 0.83, recall of 0.78, F1 score of 0.82, and mean Average Precision (mAP) at 0.5 of 0.86, YOLOv8 outperforms other models across all metrics.

TABLE 1. Performance comparison of trained detection models.

Model	Precision	Recall	F1 Score	mAP@0.5
SSD	0.75	0.70	0.72	0.70
Faster-RCNN	0.78	0.74	0.76	0.78
YOLOv5	0.80	0.75	0.77	0.82
YOLOv8	0.83	0.78	0.82	0.86

FIGURE 5. Examples for YOLOv8 detection results from testing dataset.

Some detection results for the YOLOv8 model can be seen in Fig. 5 as a way to show the detection capabilities of the trained model.

B. GENERATING DOOR-TAGGED MAP

The Meerkat robot was made to pass through the test sites with the door detecting algorithm to obtain the door locations for the respective sites.

The first test was conducted in a classroom and its adjacent storage rooms. Fig. 6(a) shows a map of the environment

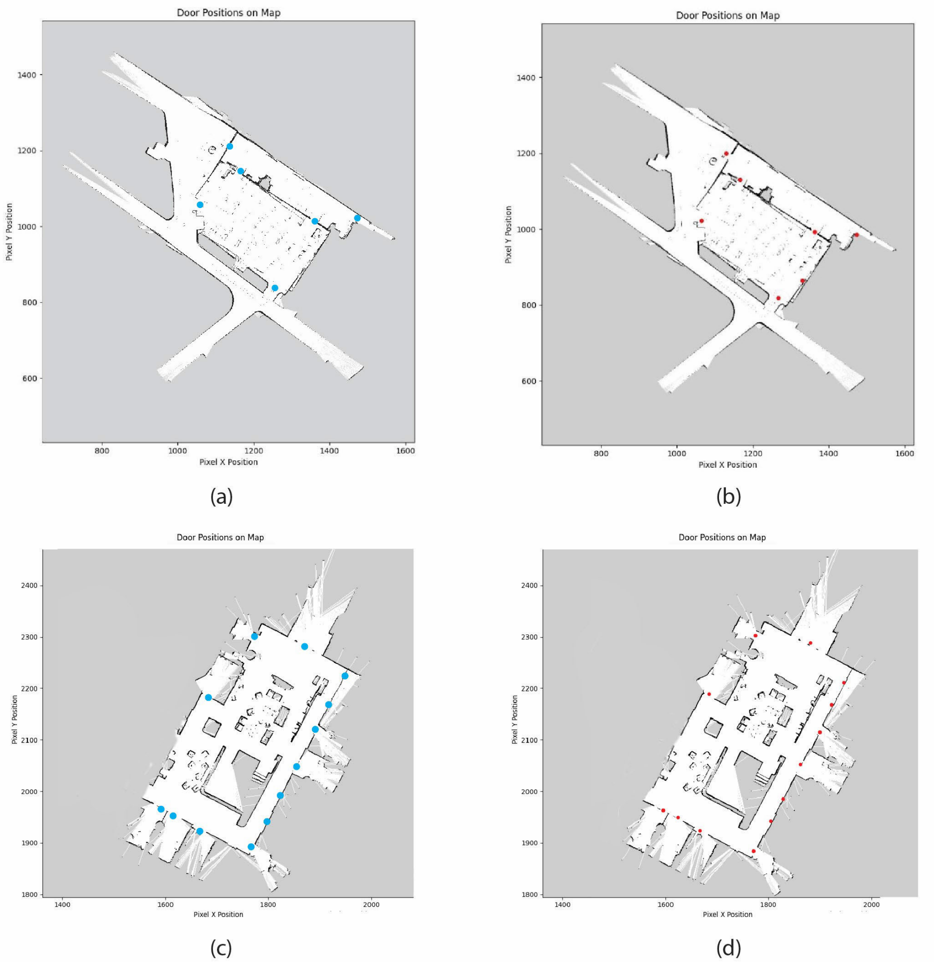


FIGURE 6. Site maps and resultant door-tagged maps. (a) manually created door layout of test site 1, (b): generated door-tagged map for test Site 1, (c): manually created door layout of test site 2, (d) generated door-tagged map for test Site 2. Blue dots on site maps indicate actual locations of doors for the given sites while red dots indicated the doors located by the robot. Map resolution is 1 pixel = 0.05 m.

with the door locations (manually annotated as the ground truth for the comparison). There were six doors on this site. The generated door-tagged map for the first site is shown in Fig. 6(b). This door-tagged map has accurately tagged all six doors. However, the robot erroneously detected and tagged one extra door on the map. Even though this extra erroneous detection would not considerably impede the path planning performance since no accessible space is recorded beyond the erroneously detected door.

The second test site is in one of the faculty office. The site contains interior meeting rooms in the center that also provide a shortcut between the corridors that connect the peripheral office rooms when the meeting rooms are not being used (see Fig. 6(c)). The generated door-tagged map of the area is shown in Fig. 6(d). All the doors were tagged successfully at the correct locations in this site.

The door-tagged map generated for the test site 1 and 2 validate that the proposed system can successfully generate the door-tagged map. The map generation is thus adequate for door-density-aware path planning purposes.

C. PATH PLANNING

The door-tagged maps of two sites were used for the DDA path planning. For benchmarking the path planning performance, the navigation of the robot in a random sequence and path planning that only considers minimizing the distance (using both GA and GWO) have been considered as baselines. The paths generated for each case are given in Fig. 7. The corresponding cost variations during the optimization processes are given in Fig. 8. The key parameters of the generated paths, distance (D), number of doors to be crossed (n), total path cost (C), and time for optimization (t) for the cases are summarized in Table 2 for comparison.

The robot navigation through 25 waypoints was considered for site 1. The path generated for site 1 considering a random sequnce is given in Fig. 7(a). The distance of this path is 143.55 m and it crosses 8 doors leading to total path cost of 94355 (see Table 2). The proposed DDA path planning approach considering GA was able to generate a path with distance of 97.00 m and the number of doors to be crossed 2 leading to a substantially lower total path

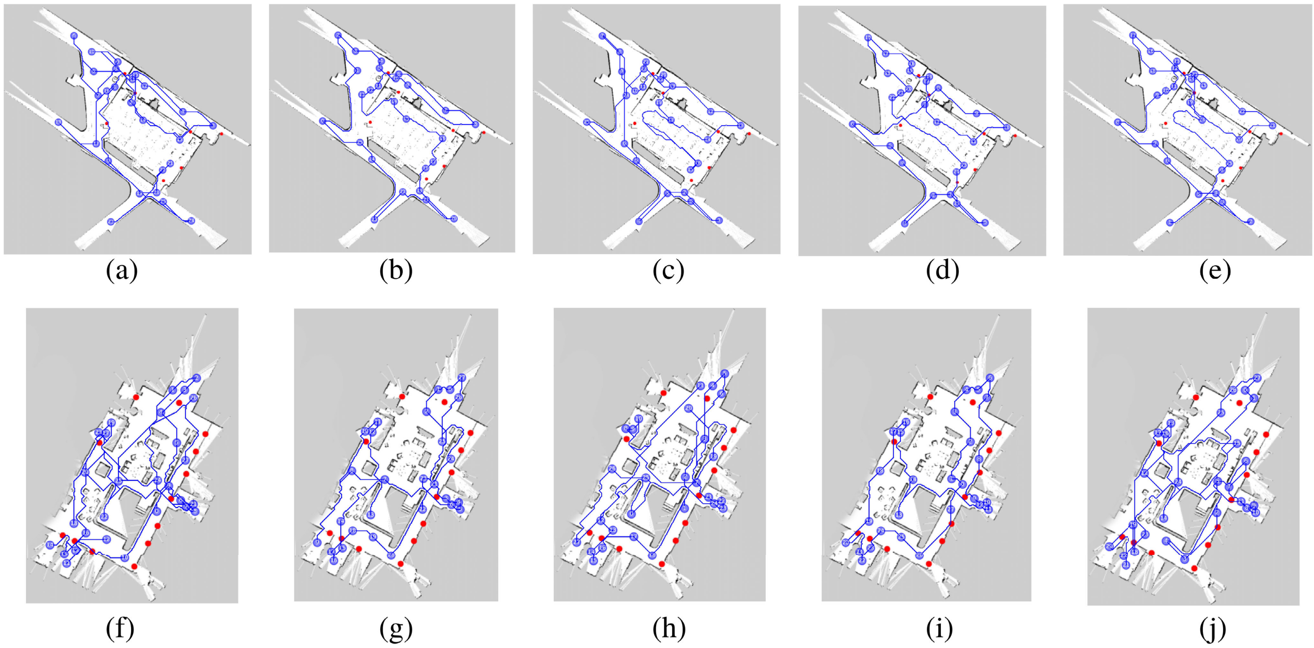


FIGURE 7. Planed paths for the two sites considering different approaches. (a): Random waypoints sequence in site 1, (b): DDA path planning using GA in site 1, (c): DDA path planning using GWO in site 1, (d): Distance-only optimization considering GA in site 1, (e): Distance-only optimization considering GWO in site 1, (f): Random waypoints sequence in site 2, (g): DDA path planning using GA in site 2, (h): DDA path planning using GWO in site 2, (i): Distance-only optimization considering GA in site 2, and (j): Distance-only optimization considering GWO in site 2.

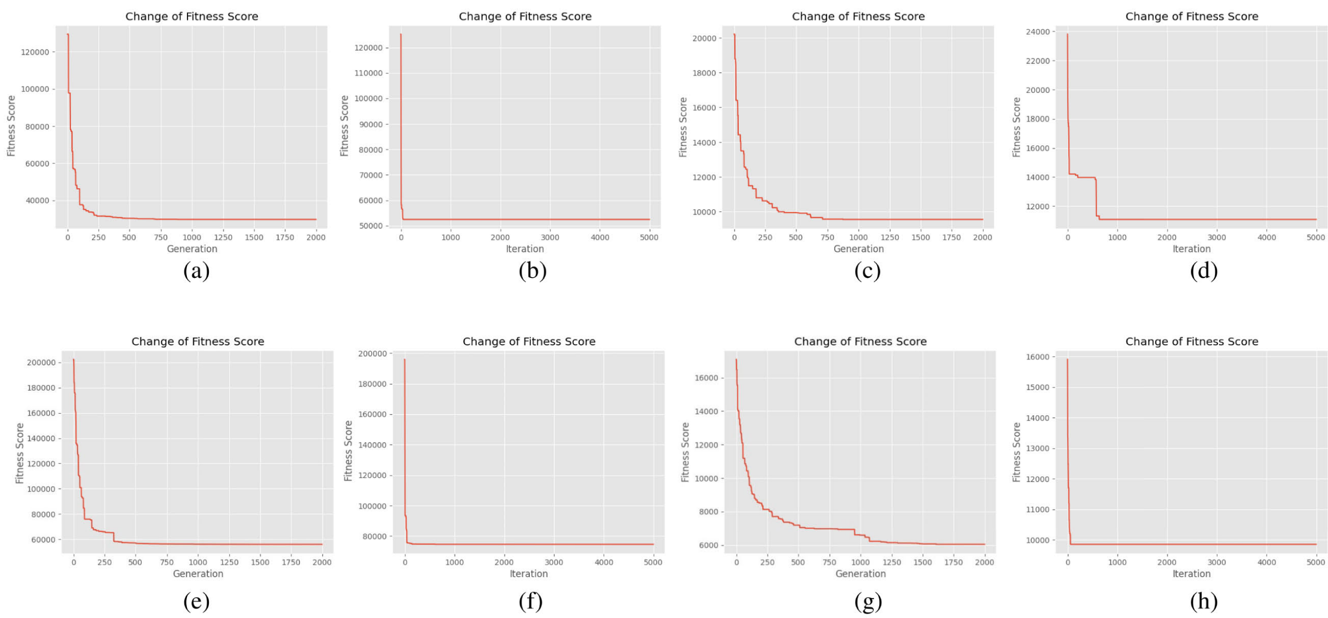


FIGURE 8. Variation of cost with generation for each case. (a): DDA path planning using GA in site 1, (b): DDA path planning using GWO in site 1, (c): Distance-only optimization considering GA in site 1, (d): Distance-only optimization considering GWO in site 1, (e) DDA path planning using GA in site 2, (f) DDA path planning using GWO in site 2, (g): Distance-only optimization considering GA in site 2, and (h): Distance-only optimization considering GWO in site 2.

cost ($C = 29700$) compared to the other cases. The GA took 6.7 minutes for terminating the optimization process in this case. In contrast, DDA path planing using GWO could generate a path that is considerably higher than the GA. In the event of minimizing only the distance, the GA could generate the shortest path of 95.53 m. However, 7 doors has to be crossed by the robot leading to substantially higher total path cost ($C = 79553$).

Similar performance was observed for site 2, where 27 waypoints were considered. Overall, the proposed DDA path planning approach with GA generated the path with the lowest total path cost compared to the other cases in both sites. The proposed DDA with GA achieved average cost reductions of 66%, 34%, 49%, and 60% compared to random selection, DDA with GWO, GA minimizing only distance, and GWO minimizing only distance, respectively.

TABLE 2. Comparison of key parameters for different path planning approaches.

Site	Random			Door-Density-Aware (DDA) path planning								Without considering door density							
				GA				GWO				GA				GWO			
	D (m)	n	C	D (m)	n	C	t (min)	D (m)	n	C	t (min)	D (m)	n	C	t (min)	D (m)	n	C	t (min)
1	143.55	8	94355	97.00	2	29700	6.7	124.62	4	52462	47.9	95.53	7	79553	6.4	110.83	7	81082	53.2
2	124.45	18	156445	90.45	6	56044	6.9	105.98	8	74597	60.0	60.45	10	86045	7.2	98.56	15	129855	55.6

Furthermore, the processing time is satisfactory. Therefore, the proposed DDA with GA is suitable for improving navigation performance of robots used in waypoint coverage applications such as patrolling and inspections.

V. CONCLUSION

This paper proposed a novel path-planning approach that considers door density in an environment. The proposed approach consists of a method to autonomously create door-density maps of building infrastructure through vision-based detection. The vision-based door detection method has been developed using the YOLOv8 model. The proposed DDA path planner utilizes these door-tagged maps to generate an efficient navigation path that minimizes the cost of path distance and the effort required for going through doors. A GA and a GWO have been proposed to find the optimum waypoint sequence during navigation.

The experimental results found that the trained YOLOv8 model performs superiorly to other detection models, SSD, Faster-RCNN, and YOLOv5. In addition, the proposed robotic system can autonomously generate door-tagged maps of building infrastructure with an accuracy adequate for path planning. The proposed DDA path-planning approaches generated paths that minimize the total path costs compared to random waypoint sequence selections and distance-minimization-only path planning. The GA produces the best results compared to GWO in DDA path planning. Therefore, the work proposed in this paper would be highly beneficial for efficient path planning of robots deployed in indoor building infrastructures. Future work would explore the consideration of other factors that impact robot inclusivity, such as surface unevenness, for efficient path planning.

REFERENCES

[1] A. Manthiou, P. Klaus, V. G. Kuppelwieser, and W. Reeves, "Man vs machine: Examining the three themes of service robotics in tourism and hospitality," *Electron. Markets*, vol. 31, no. 3, pp. 511–527, Sep. 2021.

[2] I.-M. Chen, E. Asadi, J. Nie, R.-J. Yan, W. C. Law, E. Kayacan, S. H. Yeo, K. H. Low, G. Seet, and R. Tiong, "Innovations in infrastructure service robots," in *Proc. 21st CISM-IFToMM Symp.*, Udine, Italy. Switzerland: Springer, Jun. 2016, pp. 3–16.

[3] M. A. V. J. Muthugala, S. M. B. P. Samarakoon, and M. R. Elara, "Design by robot: A human–robot collaborative framework for improving productivity of a floor cleaning robot," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2022, pp. 7444–7450.

[4] M. Hersh, "Overcoming barriers and increasing independence—Service robots for elderly and disabled people," *Int. J. Adv. Robotic Syst.*, vol. 12, no. 8, p. 114, Aug. 2015.

[5] J. Saenz, N. Elkmann, O. Gibaru, and P. Neto, "Survey of methods for design of collaborative robotics applications- why safety is a barrier to more widespread robotics uptake," in *Proc. 4th Int. Conf. Mechatronics Robot. Eng.*, Feb. 2018, pp. 95–101.

[6] Y. Dai, C. Xiang, W. Qu, and Q. Zhang, "A review of end-effector research based on compliance control," *Machines*, vol. 10, no. 2, p. 100, Jan. 2022.

[7] R. E. Mohan, N. Tan, K. Tjoelsen, and R. Sosa, "Designing the robot inclusive space challenge," *Digit. Commun. Netw.*, vol. 1, no. 4, pp. 267–274, Nov. 2015.

[8] E. B. Sandoval, R. Sosa, and M. Montiel, "Robot-ergonomics: A proposal for a framework in HRI," in *Proc. Companion ACM/IEEE Int. Conf. Human-Robot Interact.*, Mar. 2018, pp. 233–234.

[9] M. S. K. Yeo, S. M. B. P. Samarakoon, Q. B. Ng, M. A. V. J. Muthugala, and M. R. Elara, "Design of robot-inclusive vertical green landscape," *Buildings*, vol. 11, no. 5, p. 203, May 2021.

[10] G. B. Verne, "Adapting to a robot: Adapting gardening and the garden to fit a robot lawn mower," in *Proc. Companion ACM/IEEE Int. Conf. Human-Robot Interact.*, Mar. 2020, pp. 34–42.

[11] Z. Zeng, M. S. K. Yeo, C. S. C. S. Borusu, M. A. V. J. Muthugala, M. Budig, M. R. Elara, and Y. Wang, "A framework for auditing robot-inclusivity of indoor environments based on lighting condition," *Buildings*, vol. 14, no. 4, p. 1110, Apr. 2024.

[12] X. Yang and Y. Tian, "Robust door detection in unfamiliar environments by combining edge and corner features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 57–64.

[13] S. Meyer Zu Borgsen, M. Schöpfer, L. Ziegler, and S. Wachsmuth, "Automated door detection with a 3D-sensor," in *Proc. Can. Conf. Comput. Robot Vis.*, May 2014, pp. 276–282.

[14] B. Quintana, S. A. Prieto, A. Adán, and F. Bosché, "Door detection in 3D coloured point clouds of indoor environments," *Autom. Construction*, vol. 85, pp. 146–166, Jan. 2018.

[15] A. Candra, M. A. Budiman, and K. Hartanto, "Dijkstra's and A-star in finding the shortest path: A tutorial," in *Proc. Int. Conf. Data Sci., Artif. Intell., Bus. Analytics (DATABIA)*, Jul. 2020, pp. 28–32.

[16] L. Liu, X. Wang, X. Yang, H. Liu, J. Li, and P. Wang, "Path planning techniques for mobile robots: Review and prospect," *Exp. Syst. Appl.*, vol. 227, Oct. 2023, Art. no. 120254.

[17] J. R. Sánchez-Ibáñez, C. J. Pérez-del-Pulgar, and A. García-Cerezo, "Path planning for autonomous mobile robots: A review," *Sensors*, vol. 21, no. 23, p. 7898, Nov. 2021.

[18] I. Ait, E. Kofman, and T. Pire, "A travelling salesman problem approach to efficiently navigate crop row fields with a car-like robot," *IEEE Latin Amer. Trans.*, vol. 21, no. 5, pp. 643–651, May 2023.

[19] S. S. Juneja, P. Saraswat, K. Singh, J. Sharma, R. Majumdar, and S. Chowdhary, "Travelling salesman problem optimization using genetic algorithm," in *Proc. Amity Int. Conf. Artif. Intell. (AICAI)*, Feb. 2019, pp. 264–268.

[20] T. Pathmakumar, M. A. V. J. Muthugala, S. M. B. P. Samarakoon, B. F. Gómez, and M. R. Elara, "A novel path planning strategy for a cleaning audit robot using geometrical features and swarm algorithms," *Sensors*, vol. 22, no. 14, p. 5317, Jul. 2022.

[21] B. Cheng, Y. Wei, H. Shi, R. Feris, J. Xiong, and T. Huang, "Revisiting RCNN: On awakening the classification power of faster RCNN," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 453–468.

[22] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot MultiBox detector," in *Proc. 14th Eur. Conf.*, Amsterdam, The Netherlands. Cham, Switzerland: Springer, Oct. 2016, pp. 21–37.

[23] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[24] J. Terven, D.-M. Córdova-Esparza, and J.-A. Romero-González, "A comprehensive review of YOLO architectures in computer vision: From YOLOv1 to YOLOv8 and YOLO-NAS," *Mach. Learn. Knowl. Extraction*, vol. 5, no. 4, pp. 1680–1716, Nov. 2023.

[25] S. M. B. P. Samarakoon, M. A. V. J. Muthugala, and M. R. Elara, ''Metaheuristic based navigation of a reconfigurable robot through narrow spaces with shape changing ability,'' *Exp. Syst. Appl.*, vol. 201, Sep. 2022, Art. no. 117060.
[26] M. Gen and L. Lin, ''Genetic algorithms and their applications,'' in *Springer Handbook of Engineering Statistics*. Berlin, Germany: Springer, 2023, pp. 635–674.
[27] S. Mirjalili, S. M. Mirjalili, and A. Lewis, ''Grey wolf optimizer,'' *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014.

CHARAN VIKRAM received the Bachelor of Technology degree in electronics and communication from Vellore Institute of Technology, Chennai. He is currently a Visiting Research Scholar with the Robotics and Automation Research (ROAR) Laboratory, part of the Engineering Product Development (EPD) Pillar, Singapore University of Technology and Design (SUTD). His research interests include robotics, perception, and image processing. He was awarded DST Inspire Fellowship with Indian Institute of Science (IISC).

M. K. DOJITHA M. MIRIHAGALLA is currently pursuing the B.Sc. degree in electronic and telecommunication engineering with the University of Moratuwa, Sri Lanka. He is also a Visiting Research Scholar with the Robotics and Automation Research (ROAR) Laboratory, part of the Engineering Product Development (EPD) Pillar, Singapore University of Technology and Design (SUTD). His research interests include reconfigurable robotics, microwave system design, and intelligent vision systems.

MATTHEW S. K. YEO received the M.Arch. and Ph.D. degrees in engineering product development from Singapore University of Technology and Design (SUTD), Singapore, in 2019 and 2024, respectively. He is currently a Research Fellow with the Robotics and Automation Research (ROAR) Laboratory under the Engineering Product Development (EPD) Pillar, SUTD. His research interests include designing robot inclusive spaces, spatial design and reconfigurable robotics, and how they can be implemented in various spatial settings to enhance building infrastructure for improved mobile robot performance. He worked on multiple projects involving mobile reconfigurable service robots and has published multiple articles in journals on the topic of robot inclusive building infrastructure.

ZIMOU ZENG received the bachelor's degree in computer science and design from Singapore University of Technology and Design (SUTD), Singapore, in 2021. He used to work as a Research Officer with the Engineering Product Development Pillar, SUTD. His research interests include the design of robot-inclusive spaces, spatial design, and reconfigurable robotics, focusing on their implementation in various spatial settings.

CHARAN SATYA CHANDRA SAIRAM BORUSU received the B.Tech. degree in electronics and communication engineering from Amrita Vishwa Vidyapeetham and the dual master's degree in nano electronic engineering and design from SUTD and Chang Gung University. He is currently a Senior Research Assistant with the Robotics and Automation Research (ROAR) Laboratory, Singapore University of Technology and Design (SUTD). He is also a Lead Researcher in robotic projects and published multiple papers in journals and conference related to robotics. His research interests include robotic interventions that aims to address contemporary world issues through advanced technological solutions and in project management, emphasizing efficient coordination and strategic planning.

M. A. VIRAJ J. MUTHUGALA (Senior Member, IEEE) received the B.Sc. (Eng.) and Ph.D. degrees in electrical engineering from the University of Moratuwa, Sri Lanka, in 2014 and 2018, respectively.

He is currently a Research Fellow with the Engineering Product Development Pillar, Singapore University of Technology and Design (SUTD), Singapore. His current research interests include maintenance and inspection robotics, reconfigurable robotics, service robotics, human–robot interaction, and intelligent systems.

Dr. Muthugala was a recipient of several IDA International Design Awards, in 2022, Nanjing City Prize in IEEE Ro-Man, in 2018, and Runner-Up in Mini Drone Competition at IEEE/RSJ IROS, in 2018. He was the Assistant Secretary of the IEEE Robotics and Automation Society, Sri Lanka Section Chapter, from 2017 to 2019.

MOHAN RAJESH ELARA received the B.E. degree from Bharathiar University, India, in 2003, and the M.Sc. and Ph.D. degrees from Nanyang Technological University, in 2005 and 2012, respectively.

He is currently an Associate Professor with the Engineering Product Development Pillar, Singapore University of Technology and Design (SUTD). He is also a Visiting Faculty Member of the International Design Institute, Zhejiang University, China. He has published more than 80 papers in leading journals, books, and conferences. His research interests include robotics, with an emphasis on self-reconfigurable platforms and research problems related to robot ergonomics and autonomous systems. He was a recipient of SG Mark Design Award, in 2016 and 2017, ASEE Best of Design in Engineering Award, in 2012, and Tan Kah Kee Young Inventors' Award, in 2010.