Proceedings of the 2025 SICE Festival with Annual Conference
September 9-12, 2025, Chiang Mai, Thailand

Enhancing Orchard Weed Robot Autonomy via Hybrid Path Planning

En-Ming Liu [1†] and Yen-Chen Liu[2]

[1]Department of Mechanical Engineering, National Cheng Kung University (NCKU), Tainan 70101, Taiwan.
(E-mail: N16120349@gs.ncku.edu.tw)

[2]Department of Mechanical Engineering, National Cheng Kung University (NCKU), Tainan 70101, Taiwan.
(E-mail: yliu@mail.ncku.edu.tw.)

Abstract: This paper presents a hybrid path planning system for orchard weeding robots that integrates U-shaped and Z-shaped patterns to generate a global path, ensuring effective coverage of weeds beneath fruit trees. The hybrid path combines the short traversal distance of the U-shaped strategy with the high coverage rate of the Z-shaped strategy. Local path planning, enabled by the Dynamic Window Approach (DWA), ensures real-time obstacle avoidance and smooth turning. Simulations in Gazebo validate the system, showing that the hybrid path reduces traversal distance compared to the Z-shaped path and improves coverage compared to the U-shaped path, while ensuring safe and efficient navigation in dynamic orchard environments.

Keywords: Path planning, DWA, A* algorithms.

1. INTRODUCTION

As the demand for automated agricultural management continues to grow, the development of orchard robots has become increasingly important [1]. Among various operations, inter-row weeding remains particularly challenging, as it requires autonomous navigation to ensure complete coverage of the target area [2], while accounting for the limited operation range of the weeding robot. However, traditional row-based navigation strategies often struggle to adapt to the unstructured nature of orchard environments, where irregular tree spacing poses significant challenges to reliable and efficient operation. In such settings, effective navigation becomes particularly critical.

Recent literature suggests that global path planning in an unknown environment typically involves four key steps: environment modeling, path generation, path optimization, and local path navigation [3]. To support global navigation in unknown and unstructured environments, accurate mapping and localization are required. To overcome the limitations of positioning and modeling in such environments, modern approaches increasingly adopt Simultaneous Localization and Mapping (SLAM) techniques, which enable robots to perform localization without relying on the Global Navigation Satellite System (GNSS). In orchard scenarios, GNSS signals are often blocked by dense tree canopies, resulting in reduced positioning accuracy. Therefore, SLAM technology is particularly critical in these environments [4]. SLAM systems based on LiDAR and vision sensors have demonstrated strong potential for autonomous operation under GPS restricted conditions, and their performance has been validated in simulation environments such as Gazebo [5] and real environment [6].

To achieve effective traversal in specific environments, common robot paths include the spiral, U-shaped, and Z-shaped patterns [7]. The spiral pattern is often used for covering an area from the outside in, making it suitable for tasks that require sweeping a large area [8]. The U-shaped pattern is applied in narrow, uniformly spaced rows to minimize turning and traversal time, making it an efficient choice for orchards with regularly spaced trees [9]. In contrast, the Z-shaped pattern is used in wider rows to allow the robot to maneuver closer to individual trees, thereby enhancing canopy coverage and improving weed detection performance [10].

Although accurate positioning is a prerequisite, effective navigation further requires the integration of global path planning and real-time obstacle avoidance. Traditional navigation frameworks typically separate these two components: the global planner, which uses A* algorithms [11] and the Dijkstra algorithm [12] to generate rough trajectories, and the local planner, usually implemented via the DWA [13], Boundary Node Method (BNM), and Path Enhancement Method (PEM) [14]. In local planning, the global path is refined in real-time by evaluating feasible control commands. While this architecture supports real-time responsiveness, it can generate smooth and reliable trajectories in complex orchard environments, where narrow tree rows and dynamic obstacles are prevalent [15, 16]. Therefore, a method that can simultaneously account for local constraints and spatial layout variations is essential for robust field deployment.

This study presents an autonomous orchard weeding robot framework that integrates SLAM, hybrid path planning, and a layer navigation architecture. The system enables efficient inspection and treatment of weeds beneath fruit trees. The robot is required to patrol along a designated path while maximizing coverage of the weed detection and treatment areas, particularly under tree canopies. To address the coverage limitations caused by irregular row spacing, a hybrid cruising strategy is proposed. The cruising mode is adjusted based on the measured inter-row distance. After the global path is calculated using A*, it is passed to the DWA for path optimization and obstacle avoidance.

† En-Ming Liu is the presenter of this paper.

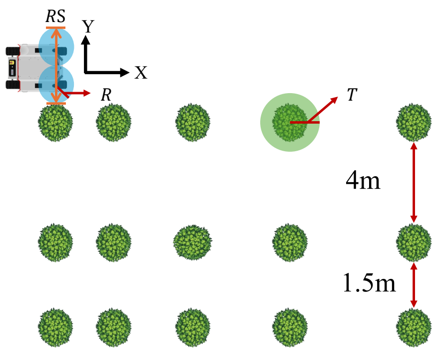

Fig. 1 Orchard layout with heterogeneous row spacings.

The main contributions of this study are as follows:

- A hybrid path planning strategy that dynamically switches between U-shaped and Z-shaped traversal modes based on spatial layout and task objectives, thereby improving coverage performance and adaptability in orchard environments.
- A layer navigation architecture that integrates A* for global path planning and the DWA for local navigation, enabling real-time, obstacle-aware path execution in complex orchard scenarios.

The remainder of this paper is organized as follows: Section 2 shows the main problem of orchard weed robot. Section 3 presents the hybrid path planning method and system design. Section 4 reports the simulation results. Section 5 concludes the paper and outlines future directions.

2. PROBLEM FORMULATION

2.1. Environment and Assumptions

To ensure effective weed detection and treatment under trees, the robot's operation range must fully cover the areas of interest. However, as shown in Fig. 1, irregular tree spacing may cause certain regions to fall outside the robot's effective operating range R, leading to incomplete coverage.

The following assumptions are made:

- The robot operates with a finite operation range R centered around its current position.
- All trees are detectable and accurately mapped during SLAM preprocessing.
- Non-tree obstacles are ignored due to minimal impact in controlled orchards; dynamic obstacles are handled in real-time using the DWA.

2.2. Coverage Completeness

The coverage rate is quantitatively evaluated based on the overlap between the robot's sensing region and the target areas around the trees. Each tree T_j is associated with a target area defined as a circle centered at $(w_x^{(j)}, w_y^{(j)})$ with a fixed radius T.

The coverage rate η is mathematically defined as:

$$\eta = \left(\frac{\sum_{j=1}^{N} \text{Area}(R \cap T_j)}{\sum_{j=1}^{N} \text{Area}(T_j)} \right) \times 100\%, \tag{1}$$

where Area$(R \cap T_j)$ denotes the intersection area between R and T_j, and N is the total number of trees. A coverage rate η close to 100% indicates that the majority of the tree target areas are effectively inspected and treated by the robot during its patrol.

The use of hybrid paths can effectively improve the robot's coverage capability in orchard environments with varying spacing and achieve shorter paths.

3. METHOD

This section introduces the proposed autonomous path planning and navigation system tailored for orchard environments. The system integrates several functional modules, including environmental mapping, tree extraction and row grouping, global path planning based on hybrid U-shaped and Z-shaped strategies, local obstacle avoidance using the DWA. Each module's design and underlying mathematical formulation are presented in detail in the following sections.

3.1. SLAM and Coordinate Transform

In this study, environmental data and real-time perception from a Velodyne LiDAR sensor are integrated using the SLAM Toolbox framework to construct a 2D orchard map. Through continuous updates, the system accurately tracks the spatial distribution of fruit trees, providing essential information for path planning. During subsequent visits to the same environment, the robot leverages the previously built offline map for localization and navigation. This map, once binarized, allows for the identification of individual tree positions. Given that each tree typically spans multiple pixels in the occupancy grid, a Euclidean distance-based clustering algorithm is applied with a threshold derived from actual tree dimensions to merge nearby pixels into unified tree entities. This process enables automatic estimation of the number of tree rows and the number of trees per row, significantly improving the efficiency of global path planning.

The transformation from pixel coordinates (p_x, p_y) to world coordinates (w_x, w_y) is given by:

$$\begin{cases} w_x = x_0 + r \cdot p_x \\ w_y = y_0 + r \cdot (H - p_y). \end{cases} \tag{2}$$

This transformation converts pixel coordinates into real-world positions using the map resolution r and origin parameters (x_0, y_0), and $(H - p_y)$ accounts for the vertical inversion between image and world coordinates.

3.2. Hybrid Global Path Planning

To accommodate varying orchard layouts, this study proposes a hybrid path planning strategy that integrates U-shaped path planning, typically used for structured orchard navigation, and Z-shaped path planning, often employed in coverage intensive inspection tasks. The U-shaped strategy offers efficient traversal with minimal maneuvering in orchards with uniform row spacing, whereas the Z-shaped strategy provides finer control around individual trees, facilitating detailed inspection in irregular layouts.

3.2.1. U-shaped Cruise Strategy

The U-shaped cruise strategy is designed for orchards with evenly spaced rows, aiming to minimize overall path length while ensuring full coverage. In this method, two navigation points are assigned per row, creating an alternating left-right trajectory across adjacent rows. Turn points are inserted at the ends of each row to facilitate smooth transitions. The path initiates from a safe starting position located at the upper-left of the first row and terminates at the lower end of the final row, completing a full traversal of the orchard.

The initial waypoint P_0 is defined in the pixel coordinate system as:

$$p_x^{(0)} = \min(x_{\text{row}_0}) - d_{\text{extra}}, \tag{3}$$

$$p_y^{(0)} = y_{\text{row}_0} - 2d_{\text{safe}}, \tag{4}$$

where d_{extra} is an additional buffer beyond the tree boundary, and $d_{\text{safe}} = \frac{s}{r}$, with s representing the safety margin.

To define turning zones between rows, the midpoint between each consecutive pair i and $i+1$ is computed:

$$y_{\text{mid}}^{(i)} = \frac{y_{\text{row}_i} + y_{\text{row}_{i+1}}}{2}, \tag{5}$$

which guides the robot through U-turn transitions.

To avoid tree collisions, each waypoint P must satisfy the minimum distance constraint from every tree T_k:

$$\text{dis}(P, T_k) = \sqrt{(p_x - x_k)^2 + (p_y - y_k)^2} \geq d_{\text{safe}}, \tag{6}$$

where $T_k = (x_k, y_k)$ denotes the pixel coordinates of the k-th tree. The trajectory alternates between the leftmost and rightmost extents of each row, with U-turns at the ends forming a continuous path. The resulting set of waypoints $\{P_k\}$ constitutes the global path for orchard navigation.

3.2.2. Z-shaped Cruise Strategy

To address the potential coverage gaps caused by the limited operating range R of the robot when using a U-shaped pattern, a Z-shaped path planning strategy is introduced to improve area coverage. This strategy generates a zigzag trajectory by assigning two navigation points for each tree: one placed above (top) and one below (bottom) the tree, separated by a predefined safety distance. Additionally, offset turning points are appended at the end of each row to facilitate smooth transitions. These navigation points are initially generated in pixel coordinates and subsequently transformed into world coordinates using (2).

Since the trajectory begins at the upper-left corner, cross-row transitions consistently occur at the rightmost tree of each row. Therefore, a turning point is placed beyond the last tree in each row, ensuring that transitions lie in free space. This point is computed using a search-based method to adjust the position if necessary, ensuring all navigation and turning points are in traversable areas. Each navigation and turning point is uniquely indexed to ensure path ordering.

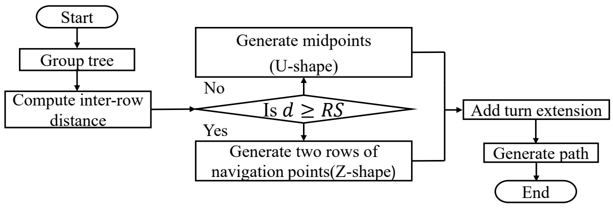

Fig. 2 Flowchart of the hybrid path planning strategy.

The total number of waypoints in the Z-shaped path is computed as $2N$ + t, where N is the total number of trees, and t represents the turning points at the end of each row.

To ensure complete row coverage, the robot alternates between top and bottom points while traversing each tree row. At the end of each row, a turning point guides the transition. For example, in row i, the robot visits top points in ascending index order $(\text{Top}(i, 0), \text{Top}(i, 1), \ldots)$, then passes through $\text{Turn}(i)$, and subsequently visits the bottom points in descending order $(\text{Bottom}(i, m-1), \text{Bottom}(i, m-2), \ldots)$.

3.2.3. Hybrid U-shaped and Z-shaped Cruise Strategy

To address varying row spacings in orchard environments, a hybrid navigation point generation strategy is proposed. This method dynamically selects between U-shaped and Z-shaped patterns based on the inter-row distance. As illustrated in Fig. 2, the process begins by grouping fruit trees into rows according to their Y coordinates, followed by the computation of distances d between adjacent rows.

When the inter-row distance is equal to or greater than robot sensing range RS, as shown in Fig. 1, a Z-shaped pattern is applied: two rows of navigation points are generated one positioned 1 meter below each tree in row i, and another 1 meter above the corresponding tree in row $i+1$. Conversely, if the spacing is less than RS, a U-shaped pattern is adopted, with navigation points placed at the midpoints between the closest tree pairs in adjacent rows. To facilitate smooth transitions between rows, 1-meter turning extensions are added at the end of each row. The final navigation trajectory is computed using the A* algorithm, which connects all navigation points while balancing inspection coverage and traversal efficiency.

This hybrid strategy integrates the shortest length of the U-shaped pattern with the fine-grained inspection capability of the Z-shaped pattern. By dynamically adjusting the navigation mode based on local spatial characteristics, the system adapts to heterogeneous orchard layouts, ensuring comprehensive tree coverage and optimized path length for autonomous navigation.

3.3. Local Path Planning using DWA

The DWA is employed for local path planning, enabling the robot to navigate toward global waypoints while avoiding obstacles and responding to grass detection events. The DWA planner evaluates multiple con-

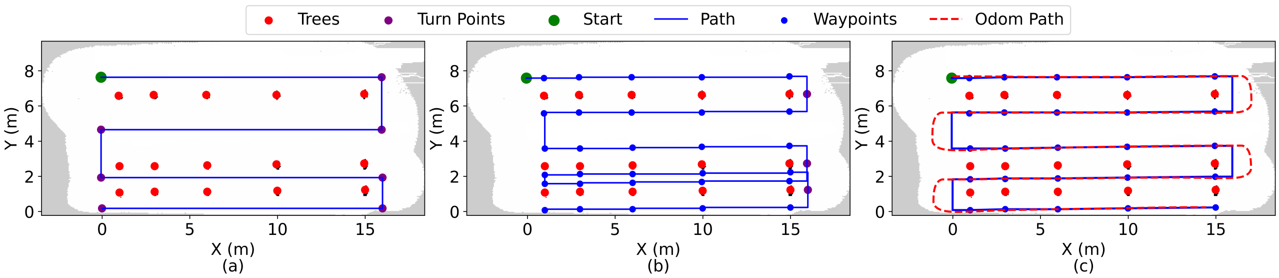

Fig. 3 Path planning performance in a simulated environment with varying tree spacing. (a) U-shaped path. (b) Z-shaped path. (c) Hybrid path. The red dashed line in (c) represents the trajectory generated by combining the DWA.

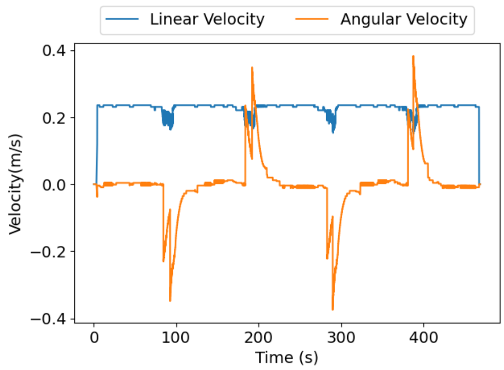

Fig. 4 Linear velocity and angular velocity profile during patrol.

trol rollouts in parallel to generate optimal control commands (v, ω), minimizing a cost function that balances goal alignment, velocity maintenance, and obstacle clearance.

3.3.1. Trajectory Rollout

For each candidate control pair (v, ω), a trajectory is simulated over $T_{sim}/\Delta t$ steps using a forward kinematic model:

$$x_{t+1} = x_t + v \cdot \cos(\theta_t) \cdot \Delta t. \tag{7}$$

$$y_{t+1} = y_t + v \cdot \sin(\theta_t) \cdot \Delta t. \tag{8}$$

$$\theta_{t+1} = \theta_t + \omega \cdot \Delta t. \tag{9}$$

Here, x_t and y_t denote the robot's position at time t, and θ_t is the orientation angle. The position updates in (7) and (8) are governed by the current heading, which dictates the motion direction for a given linear velocity v.

The heading angle is updated in (9), where ω is the angular velocity input. The equation integrates ω over timestep Δt across the total simulation duration T_{sim}, modeling the robot's rotational motion. Thus, ω directly shapes the trajectory's curvature. By adjusting the ratio of v to ω, the turning radius $t_r = \frac{v}{\omega}$ can be controlled.

3.3.2. Cost Function Evaluation

Each trajectory is evaluated using a weighted sum of three cost components C:

$$C(v, \omega) = w_{\text{ang}} \cdot C_{\text{ang}} + w_{\text{vel}} \cdot C_{\text{vel}} + w_{\text{obs}} \cdot C_{\text{obs}}. \tag{10}$$

The angle cost C_{ang} is defined as $C_{\text{ang}} = \pi - |\Delta\theta|$, where $\Delta\theta$ is the angle difference between the target direction and the final orientation of the trajectory. The velocity cost C_{vel} is defined as $C_{\text{vel}} = v$, where v is the linear velocity of the control command. The obstacle cost C_{obs} is defined as $C_{\text{obs}} = \min(\text{dis}_{\text{obstacle}})$, where $\text{dis}_{\text{obstacle}} = -\text{score}$, and score is a weighted sum of nearby costmap values. Higher scores near obstacles yield more negative $\text{dis}_{\text{obstacle}}$, penalizing unsafe trajectories. During navigation, the robot switches between three driving states: approaching a target, moving straight, and turning. Each state uses different weights in the cost function; angular alignment w_{angle} is emphasized during turning and approach, while linear velocity w_{vel} is prioritized during straight motion to enhance efficiency.

3.3.3. Summary

The DWA local planner enables real-time and responsive navigation in orchard environments by optimizing control commands for trajectory tracking and obstacle avoidance. By dynamically adjusting the weighting factors within the cost function, the planner can adapt the generated control commands to different navigation states.

4. SIMULATION

An orchard simulation was built in Gazebo, with Robot Operating System 2 (ROS2) handling communication and control. The system processes sensor data (images and LiDAR) in real time and sends velocity commands for autonomous patrol. As shown in Fig. 1, the orchard includes three tree rows, with 4 m and 1.5 m spacing between rows.

Fig. 3 illustrates three distinct path planning strategies. For each strategy, both the path length and the corresponding coverage rate are evaluated using (1), with the results summarized in Table 1. The results from the Table 1 clearly highlight the advantages of the Hybrid path. In terms of traversal efficiency, the Hybrid path reduces the total distance traveled by 13.7 meters compared to the Z-shaped path, resulting in a more efficient weeding process. Furthermore, while the Hybrid path's coverage rate is only 0.5% lower than the Z-shaped path, it significantly outperforms the U-shaped path, with an increase of 19.5% in coverage. This demonstrates that the Hybrid path effectively balances path length reduction and

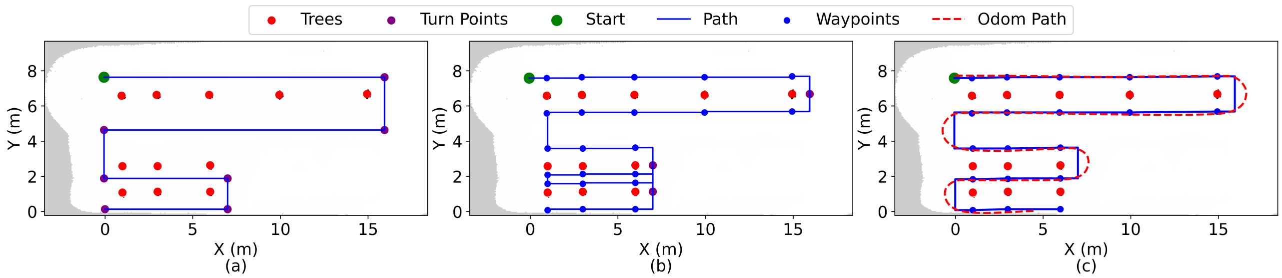

Fig. 5 Path planning performance in a simulated environment with areas lacking trees. (a) U-shaped path. (b) Z-shaped path. (c) Hybrid path.

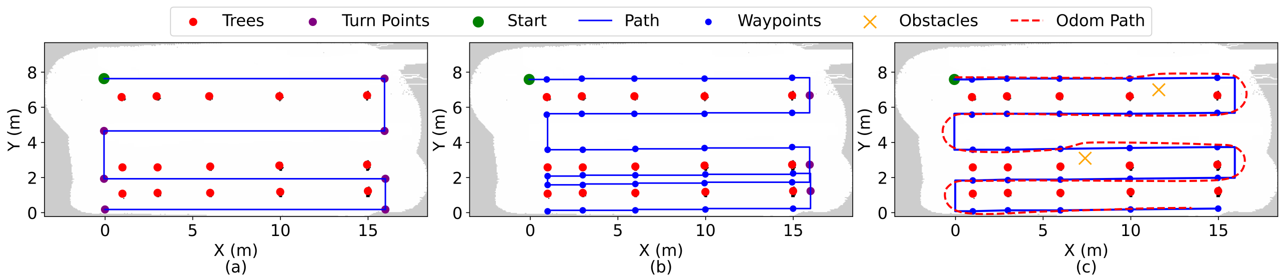

Fig. 6 Path planning performance in a simulated environment with varying tree spacing. (a) U-shaped path. (b) Z-shaped path. (c) Hybrid path. The yellow markers in (c) represent obstacles in the orchard.

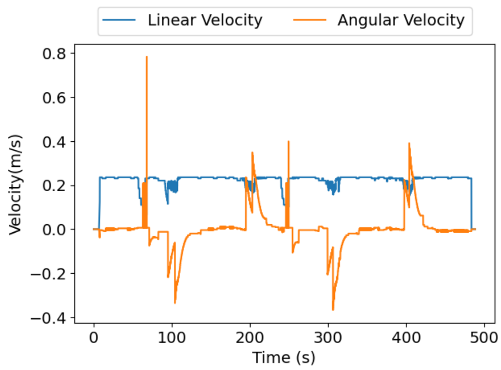

Fig. 7 Linear velocity and angular velocity profile during obstacle.

coverage performance, offering a superior solution for orchard weeding tasks. These results indicate that the hybrid path can effectively balance path efficiency and coverage performance. Specifically, it can significantly reduce path length and energy consumption compared to the Z-shaped path, while avoiding the substantial coverage loss observed when using the U-shaped path alone.

Table 1 Comparison of path length and coverage rate η under different strategies

Path Type	U-shape	Z-shape	Hybrid
Length (m)	71.5	100.2	86.5
η (%)	80.0	100.0	99.5

Fig. 3(c) shows the odometry paths, demonstrating how the local planner further refines the hybrid global path using the DWA. Table 2 summarizes the weights used by the DWA algorithm to adjust the robot's (v, ω). During turning, the robot's kinematic constraints are considered, and the ratio between v and ω is adjusted to produce a smooth trajectory, thereby effectively controlling the turning radius. As shown in Fig. 4, the robot is actively turning when ω remains nonzero and is not the result of a small angular error. By limiting the ratio between v and ω, the turning angle can be controlled and the turning speed can be reduced, allowing the robot to perform smooth turns.

Table 2 Weight configurations for different states.

State	w_{angle}	w_{vel}	w_{obs}
Near Goal	0.6	0.05	0.1
Turning Mode	0.5	0.05	0.1
Straight Driving	0.025	0.1	0.1

Fig. 5 demonstrates that even when the number of trees in each row of the orchard varies, the robot can plan its global path according to the current situation, effectively avoiding areas without trees. The three path planning methods proposed are capable of achieving this result. Fig. 6(c) illustrates the robot's obstacle avoidance capability, showing that it can return to the planned trajectory after avoiding obstacles. In Fig. 7, a particularly large ω is generated when the robot passes the first and second obstacles along the straight section, indicating that the robot is very close to the obstacles and triggering large-angle maneuvers for avoidance. After bypassing the obstacles, the robot returns to the original planned path.

In summary, the simulation results validate the effectiveness of the proposed hybrid path planning framework. The hybrid strategy successfully balances path efficiency and coverage performance, achieving near complete coverage with a reduced path length compared to conventional strategies. Moreover, the integration of DWA enables the robot to perform smooth turns and robust obstacle avoidance in a dynamically changing environment, ensuring safe and efficient autonomous navigation throughout the orchard.

5. CONCLUSION AND FUTURE WORK

This research makes an important contribution to the field of path planning for autonomous orchard weeding robots. A hybrid cruising strategy, combining U-shaped and Z-shaped patterns, is proposed to dynamically adapt to different orchard layouts and operational requirements, achieving a high degree of coverage and a shortened operation path. In addition, a hybrid path planning framework is developed by integrating the A* algorithm for global path generation with the DWA algorithm for real-time local obstacle avoidance, ensuring safe and efficient navigation in complex orchard environments.

In future work, the mobile platform can be extended beyond weed detection to support tasks such as pest identification, fruit ripeness assessment, and disease monitoring by training dedicated visual models. These extensions are aimed at establishing a comprehensive, smart orchard management system that will improve overall agricultural productivity and promote the advancement of smart farming technologies. Subsequently, both simulation and field experiments will be conducted to validate the proposed functions and assess system performance, thereby ensuring the feasibility and practical applicability of the system in real-world agricultural environments.

6. ACKNOWLEDGMENT

This work was supported in part by the National Science and Technology Council (NSTC), Taiwan, under Grant NSTC 114-2218-E-006-021 and NSTC 114-2628-E-006-010.

REFERENCES

[1] R. R. D. Abeyrathna and T. Ahamed, "Autonomous robots in orchard management: present status and future trends," *IoT and AI in Agriculture: Self-sufficiency in Food Production to Achieve Society 5.0 and SDG's Globally*, pp. 335–366, 2023.

[2] D. Reiser, E.-S. Sehsah, O. Bumann, J. Morhard, and H. W. Griepentrog, "Development of an autonomous electric robot implement for intra-row weeding in vineyards," *Agriculture*, vol. 9, no. 1, 2019.

[3] T. Schönegg, T. Tuna, F. Yang, G. Waibel, M. Mattamala, and M. Hutter, "Global path planning for autonomous vehicles in orchards and vineyards," in *2024 13th International Workshop on Robot Motion and Control (RoMoCo)*, pp. 1–8, 2024.

[4] K. Inoue, Y. Kaizu, S. Igarashi, and K. Imou, "The development of autonomous navigation and obstacle avoidance for a robotic mower using machine vision technique," *IFAC-PapersOnLine*, vol. 52, pp. 173–177, 01 2019.

[5] Q. Li and H. Zhu, "Performance evaluation of 2D lidar SLAM algorithms in simulated orchard environments," *Computers and Electronics in Agriculture*, vol. 221, p. 108994, 06 2024.

[6] S.-H. Chan, P.-T. Wu, and L.-C. Fu, "Robust 2D indoor localization through laser SLAM and visual SLAM fusion," in *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 1263–1268, 2018.

[7] S. Kalaivanan and R. Kalpana, "Coverage path planning for an autonomous robot specific to agricultural operations," in *2017 International Conference on Intelligent Computing and Control (I2C2)*, pp. 1–5, 2017.

[8] H. Zhang, W. Hong, and M. Chen, "A path planning strategy for intelligent sweeping robots," in *2019 IEEE International Conference on Mechatronics and Automation (ICMA)*, pp. 11–15, 2019.

[9] Q. Qiu and X. Li, "Lidar point-cloud odometer based mobile robot routine tracking in orchards," in *2022 12th International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*, pp. 735–739, 2022.

[10] L. Zhu, S. Yao, B. Li, A. Song, Y. Jia, and J. Mitani, "A geometric folding pattern for robot coverage path planning," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 8509–8515, 2021.

[11] S. Erke, D. Bin, N. Yiming, Z. Qi, X. Liang, and Z. Dawei, "An improved A-star based path planning algorithm for autonomous land vehicles," *International Journal of Advanced Robotic Systems*, vol. 17, no. 5, p. 1729881420962263, 2020.

[12] L. Parungao, F. Hein, and W. Lim, "Dijkstra algorithm based intelligent path planning with topological map and wireless communication," *ARPN Journal of Engineering and Applied Sciences*, vol. 13, no. 8, pp. 2753–2763, 2018.

[13] C.-T. Vu, H.-C. Chen, and Y.-C. Liu, "Toward autonomous navigation for agriculture robots in orchard farming," in *2024 IEEE International Conference on Recent Advances in Systems Science and Engineering (RASSE)*, pp. 1–8, 2024.

[14] R. A. Saeed, G. Tomasi, G. Govindarajan, R. Vidoni, and K. D. Von Ellenrieder, "Metrology-aware path planning for agricultural mobile robots in dynamic environments," in *2021 IEEE International Workshop on Metrology for Agriculture and Forestry (MetroAgriFor)*, pp. 448–453, 2021.

[15] L. Liu, X. Wang, X. Wang, X. Jinyan, H. Liu, J. Li, P. Wang, and X. Yang, "Path planning and tracking control of tracked agricultural machinery based on improved A* and fuzzy control," *Electronics*, vol. 13, p. 188, 01 2024.

[16] J. Niu, C. Shen, L. Zhang, G. Gao, and J. Zheng, "Study on multiobjective path optimization of plant protection robots based on improved aco-dwa algorithm," *Engineering Reports*, vol. 7, no. 2, p. e70009, 2025.