

Simultaneous Localization and Mapping and Tag-Based Navigation for Unmanned Aerial Vehicles

Mohammad Soleimani Amiri^{1,2}, Rizauddin Ramli^{1,3*}, Aiman Hakimi Faizal¹

¹Department of Mechanical and Manufacturing Engineering, Faculty of Engineering & Built Environment, Universiti Kebangsaan Malaysia, 43600 UKM Bangi, MALAYSIA

²Faculty of Mechanical and Manufacturing Engineering Technology, Universiti Teknikal Malaysia Melaka, 76100 Durian Tunggal, Melaka, MALAYSIA

³Center for Automotive Research (CAR), Faculty of Engineering & Built Environment, Universiti Kebangsaan Malaysia, 43600 UKM Bangi, MALAYSIA

*Corresponding Author

DOI: <https://doi.org/10.30880/ijie.2023.15.05.024>

Received 01 August 2023; Accepted 15 August 2023; Available online 19 October 2023

Abstract: This paper presents navigation techniques for an Unmanned Aerial Vehicle (UAV) in a virtual simulation of an indoor environment using Simultaneous Localization and Mapping (SLAM) and April Tag markers to reach a target destination. In many cases, UAVs can access locations that are inaccessible to people or regular vehicles in indoor environments, making them valuable for surveillance purposes. This study employs the Robot Operating System (ROS) to simulate SLAM techniques using LIDAR and GMapping packages for UAV navigation in two different environments. In the Tag-based simulation, the input topic for April Tag in ROS is camera images, and the calibration of position with a tag is done through assigning a message to each ID and its marker image. On the other hand, navigation in SLAM was achieved using a global and local planner algorithm. For localization, an Adaptive Monte-Carlo Localization (AMCL) technique has been used to identify factors contributing to inconsistent mapping results, such as heavy computational load, grid mapping accuracy, and inadequate UAV localization. Furthermore, this study analyzed the April Tag-based navigation algorithm, which showed satisfactory outcomes due to its lighter computing requirements. It can be ascertained that by using ROS packages, the simulation of SLAM and Tag-based UAV navigation inside a building can be achieved.

Keywords: Unmanned Aerial Vehicles, autonomous navigation, April Tag

1. Introduction

Autonomous navigation of vehicles, especially Unmanned Aerial Vehicles (UAVs) or drones, played an essential role in Industrial Revolution 4.0. The accuracy of the drone's mathematical model is crucial for precise navigation, ensuring the avoidance of potential dangers. This has led to an increased demand for the establishment of simultaneous localization and navigation algorithms. In general, drone can reach an environment that are difficult to be entered by humans and other vehicles [1]. These vehicles require continuous signals, such as throttle, pitch, roll, and yaw, while deriving their movement from collected information. Currently, their usage significantly impacts the development of the Internet of Things (IoT), which is one of the categories of Industrial Revolution 4.0 [2].

In recent years, several studies have focused on the components and configurations of Unmanned Aerial Vehicles (UAVs); for instance, wings and rotors are used to classify them [3]. UAVs with more than two generating rotors are known as multi-rotor UAVs. On one hand, the fixed-wing type uses wings like a regular airplane, while on the other

*Corresponding author: rizauddin@ukm.edu.my

225

2023 UTHM Publisher. All rights reserved.

penerbit.uthm.edu.my/ojs/index.php/ijie

hand, a single-rotor type has a design and structure like a regular helicopter. The advantage of a fixed-wing UAV is longer flight time, combined with the capacity to hover like a rotor-based drone in the VTOL Hybrid UAV. Due to the cost and risk of damage to UAVs, simulation has become essential to test their performance before conducting actual experiments. Several state-of-the-art drone simulation tools, such as Robot Operating Systems (ROS) Gazebo, and RVIZ, have been used to carry out flight simulations. These simulations combine physical and real-world aspects, including robot behaviors. Furthermore, various integrated sensors such as LIDAR (Light Detection and Ranging), Inertia Measurement Units (IMUs), ultrasonic sensors, and Global Positioning Systems (GPS) have been employed to measure the distance, position, and orientation of the drone in relation to its target object [4]. In another study, intelligent methods, such as Convolutional Neural Networks (CNNs), have been used to study multiple filters in parallel with a dataset, particularly in the context of specific predictive modeling problems like picture classification. Additionally, fuzzy logic control (FLC) has been utilized as a control system that gathers information about resistance with the aid of sensors, enabling it to adjust the direction, angle, and speed of the UAV [5].

Based on sensor input and mapping, Simultaneous Localization and Mapping (SLAM) has been widely adopted as one of the most popular methods to serve as the foundation of a self-navigating system for drones or robots. SLAM provides a framework for the drone to determine its location, build a map of its surroundings, and plan a path or trajectory to its destination. In dynamic environments, the drone needs to continuously update its mapping, and SLAM supports this in real-time, involving a continuous process of estimating the robot's position and orientation while generating a map in an unfamiliar environment [6]. Azril et al. studied different types of localization and mapping, including Filtering-Based SLAM [7], which relies on particle usage to solve various types of optimizations, pathfinding, and searching problems, such as routing problems and environment mapping using laser sensors. On the other hand, Vision-Based SLAM, also known as graph optimization, estimates the robot's position by using image data from the camera simultaneously, and its optimization process is used to refine the robot's pose in the current environment. Another method, Adaptive Monte-Carlo Localization (AMCL), is based on Filtering-Based SLAM and sampled adaptively using error estimation with Kullback-Leibler Differential (KLD) [8]. The AMCL method has been used for UAV disaster response operations and is widely employed as an improved version of Monte-Carlo Localization (MCL) [9].

However, the AMCL uses a high computational rate, which can be wasteful when used in smaller environments. Similarly, GMapping, another Filtering-Based SLAM technique, utilizes Rao-Blackwellized particle filters to construct 2D grid maps. It is widely known for UAV mapping integrated with LIDAR and odometry, and it's one of the most used SLAM techniques due to its ability to produce good mapping results. However, studies have shown that GMapping may not produce accurate maps for outdoor UAV navigation [10]. As a result, the hector SLAM technique, a 2D method, has been adopted for mapping indoor environments without utilizing sensor odometry, roll, pitch, and yaw motion. It is mainly employed for search and rescue operations [11]. While this technique has proven to be efficient for drone navigation in indoor environments, it lacks feedback from a closed-loop system, and its performance can be affected by noise [12].

In addition, 3D localization using HDL Graph SLAM, which is based on a graph with odometrical calculations using normal distributions transform matching and loop sensing, has been proposed by researchers [13]. Compared to 2D LIDAR-based SLAM methods such as Hector SLAM and GMapping, HDL SLAM supports the integration of GPS, IMU acceleration and orientation, and point cloud data. However, HDL SLAM requires a larger LIDAR, which is not suitable for smaller robots. Therefore, an LSD SLAM, a visual SLAM method that uses direct and probabilistic image alignment and semi-compact depth maps [14] (Engel et al., 2015), has also been proposed. For instance, LSD SLAM has been used for mapping and obstacle avoidance for UAVs in outdoor areas, as demonstrated in [15], showing its efficiency for vision-based SLAM in UAV navigation [16] [17]. Another SLAM method, RTAB-Map, has been applied for mobile robot mapping in the lobby area of a building to estimate robot positioning [18]. It utilizes RGB data better than most vision-based SLAM approaches. To navigate a robot or UAV to its goal, navigation using a recorded map is a commonly practiced method by many researchers. For instance, the local planner called the Dynamic Window Approach (DWA) can avoid barriers in a mapped environment (http://wiki.ros.org/dwa_local_planner). The planner can select speeds that allow the robot to stop safely if a dynamic impediment is encountered. By maximizing an objective function that handles the progress toward the target, the velocity, and the distance from any barriers on the route, the best speed is determined.

This research aims to conduct a UAV movement simulation for SLAM and navigation using April Tag markers inside a building, with a particular focus on identifying the methods used to construct indoor environment maps and create a map for determining the position of the drone.

2. Autonomous Tag-Based UAV Navigation

In this study, the Gazebo simulator is incorporated and utilized to develop the autonomous drone application. Gazebo is a well-known industrial platform that facilitates the creation and testing of realistic 3D simulations of autonomous robots in both indoor and outdoor environments. It offers support for real-world sensors, including image capture devices and LIDAR. Fig. 1 represents the location of the UAV regarding the home location and the coordinate system.

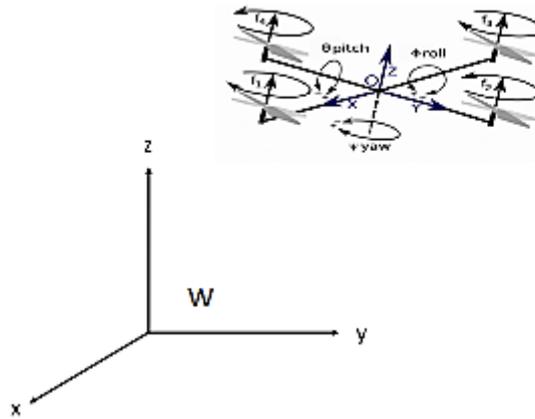


Fig. 1 - UAV and its coordinate systems

The type of UAV used in this simulation is a quadcopter, which is equipped with four motors and rotors. The quadcopter structure comprises two motors that rotate clockwise and two others that rotate counterclockwise. By having the motors on the diagonals turn in the same direction, it becomes feasible to control the UAV's directions effectively. The motors are numbered one through four, and the position and direction of each motor are illustrated in Fig. 1. The UAV dynamic model using Euler-Lagrange method is given as follows,

$$\ddot{\mathbf{x}} = \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \frac{\mu}{m} \begin{bmatrix} \sin\psi\sin\phi + \cos\psi\sin\theta\cos\phi \\ \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi \\ \cos\theta\cos\phi - \frac{g \cdot m}{u_q \cdot \mu} \end{bmatrix} u_q \tag{1}$$

where $\mathbf{x} = [x, y, z]^T$ is the position of the quadcopter with respect to W coordinate frame. $\sigma = [\psi, \theta, \phi]^T$ are the Tait-Bryan angles with respect to W, m is the mass of the quadcopter in kg. μ is the thrust factor in (N.s²), g is the constant of gravity in(m/s²) and u_q is the total force, in N, supplied by the four rotors over the z-axis with respect to the coordinate system Q [19]. The dynamic system of UAV fixed body coordinate system relative to W coordinate frame is given as follows:

$$\ddot{\sigma} = \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} \left(\frac{I_y - I_z}{I_x}\right) \dot{\theta}\dot{\psi} + \left(\frac{J_p}{I_x}\right) \dot{\theta}\Omega + \left(\frac{\mu}{I_x}\right) \tau_\phi \\ \left(\frac{I_z - I_x}{I_y}\right) \dot{\phi}\dot{\psi} + \left(\frac{J_p}{I_y}\right) \dot{\phi}\Omega + \left(\frac{\mu}{I_y}\right) \tau_\theta \\ \left(\frac{I_x - I_y}{I_z}\right) \dot{\phi}\dot{\theta} + \left(\frac{d}{I_z}\right) \tau_\psi \end{bmatrix} \tag{2}$$

$$\ddot{\sigma} = \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} \left(\frac{I_y - I_z}{I_x}\right) \dot{\theta}\dot{\psi} + \left(\frac{J_p}{I_x}\right) \dot{\theta}\Omega + \left(\frac{\mu}{I_x}\right) \tau_\phi \\ \left(\frac{I_z - I_x}{I_y}\right) \dot{\phi}\dot{\psi} + \left(\frac{J_p}{I_y}\right) \dot{\phi}\Omega + \left(\frac{\mu}{I_y}\right) \tau_\theta \\ \left(\frac{I_x - I_y}{I_z}\right) \dot{\phi}\dot{\theta} + \left(\frac{d}{I_z}\right) \tau_\psi \end{bmatrix} \tag{3}$$

where, J_p is the drag factor in (N. m. s²). I_x, I_y , and I_z are the inertia moment over each axis, τ_ϕ, τ_θ , and τ_ψ are the torque supplied by the rotors over each axis, in (N. m). Ω represents the overall propeller's speed in (rad/s). The input propeller's speed given as follows:

$$\begin{bmatrix} u_q \\ \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -l & 0 & l & 0 \\ 0 & l & 0 & -l \\ 1 & -1 & 1 & -1 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} \tag{4}$$

where,

$$f_j = \omega_j^2 \quad (5)$$

$$\Omega = \omega_1 - \omega_2 + \omega_3 - \omega_4 \quad (6)$$

Here, $j = 1, 2, \dots, 4$ is denoted as the number of the propellers, l (m) is the distance from the center of mass of the quadcopter to the center of mass of one rotor. $f_j(N)$ is the trust force supplied for the j rotor and ω_j (rad/s) is the angular velocity of the rotor.

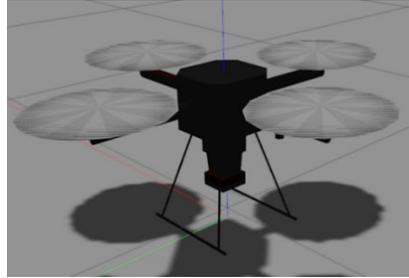


Fig. 2 - UAV 3D model in Gazebo environment

Fig. 2 depicts the 3D model of the UAV used for simulation in the ROS Gazebo environment and Fig. 3 shows the flowchart of navigation for UAV using SLAM. It starts from launching the Gazebo model, UAV take off and making map using 2D LIDAR. In the ROS Gazebo simulation environment, the LIDAR sensors can be incorporated to mimic real-world LIDAR functionality within the simulated environment which emulates the real-world functionality and provides essential perception capabilities for simulated UAV. By using the simulated environment, it navigates the UAV to destination point and finally landed.

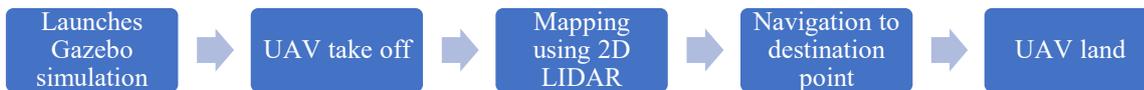


Fig 3 - Flowchart of navigation for UAV using SLAM

In this study, we utilized the AprilTag integrated with ROS to enable UAV navigation in the Gazebo simulation environment. The subsequent stage of the research involved mapping the indoor environment, and the outcomes of the mapping process were visualized in RVIZ. By following the mapping, the UAV is autonomously navigated within the indoor environment using both local and global planner algorithms. In this configuration, Dijkstra's algorithm is used as the global planner, and the DWA algorithm serves as the local planner as depicted in Fig. 4. The Dijkstra's algorithm is a classical graph search algorithm that finds the shortest path between two points in a graph with non-negative edge weights. In the context of UAV navigation, the environment is represented as a grid or graph, where each cell or node represents a location, and the edges between nodes have weights corresponding to the traversal cost. Dijkstra's algorithm searches the graph from the start position to the goal position, considering the cumulative cost of reaching each node, and generates the optimal path. The DWA algorithm uses the UAV's current state, LIDAR scans data, and kinematic constraints to predict the drone's motion over a short time horizon. It explores a dynamic window of potential velocities and headings that the robot can achieve in the next few time steps. In addition, the DWA algorithm adjusts the UAV's trajectory to handle immediate obstacles and uncertainties in real-time, ensuring effective and safe navigation in complex and dynamic environments. The tag-based navigation involved navigating the UAV using AprilTag markers. The family tag used is 36h11. The drone's video stream detects the AprilTag, and whenever an ID tag is detected, a command is given to the drone for autonomous navigation within the Gazebo environment.

3. Results and Discussion

This study uses project Gaze to build a moderately sized interior environment of 23 m x 23 m with barriers. Some of these wall objects are used and placed in the right direction to produce an internal environment consisting of 4 small rooms shown in Fig 5 (a). Next, GMapping is launched for 2D mapping of the indoor environment, and UAV collects laser detection input data. The input is laser scan data from a Hokuyo 2D LIDAR sensor, and the output is a probabilistic 2D environment grid map. Environmental grid maps are updated as they move autonomously using global road planners and local road planners as navigation algorithms. White color on the map means space, the black lines are the obstacle, and grey space means the area has not been explored. The mapping results are shown in Fig. 5 (b). Table 1 summarizes the results of navigation from different destinations using the SLAM algorithm.

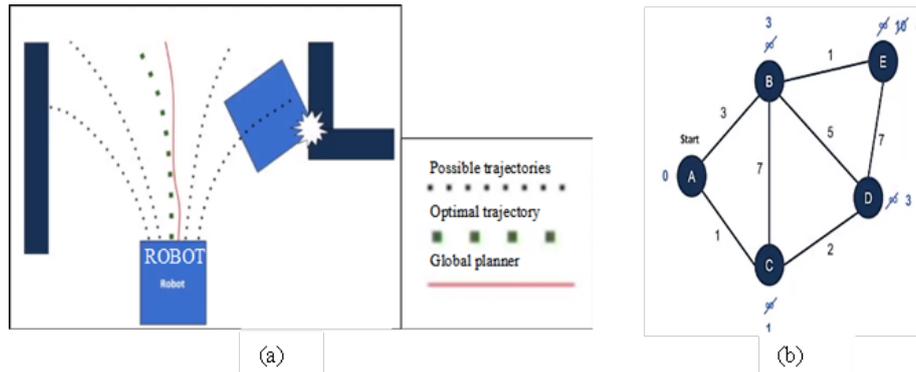


Fig 4 - (a) DWA algorithm; (b) Dijkstra's algorithm

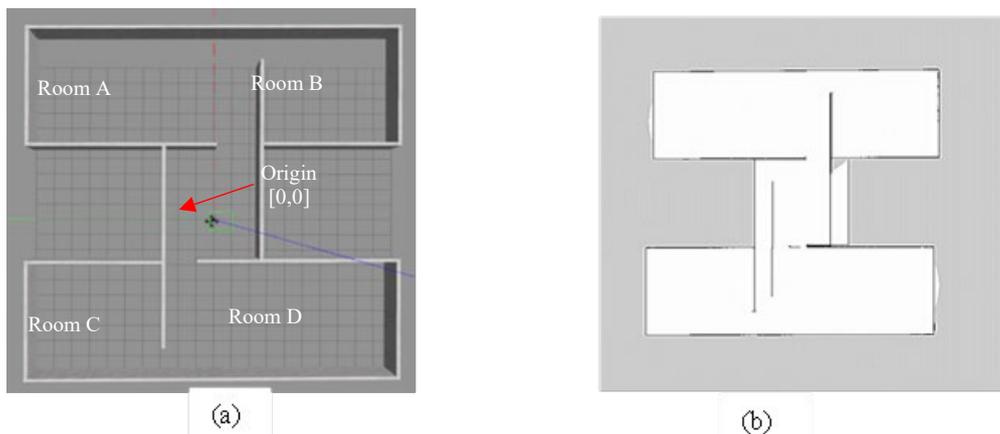


Fig 5 - (a) Top view of the simulation environment in Gazebo; (b) mapping result of the indoor environment with GMapping

Table 1 - Navigation results for various destination goals

Starting point (m)	Target destination	Final position (m)	Distance (m)	Time elapsed (s)	Average speed (m/s)
[0,0]	A	[10.73, 8.91]	13.94	68	0.31
[0,0]	B	[10.73, -7.93]	13.34	59	0.23
[0,0]	C	[-8.99, 8.91]	12.65	171	0.11
[0,0]	D	[-8.31, -7.71]	11.34	152	0.13

Figure 6 presents the navigation outcomes in Room A-D, depicting the distance covered, time taken, and average velocity of the drone while traveling from its initial home position to the designated target position. The goal of the 2D navigation was successfully determined, and the UAV autonomously navigated from the starting point to the specified destination using Dijkstra's global planner algorithm and DWA local planner to avoid wall obstacles along the path. The time taken for the UAV to complete each 2D navigation goal depends on the distance between the starting point and the destination room, as well as the UAV's selected velocity. The details of the starting point, target point, travel distance, time taken, and velocity for each room are recorded and presented in Table 1.

The important success criteria for this study are that the UAV successfully navigates to the target destination. According to Table 1, the maximum velocity achieved by the UAV is 0.31 m/s. The navigation from the central point to Room A takes 68 seconds, and the UAV proves to be slower at other times. For instance, the time taken from the main point to Room C is longer, approximately 171 seconds, despite the shorter travel distance. This indicates that more work is needed to achieve a more consistent UAV navigation runtime simulation.

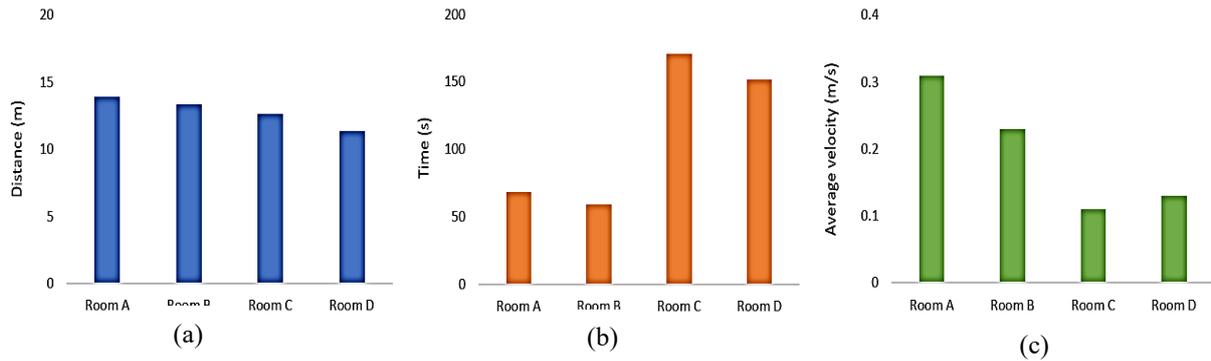


Fig. 6 - SLAM based navigation result (a) distance passed by UAV; (b) the duration of navigation in seconds; (c) the average speed of the UAV

The second success criteria involve the production of a quality grid map by the UAV. The mapping of the interior area in this study took approximately 2 to 3 hours. Ideally, for a moderately sized indoor environment without distortion, the mapping process should take only 30 minutes to 1 hour [20]. The longer time to produce the grid map was attributed to the UAV's navigation computation and obstacles. To capture the full dimensions of environmental features, the UAV needs to move at a slow velocity, resulting in a larger number of LIDAR scans and intensive computation, contributing to the time required for more extensive grid maps.

An alternative to 2D mapping is a 3D environment map that can detect obstacles both above and below the height of a hovering UAV. However, 3D mapping requires more intensive computing than 2D mapping and demands more powerful processing in the UAV [21]. In this paper, we navigated the UAV based on the April Tag. The interior construction in the Gazebo environment consists of an area measuring 17m x 17m. The construction includes six small walls, each with a size of 1.5m x 2.8m. On these small walls, a total of 6 April Tag IDs, each measuring 0.5m x 0.5m, are pasted. The height of the small walls is 1.5m. The type of April Tag used in this simulation is from the 36H11 family, with ID tags ranging from 1 to 6. Fig. 7 illustrates simulated environment in Gazebo for UAV tag-based navigation. Fig. 8 shows relative position between the UAV camera frame and April Tag frame. In Fig. 8, the blue-colored plane represents the z-plane, the red-colored plane represents the x-plane, and the green-colored plane indicates the y-plane. In this figure, the UAV successfully detects the April Tag displayed in RVIZ and the UAV camera. By opening the /front cam/camera/image topic, which provides the video stream from the camera and includes the image acquisition matrix, the video can also display the planes. Table. 2 UAV navigation results using April Tag.

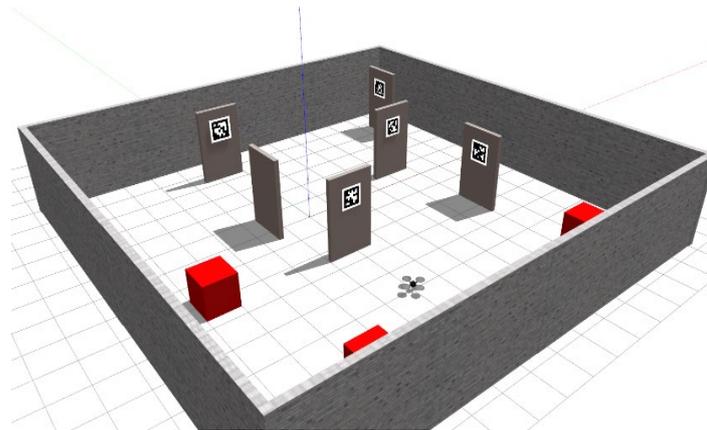


Fig. 7 - Tag-based navigation environment in Gazebo

The UAV navigated from ID tag 1 to ID tag 6 in a tag-based simulated environment. Throughout the navigation process, the UAV maintained a consistent velocity of 0.5 m/s from the beginning to the end of the navigation. The time taken for the UAV to travel from its initial position to ID tag 1 was 8 seconds, while the longest time taken from ID tag 5 to ID tag 6 was 64 seconds. The error tolerance for tag-based navigation was found to be less than 0.1 m. Based on the results, it can be concluded that navigation using April Tag is more satisfactory compared to using SLAM because it places a lighter computing load on the UAV. The navigation using April Tag involves a tag algorithm customized with built-in Python instructions, and the detection algorithm relies on the UAV camera. Overall, the use of April Tag for navigation proves to be efficient and effective, offering advantages over SLAM in terms of computing load and accuracy.

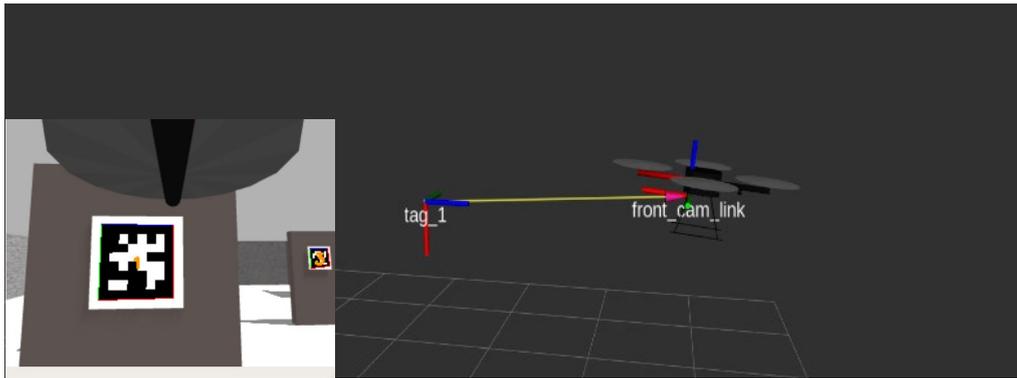


Fig. 8 - The relative position between the UAV camera and April Tag

Table 2 - UAV navigation results using April Tag from starting position [0, -5]m

ID tag number	ID tag position (m)	Distance between ID tag and camera (m)	UAV position (m)	Distance (m)	Time elapsed (s)	Average speed (m/s)
1	[-0.2, -2.5]	0.086	[0, -3.7]	1.3	8	0.5
2	[4.9, -2.8]	0.085	[3.7, -3.5]	5	19	0.5
3	[4.1, 0.6]	0.099	[3.7, -0.4]	8.1	30	0.5
4	[-1.3, 0.2]	0.099	[-0.5, 3.4]	12.3	41	0.5
5	[-0.7, 4.3]	0.091	[-0.5, 3.4]	16	52	0.5
6	[6.4, 4.0]	0.141	[3.2, 3.4]	19.7	64	0.5

4. Conclusion

This paper presented novel navigation techniques for UAVs based on April Tag. The tag-based UAV navigation technique that utilized April Tags for scanning and receiving commands was implemented. The autonomous UAV movement simulation using the SLAM algorithm and April Tag detection was successfully carried out within the virtual environment. These navigation techniques tested in a virtual simulation have been developed successfully, serving as the novelty of this study. Therefore, this paper presents a comparative analysis between the two navigation methods for UAVs, contributing to the growth of the robotics industry. Based on the findings in this study, we can recommend several improvements that can be implemented for future studies. The main recommendations and follow-up studies are for implementing the simulated UAV movement using SLAM and using April Tags in the real world. Also, in future studies, 3D mapping and road planning, computer vision-based obstacle avoidance, and object detection using artificial intelligence such as Neural Network, Fuzzy Logic, and others can be considered.

References

- [1] A. D. Acharya, S. Bhandari and Z. Aliyazicioglu, "Autonomous Navigation of a Quadcopter in Indoor Environments," in *AIAA Scitech 2019 Forum*, San Diego, California, 2019.
- [2] Y. Liu, K. Liu, J. Han, L. Zhu, Z. Xiao and X.-G. Xia, "Resource Allocation and 3-D Placement for UAV-Enabled Energy-Efficient IoT Communications," *IEEE Internet of Things Journal*, vol. 8, no. 3, pp. 1322-1333, 2021.
- [3] V. Chamola, P. Kotes, A. Agarwal, G. N. Naren and M. Guizani, "A Comprehensive Review of Unmanned Aerial Vehicle Attacks and Neutralization Techniques," *Ad Hoc Networks*, vol. 111, p. 102324, 2021.
- [4] D. G. Davies, R. C. Bolam, Y. Vagapov and P. Excell, "Ultrasonic Sensor for UAV Flight Navigation", in *25th International Workshop on Electric Drives: Optimization in Control of Electric Drives (IWED)*, Moscow, Russia, 2018.
- [5] M. Rabah, A. H. Y. J. Rohan and S. H. Kim, "Design of fuzzy-PID controller for quadcopter trajectory-tracking," *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 18, no. 3, pp. 204--213, 2018.
- [6] C. Cadena, L. Carlone, H. Carrilo and Y. Latif, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1309-1332, 2016.
- [7] A. B. Z. N., S. Abdul-Rahman, S. Mutalib and M. Razif Shamsuddin, "Applying Graph-based SLAM Algorithm in a Simulated Environment," in *IOP Conference Series: Materials Science and Engineering*, Pahang, Malaysia, 2020.

- [8] F. J. Perez-Grau, F. Caballero, L. Merino and A. Viguria, "Multi-modal mapping and localization of unmanned aerial robots based on ultra-wideband and RGB-D sensing," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vancouver, BC, Canada, 2017.
- [9] C. Alex and A. Vijaychandra, "Autonomous cloud based drone system for disaster response and mitigation," in *International Conference on Robotics and Automation for Humanitarian Applications, RAHA*, 2016.
- [10] B. L. E. A. Balasuriya, B. A. H. Chathuranga, B. H. M. D. Jayasundara, N. R. A. C. Napagoda, S. P. Kumarawadu, D. P. Chandima and A. G. B. P. Jayasekara, "Outdoor robot navigation using Gmapping based SLAM algorithm," in *Moratuwa Engineering Research Conference (MERCCon)*, Moratuwa, Sri Lanka, 2016.
- [11] S. Kohlbrecher, J. Meyer, T. Graber, K. Petersen, U. Klingauf and O. Von Stryk, "Hector open source modules for autonomous mapping and navigation with rescue robots," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Berlin, Heidelberg, 2014.
- [12] J. F. Chow, B. B. Kocer, J. Henawy, G. Seet, Z. Li, W. Y. Yau and M. Pratama, "Toward Underground Localization: Lidar Inertial Odometry Enabled Aerial Robot Navigation," *CoRR*, 2019.
- [13] A. V. Kanhere and G. X. Gao, "LiDAR SLAM utilizing normal distribution transform and measurement consensus," in *Proceedings of the 32nd International Technical Meeting of the Satellite Division of the Institute of Navigation*, Miami, Florida, 2019.
- [14] J. S. J. Engel and D. Cremers, "Large-scale direct SLAM with stereo cameras," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Hamburg, Germany, 2015.
- [15] L. Messina, M. S., F. A. E., A. Massa and W. Matta, "Industrial Implementation and Performance Evaluation of LSD-SLAM and Map Filtering Algorithms for Obstacles Avoidance in a Cooperative Fleet of Unmanned Aerial Vehicles," in *2020 3rd International Conference on Intelligent Robotic and Control Engineering (IRCE)*, Oxford, UK, 2020.
- [16] L. Messina, S. Mazzaro, A. E. Fiorilla, A. Massa and W. Matta, "Industrial Implementation and Performance Evaluation of LSD-SLAM and Map Filtering Algorithms for Obstacles Avoidance in a Cooperative Fleet of Unmanned Aerial Vehicles," in *3rd International Conference on Intelligent Robotics and Control Engineering*, Oxford, UK, 2020.
- [17] D. O. Dantas, S. R. B. d. Santos, F. A. M. Cappabianco and A. A. d. Neto, "Design of Automated Construction System for Modular Structures based on Parameterized Learning Automata," in *International Joint Conference on Neural Networks (IJCNN)*, Rio de Janeiro, Brazil, 2018.
- [18] N. Ragot, Khemmar, P. A. R., R. Rossi and J. Y. Ertaud, "Benchmark of visual SLAM algorithms: ORB-SLAM2 vs RTAB-map," in *2019 Eighth International Conference on Emerging Security Technologies (EST)*, Colchester, UK, 2019.
- [19] M. S. Amiri and R. Ramli, "Visual Navigation System for Autonomous Drone using Fiducial Marker Detection," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 9, pp. 683-690, 2022.
- [20] W. Norzam, H. Hawari and K. Kamarudin, "Analysis of Mobile Robot Indoor Mapping using GMapping Based SLAM with Different Parameter," in *IOP Conference Series: Materials Science and Engineering*, Pulau Pinang, Malaysia, 2019.
- [21] R. A. Persad and C. Armenakis, "Comparison Of 2d And 3d Approaches For The Alignment," in *he International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Bonn, Germany, 2017.