

MSOT-Q: Multi-Sensor Fusion SLAM with Object Detection Tracking for Quadruped Robots

Yiming Xu¹, Bin Li^{2*}, Shuhui Yang², Yumin Zhi², Rukun Li¹, Teng Chen³

¹School of Electronics and Control, Qilu University of Technology (Shandong Academy of Sciences), Jinan, 250353, China.

²School of Mathematics and Statistics, Qilu University of Technology (Shandong Academy of Sciences), Jinan, 250353, China.

³School of Control Science and Engineering, Shandong University, Jinan, 250061, China

*Corresponding author(s).E-mail(s): ribbonlee@126.com

Contributing authors: xym1260159701@126.com; yangshuhui1718@163.com; zhiyumin07@163.com; backpacker123@126.com; chenteng100@sdu.edu.cn

Abstract—In recent years, unmanned robots, particularly quadruped robots, have attracted considerable attention owing to their remarkable mobility and payload capabilities. This paper introduces a system to enhance the autonomous perception of quadruped robots through object detection and map building. The primary goal is to improve target tracking and map construction capabilities. To enhance navigation proficiency in challenging environments, we employ a hierarchical controller integrating a proportional derivative control scheme with the model predictive control algorithm based on differential evolution. This controller is strategically crafted to navigate challenging environments effectively. The system underscores the fusion of data from various sensors for simultaneous localization and mapping applications, leading to precise map generation and accurate target tracking. Notably, the dynamic window approach algorithm is utilized to determine the optimal trajectory for target tracking, achieving a harmonious balance between traversability and target localization. Rigorous testing in a demanding simulation environment affirms the effectiveness of the proposed system in significantly improving both target tracking and map construction for the quadruped robot.

Keywords—Quadruped robot, SLAM, Object detection, Navigable paths

I. INTRODUCTION

In recent years, quadruped robots have emerged as up-and-coming platforms with exceptional mobility and payload capacities for various applications. However, standalone object detection methods on quadruped robots face inherent limitations, potentially leading to issues related to speed and accuracy.

In navigating unfamiliar environments, quadruped robots encounter numerous challenges that demand advanced simultaneous localization and mapping (SLAM) methods or fusion approaches. These difficulties include negotiating diverse obstacles, navigating irregular terrains and adapting to dynamic elements. The robot must exhibit real time perceptual acuity and a nuanced understanding of its surroundings to avoid collisions and ensure safe traversal. It is crucial to achieve precise localization and mapping for effective navigation. The lack of pre-existing cartographic data necessitates the generation of real time maps and accurately pinpointing the robot coordinates.

This is vital for preventing disorientation, devising strategic navigational routes and executing tasks meticulously. When applied to the quadruped robot, which often experiences intense vibrations, substantial drift errors and inaccurate positioning, these challenges underscore the need to integrate various sensors for improved positioning and perception accuracy.

To overcome these challenges, we propose “MSOT-Q: Multi-sensor fusion SLAM with Object detection Tracking for Quadruped robots,” an innovative framework that leverages the fusion of SLAM with real time object detection. This integration is complemented by advanced trajectory selection techniques, aiming to enhance the perception and decision-making capabilities of the quadruped robot. The goal is to enable the robot to navigate and interact intelligently in complex and dynamic environments.

The proposed system builds upon the methodology outlined in TROT-Q [1], strongly emphasizing generating traversable trajectories by fusing data from diverse sensors in SLAM and state estimation based on object detection. This paper introduces a hierarchical control framework, incorporating model predictive control (MPC) for trajectory tracking. Refining trajectories to align with computational constraints on the onboard computer facilitates the integration of simultaneous localization and mapping (SLAM) techniques. This integration, coupled with lightweight object detection methods [2], enables practical task applications within the limitations of onboard computing resources. To optimize target tracking, a persistent integration of the dynamic window approach (DWA) algorithm is employed and harmonized with MPC and PD control [3]. The main contributions of this paper are as follows:

- (1). Enhanced Object Detection Precision: By integrating SLAM with real time object detection, our framework can meet the quadruped robot object detection and map building task. This enhancement is crucial for tasks that require precise identification and tracking of objects in dynamic environments.
- (2). Efficient Path Planning and Navigation: By utilizing DWA, MPC based on differential evolution (DE) and PD control for leg motion, the robot efficiently traverses complex terrains. This integrated approach streamlines path planning while enabling the robot to make informed decisions based on its spatial understanding of the environment. Through the synergistic application of

This work was supported by the Colleges and Universities Twenty Terms Foundation of Jinan City (2021GXRC100) and the Development Plan of Young Innovation Team in Colleges and Universities of Shandong Province (2019KJN011).

* Corresponding author Bin Li.

these techniques, the system strikes a significant balance between traversability and computational efficiency.

The rest of this paper is arranged as follows. Section II briefly reviews related works, while Section III describes the proposed system in detail. The validation of the proposed system through simulations is covered in Section IV, followed by the conclusion and future works in Section V.

II. RELATED WORKS

To solve the disadvantages of the single sensor of SLAM, the multi-sensor fusion SLAM technology is introduced in this paper, aiming to enhance localization and navigation performance across various terrains and conditions.

Multi-sensor fusion SLAM algorithms utilizing graph optimization have also found extensive application. The ORB-SLAM2 algorithm, presented by Mur-Artal et al. in 2017 [4], integrated multiple sensors such as camera and IMU to enhance the accuracy and robustness of SLAM. Additionally, LVI-SAM and VINS-Mono algorithms represent further instances of multi-sensor SLAM techniques. LVI-SAM [5] integrates LiDAR and camera and VINS-Mono [6] combines camera and IMU, respectively. Rahman proposed a keyframe-based multi-sensor fusion SLAM [7].

Existing works have proposed numerous methodologies for human tracking in ground robots. For instance, Gritti et al. [8] introduced a technique employing an equipped RGB-D camera to cluster human legs within the viewpoint. However, this approach focuses on human detection and presupposes fully traversable ground. To compensate for the limited field of view of the camera and to develop a more resilient target tracking system, an algorithm based on Ultra-Wideband (UWB) was suggested [9].

Leigh et al. [10] employed a 2D laser scanner for human tracking, leveraging confidence scores from a trained ensemble classification model. While the algorithm was successfully validated on both a wheelchair robot and a mobile robot, it was exclusively tested on flat or gently sloping ground. Li et al. proposed a strategy for obstacle avoidance and human tracking in quadruped robots, utilizing an ultrawide-band three-dimensional LiDAR [11]. Furthermore, Sutyasadi et al. [12] introduced a robust leg control system using differential evolution for effective tracking.

Walking robots, exemplified by the quadruped robot, can navigate diverse terrains with varying elevations in contradistinction to wheeled mobile robots. Nevertheless, walking robots necessitate considering contact constraints, denoted as traversability, distinct from the considerations applicable to aerial robots [13]. Given these intricate conditions, executing tasks while accounting for the traversability of the quadruped robot continues to present a noteworthy challenge.

In the realm of the quadruped robot, continuous advancements in SLAM technology are propelled by noteworthy studies. The fusion of LiDAR and IMU sensors by Zhang et al. was employed to achieve localization in challenging terrains for the quadruped robot [14]. Furthermore, Dudzik et al. have introduced a robust visual-inertial SLAM system tailored for quadruped robots to enhance navigation performance [15].

Despite the significant progress brought by these studies to the perception and navigation capabilities of quadruped robots, challenges such as intense vibrations, significant drift errors and inaccurate positioning persist. This paper introduces MOST-Q, which integrates SLAM with object detection and tracking technologies to address these limitations. The purpose of MOST-Q is to elevate the real time perception and navigation capabilities of quadruped robots in complex environments. The design of MOST-Q aims to provide a comprehensive solution for quadruped robots navigating unknown or challenging environments, enhancing their accuracy in perception, target recognition and navigational flexibility, thereby advancing their agility and reliability in real world applications.

III. METHODS

The proposed quadruped robot environmental interaction system primarily comprises three components: SLAM map construction, target detection and reference trajectory planning, as illustrated in Fig. 1. Equipped with sensors in terms of LiDAR, RGB-D camera and IMU, the quadruped robot constructs a map in the unknown environment. Similarly, utilizing neural network-based object detection, the depth of the target within the detected bounding box is measured. Considering traversable regions, the optimal reference trajectory is selected based on the estimated states of both the robot and the target. Ultimately, the tracking controller is responsible for following the chosen reference trajectory.

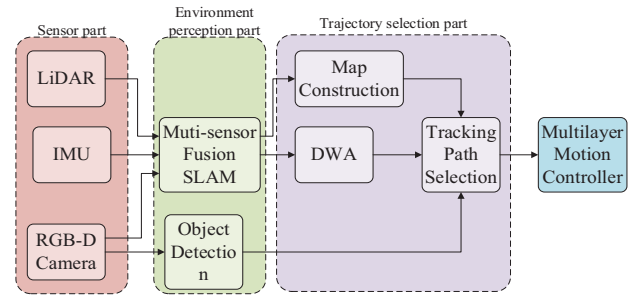


Fig. 1 The quadruped robot environmental interaction system

A. Map Construction

Compared with multi-sensor fusion SLAM, LiDAR SLAM and Visual SLAM are faster but less robust and accurate than multi-sensor fusion SLAM. Therefore, a multi-sensor fusion SLAM algorithm is recommended if a mapping task requires a highly precise and reliable map.

This paper proposes a multi-sensor SLAM method for environmental perception tasks in the quadruped robot. The method combines 3D LiDAR, stereo camera and IMU to achieve high-precision positioning and mapping in complex environments.

Two types of Odometry are developed and leveraging sensor information for enhancing perception performance, including LiDAR Odometry and Visual Odometry. Our method effectively combines different sensors strengths and complementary aspects to improve overall system performance and achieve impressive results.

The map construction process is illustrated in Fig. 2, comprising two Odometry: the LiDAR Odometry and the Visual Odometry. The LiDAR Odometry initially mitigates

motion distortions in LiDAR scans through backward propagation [16], followed by the computation of frame-to-map point-to-plane residuals. By transitioning from monocular to stereo Visual Odometry and incorporating depth information, the accuracy and reliability of Visual Odometry estimation are significantly improved. This leads to more robust feature tracking, depth estimation and map reconstruction, resulting in higher localization and mapping precision. The LiDAR point-to-plane [17] residuals and the image photometric errors are tightly integrated with IMU propagation within an error-state iterated Kalman filter. The integrated pose is then utilized to append new points to the global map.

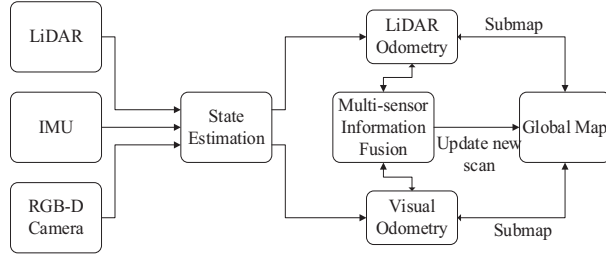


Fig. 2 Map construction process

a) LiDAR Odometry

The LiDAR Odometry, stemming from the multi-sensor fusion SLAM framework [18], encompasses all historical 3D points organized within an incremental k-d tree structure known as the ikd-Tree [19]. This tree structure facilitates efficient point inquiry insertions and deletions. Additionally, it internally down-samples the point cloud map at a specified resolution. It continuously monitors its tree structure and dynamically balances sub-trees when necessary. Upon receiving a new LiDAR scan, each point is queried and transformed with the predicted pose in the ikd-Tree.

Let P_{scan} represent a point from the new LiDAR scan and the transformation matrix obtained from the predicted pose in the ikd-Tree. The transformed point P_{global} in the global frame can be calculated as:

$$P_{global} = T_{ikd} \cdot P_{scan} \quad (1)$$

To identify the nearest points. Following the fusion of the scan with IMU data to obtain gives $\bar{x}_k$, it is employed to transform the scan points into the global frame, subsequently inserting them into the ikd-Tree at the rate of the LIO subsystem. This is done using an extended Kalman filter (EKF), which involves a prediction step and an update step and details are as follows.

Prediction Step:

$$\bar{x}_k = f(x_{k-1}, I_k, \Delta t) \quad (2)$$

Update Step:

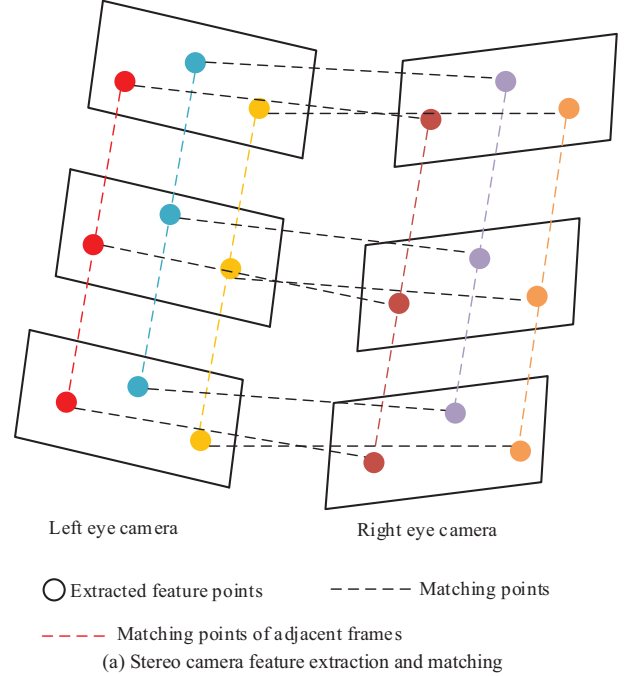
$$x_k = \bar{x}_k + K_k (y_k - h(\bar{x}_k)) \quad (3)$$

where x_{k-1} is the predicted state at the time k , f is the state transition function, I_k represents control inputs, Δt is the

time step, K_k is the Kalman gain, y_k is the measurement vector and h is the measurement function, respectively.

b) Visual Odometry

The generation of Visual Odometry relies on the seamless fusion of visual and IMU data. The features are extracted from the left and right stereo images using the oriented fast corner detector, which detects corners by comparing the grayscale values of pixels near the point to be detected. BRIEF binary descriptors [20] are employed for feature point extraction, where the value of each descriptor is determined by the relative magnitudes of the grayscale values for the surrounding pixels. The feature extraction model is depicted in Fig. 3.



(b) Feature point extraction results of the KITTI data set

Fig.3 Extraction and matching of ORB feature points

The above algorithm utilizes the oriented fast corner detector, which incorporates grayscale values and pixel orientations, enhancing corner detection accuracy and robustness compared to conventional methods. For feature point extraction, BRIEF binary descriptors are employed. BRIEF descriptors are binary codes based on pixel grayscale values, which are determined by comparing the grayscale values of neighboring pixels. Compared to traditional floating point descriptors, BRIEF descriptors require less storage and

computational complexity while maintaining good feature representation capabilities.

The efficient extraction of well-matched feature points from stereo images can be achieved by employing the oriented fast corner detector and BRIEF binary descriptors. These feature points can be utilized for localization and mapping in subsequent SLAM systems, thereby improving the accuracy and stability of the system. Moreover, the fast computation capability of the oriented fast corner detector and BRIEF binary descriptors makes them suitable for real time applications, meeting the real time requirements of SLAM systems.

The selection of ORB feature descriptors is motivated by their binary nature, which imparts them with rotational and scale invariance, as well as computational efficiency and robustness to noise and lighting variations. The computation of ORB feature descriptors involves Constructing a scale space pyramid and a difference of gaussian pyramid. Second, generating binary descriptors utilizing the BRIEF algorithm. The BRIEF algorithm is computed using the following formula:

$$BRIEF(x) = \begin{cases} 1 & \text{if } I_{p(x)} < I_{q(x)} \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

where $I_{p(x)}$ and $I_{q(x)}$ represent the brightness values of two points on the image and express their corresponding position, respectively.

The formula for computing ORB descriptors is as follows:

$$d(x) = [BRIEF(x_1), BRIEF(x_2), BRIEF(x_3), \dots, BRIEF(x_n)]^T \quad (5)$$

where $x_1, x_2, \dots, x_n$ represents the number of n sampled points around the corner. ORB descriptors use binary descriptors, where each element takes a value of 0 or 1.

In the proposed SLAM system, the fast corner detector detects key points in each image. It is then utilized to extract features by computing ORB descriptors and their corresponding feature descriptors. These results are in two sets of feature points:

$$\begin{aligned} F_L &= f_{L,1}, f_{L,2}, \dots, f_{L,n} \\ F_R &= f_{R,1}, f_{R,2}, \dots, f_{R,n} \end{aligned} \quad (6)$$

Regarding feature matching, the bag-of-words approach clusters feature descriptors and match key points by measuring their Hamming distances. For two key points, their distance is calculated using the following formula:

$$d_{(i,j)} = \sum_{k=0}^n \delta(b_i^k \neq b_j^k) \quad (7)$$

where b_i^k and b_j^k represents the k binary bit of descriptor d_i and d_j respectively and δ represents the Kronecker delta function. During the matching process, the descriptor distance ratio is used to determine whether two feature points match where d_1 and d_2 represent the distance between the point to be matched and its two nearest neighbors, respectively. If d_1/d_2 is less than a preset threshold, then the two feature

points are considered a match, resulting in a set of matched point pairs:

$$M = (f_{L,i}, f_{R,i})i = i^n \quad (8)$$

where $f_{L,i}$ and $f_{R,i}$ represents the corresponding feature point in the left and right stereo images.

In addition, optical flow is utilized to track the positional changes of matched points between adjacent frames, aiming to correct their positions and enhance the accuracy and robustness of the system. This is especially crucial in complex scenarios where the quadraped robot undergoes rapid movements, experiences strong vibrations, or encounters environments with low texture.

For every pair of matched points, the corresponding 3D point can be derived using the geometric constraints of the stereo camera. The calculation can be performed using the following formula:

$$P_i = \frac{B\xi}{d_i} (z_{L,i} - z_{R,i}, v_{L,i} - v_{R,i}, 1)^T \quad (9)$$

where B represents the baseline length of the stereo camera, ξ represents the focal length of the camera, d_i represents the disparity of the matched point pair $f_{L,i}, f_{R,i}$, $z_{L,i}$ and represents the pixel coordinates of the matched point in the left camera and represents the pixel coordinates of the matched point in the right camera respectively. Through the stereo camera, the pixel coordinates and disparity information of the corresponding point can be obtained in the left and right cameras. We can convert this information into three dimensional coordinates in the point cloud using the triangulation formula. The resulting point cloud can represent the structure of the scene and provide vital information for subsequent pose estimation and map construction.

c) Multi-sensor Information fusion

To better integrate the two types of Odometry information. This paper introduces a novel framework that combines LIO-SAM and LVI-SAM to improve the SLAM performance in challenging environments. The Odometry data from two subsystems is integrated into a factor graph framework, which models the relationships between robot state variables and sensor measurements. The proposed framework effectively fuses two Odometry information through factor graph optimization, resulting in accurate robot state and trajectory estimation. The algorithm incorporates 3D LiDAR, stereo camera and IMU data, utilizing factor graph optimization to capture complex variable relationships. Fig. 4 illustrates the architecture of the factor graph framework.

The proposed factor framework achieves integration and mutual utilization of visual, LiDAR and IMU data in each subsystem. Visual Odometry leverages stereo images to estimate relative motion. LiDAR Odometry utilizes LiDAR data for point cloud registration and motion estimation, these subsystems exchange information and contribute to the overall SLI-SLAM algorithm. The framework successfully fuses sensor data by formulating an optimization problem that minimizes a cost function and considers errors related to feature extraction, data association and other factors. The fusion allows for the integration of various sources of

information, providing a more comprehensive and accurate representation of the environment. This integration allows for precise and robust localization and mapping in complex environments.

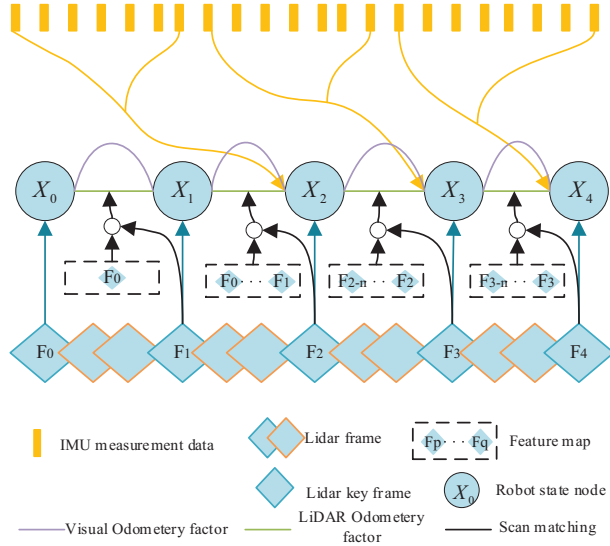


Fig. 4 The multi-sensor fusion system framework (four state nodes are selected as examples)

Specifically, the proposed factor graph framework employs a factor graph optimization algorithm to fuse sensor data and minimize a cost function. The cost function is formulated as follows:

$$E(X_r, S) = -\log P(X_r | S) \quad (10)$$

This cost function captures the relationship between the quadruped robot state X_r and the environment map S . By converting the sensor data into a factor graph representation, the framework leverages the factor graph optimization algorithm to estimate the state of the quadruped robot and the map, thereby achieving sensor fusion.

B. Object Detection

To measure the target information (location, size, credibility) $p \in R^3$, the YOLOv4-tiny network [21] is trained and inferred on the obtained RGB image. A bounding box is acquired and the point cloud of the target inside the box can be obtained from the depth of the RGB-D camera. Here, we ignore the unnecessary ground point cloud within the box and calculate the centroid $\bar{p}_{target}$ of the point cloud. The centroid is approximately used as the target state p .

The centroid calculation can be expressed as:

$$\bar{p}_{target} = \frac{1}{N_1} \sum_{i=1}^N p_i \quad (11)$$

where N_1 is the number of points in the point cloud and p_i represents the i_{th} point in the point cloud. The detected target and the corresponding point clouds can be seen in Fig. 5.

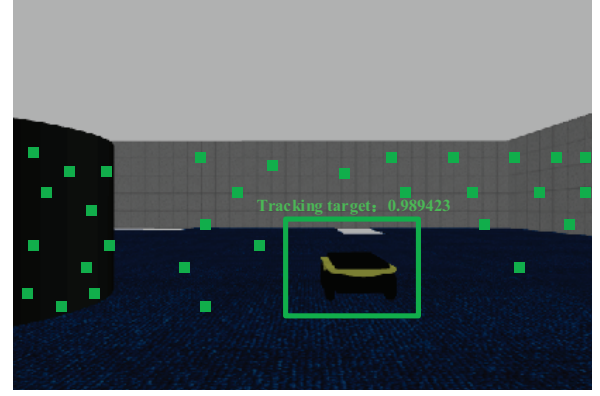


Fig. 5 Quadruped robot object detection display

C. Reference Trajectory Planner

Upon estimating the state of the robot, the collected point cloud data from the attached LiDAR and RGB-D camera is processed to construct a comprehensive 3D occupancy grid map. OctoMap [22] is employed with the DWA [23] to evaluate traversability and meticulously identify potential obstacles. This process involves assessing candidate precomputed trajectories, called peacock trajectories, to generate a score matrix. Leveraging the kinodynamics of the robot and exclusively focusing on the score matrix, the optimal local trajectory is readily determined based on a straightforward policy [24].

When facing tasks such as SLAM mapping, target detection and trajectory tracking for quadruped robots, selecting an appropriate path planning algorithm is paramount. Compared to QR-SCAN, DWA exhibits advantages in real time performance, consideration of robot dynamics and adaptability to dynamic environments. Therefore, DWA suits scenarios demanding high real time performance, prominent robot dynamic constraints and limited team resources.

During the practical implementation, we encountered specific challenges that significantly influenced the choice of our path planning algorithm. Specifically, when employing the QR-SCAN algorithm, we observed the following issues:

- (1). Insufficient Real Time Performance: In some cases, the real time responsiveness of the QR-SCAN algorithm is low and its computation time is extended, making it unsuitable for situations that require fast response and dynamic path adjustment.
- (2). Stability in Dynamic Environments: QR-SCAN exhibits suboptimal stability when confronted with dynamic environmental conditions, demonstrating susceptibility to interference from moving objects, ultimately compromising the accuracy of path planning.
- (3). Local Optima Convergence: In complex environments, QR-SCAN occasionally becomes trapped in local optima, hindering the attainment of the optimal path planning solution.

Given these considerations, we adopt the DWA as our chosen path planning methodology. In contrast to QR-SCAN, DWA boasts superior attributes in terms of real time responsiveness, adaptability to dynamic environments and

consideration of robot dynamics, aligning more effectively with practical requirements.

This study seamlessly integrates the DWA algorithm into the existing framework. Additionally, a supplementary score matrix is introduced by computing the Euclidean distances between the measured target state and the terminal points of the peacock trajectory. Equation (13) is then used with the DWA algorithm to identify the trajectory with the highest score. This ensures that the selected trajectory provides a clear, obstacle-free path to the target position.

The Hadamard product $(\circ)H$ of two matrices A and B , is defined as:

$$H = A \circ B \quad (12)$$

By combining the score matrix M and the distance matrix D through the Hadamard product, the combined score matrix S can be obtained:

$$S = M \circ D \quad (13)$$

The trajectory with the highest score S is selected as the optimal local trajectory for the robot. The trajectory of the target robot and the quadruped robot in the simulation environment is shown in Fig. 6.

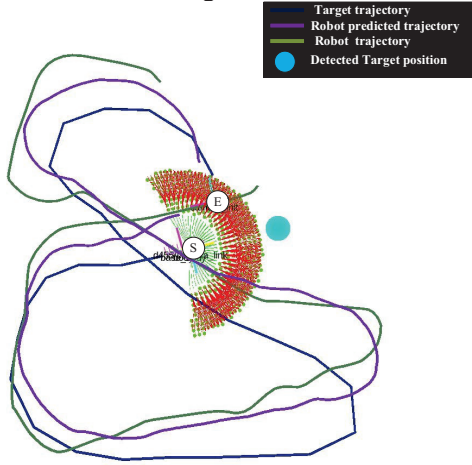


Fig. 6 Quadruped robot tracking path of the target robot path. S is the initial position and E is the end position

Based on the results illustrated in Fig. 6, it is evident that the quadruped robot can effectively detect target and evade obstacles while maneuvering around impediments. This is achieved through a combination of SLAM and tracking algorithms. In cases where the robot deviates from its tracking objective, it possesses the capability to navigate around obstacles and reestablish target traces through the utilization of the SLAM algorithm.

To maintain continuity with the established methodology from earlier studies, only the initial step of the best trajectory is tracked, thereby minimizing the potential for abrupt disruptions due to sudden obstacles or openings. The process of trajectory selection is then iterated to maintain an optimal path consistently.

For enhanced clarity, the process of reference trajectory selection is visually represented in Fig. 7, providing a tangible illustration of the trajectory assessment process. It also draws inspiration from the hierarchical controller approach outlined in reference [24] for improved motion control.

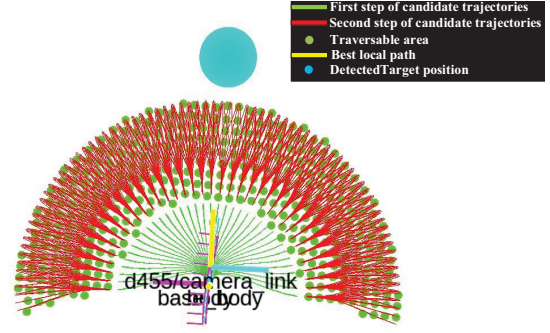


Fig. 7 Illustrates the various path options that the quadruped robot encounters during path planning

Based on the reference [25], we design a tracking controller incorporating advanced speed MPC and low-level leg motion PD controller to track the selected optimal trajectory from the results based on the DWA algorithm. Since the low-level PD controller receives linear velocity and angular velocity, the system dynamics of the high-level velocity MPC can be described as follows:

$$\dot{X} = f(x, u) = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{cases} v \cos \theta - s \sin \theta \\ v \sin \theta + s \cos \theta \\ \omega \end{cases} \quad (14)$$

where x represents the state vector and $u = [v, s, \omega]^T$ represents the input vector with linear velocity in the body-fixed x-axis v , linear velocity in the body-fixed y-axis s and angular velocity in the body-fixed z-axis ω , respectively.

Since the dynamics of the system are nonlinear, we discretize the system with a time interval Δt as follows:

$$\begin{aligned} x_{k+1} &= x_k + v_k \cos \theta_k \Delta t \\ y_{k+1} &= y_k + v_k \sin \theta_k \Delta t \\ \theta_{k+1} &= \theta_k + \omega_k \Delta t \end{aligned} \quad (15)$$

where k represents every discrete time instant. At every discrete time moment, the control policy is defined by solving the following optimal control problem:

$$u^* = \arg \min_u J$$

$$J = \sum_{k=0}^N (x_{ref,k} - x_k)^T w_1 (x_{ref,k} - x_k) + (u_{ref,k} - u_k)^T \quad (16)$$

$$w_2 (u_{ref,k} - u_k) + (u_{ref,k} - u_k)^T w_3 (u_{ref,k} - u_k)$$

Subject to:

$$x_{k+1} = Cx_k + Du_k$$

$$C = \begin{bmatrix} 1 & 0 & -v_k \sin \theta_k \Delta t \\ 0 & 1 & v_k \cos \theta_k \Delta t \\ 0 & 0 & 1 \end{bmatrix} \quad (17)$$

$$D = \begin{bmatrix} \cos \theta_k \Delta t & 0 \\ \sin \theta_k \Delta t & 0 \\ 0 & \Delta t \end{bmatrix}$$

$$u_{\min} \leq u_k \leq u_{\max}$$

where J represents the cost function, $x_{ref,k}$ represents the reference trajectory, N is the prediction horizon of MPC and w_1, w_2, w_3 are positive semidefinite matrices for weighing

corresponding cost function terms. Especially, w_3 is adopted here to consider the smoothness of the inputs.

To solve the optimization problem at each discrete time instant, we adopt the DE algorithm, a population-based optimization algorithm that has proven its efficacy in solving nonlinear optimization problems. In contrast to other optimization algorithms like SLSQP, the DE algorithm searches for the optimal solution by performing crossover and mutation operations on individuals in the population.

DE algorithm outperforms the SLSQP algorithm in searching for the global optimum solution for high dimensional, nonlinear and nonconvex optimization problems. Additionally, the computational cost of the DE algorithm is relatively small, making it suitable for real time control. The optimal input o_i^* obtained from the optimization by DE is then used as input to the low-level PD controller, which generates the leg motion, drives the torques and forces the quadruped robot to track the reference trajectory. The optimization can be solved within 10ms using the SciPy library [26]. Overall, using the DE algorithm in our optimization process has enabled our system to achieve efficient optimization with good global search capability and convergence performance, making it an ideal approach for our application needs.

IV. EXPERIMENTS AND RESULTS ANALYSIS

We conduct simulation-based experiments using the ROS platform to verify the performance of our proposed multi-sensor fusion SLAM system with target detection and tracking capabilities. We use the Gazebo simulator in an environment full of various challenges, including walls, block obstacles, etc., as shown in Fig. 8(a). In this simulation environment, we use the A1 quadruped robot model in Fig. 8(b), equipped with 3D LiDAR and RGB-D cameras. LiDAR data streams and RGB-D camera feeds are acquired at 10Hz and 30Hz, respectively. The SLAM framework and the object detection and tracking module work simultaneously at 10Hz and 30Hz.

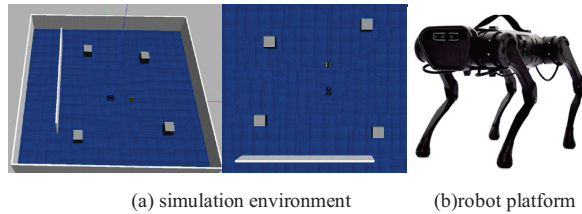


Fig. 8 Gazebo simulation environment

Illustrated in Fig. 9, the simulation results offer valuable insights into the performance of our system, with particular emphasis on the SLAM component. It is worth noting that the quadruped robot can track the target robot well and complete the drawing task smoothly. In Fig. 9, we can observe the performance of the quadruped robot. It follows the target robot precisely, always maintaining a close connection with the target. Whether the target robot changes its speed and direction or abruptly alters its path, the quadruped robot reacts swiftly, maintaining a stable tracking state showcasing its agility and responsiveness. Additionally, the green and purple trajectories showcase the motion control and path planning capabilities of the quadruped robot throughout the process. It can adjust its motion trajectory dynamically to ensure a favorable relative position with the target robot.

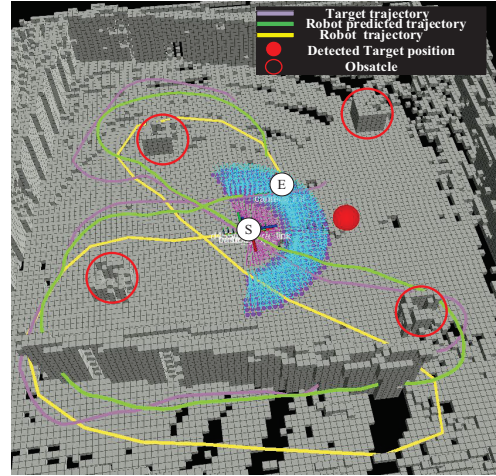


Fig. 9 Quadruped robot mapping results tracking target robot

Fig. 10 encapsulates PCL representation, underscoring the remarkable performance of our system. We skillfully maneuver the target robot with a joystick, navigating it to a point where the visual tracking of the quadruped robot experiences a momentary disruption. The depicted trajectory eloquently showcases the quadruped robot autonomous obstacle avoidance capabilities. Once visual contact is reestablished and the target is reacquired, we seamlessly transition back to manual control of the target robot. This orchestrated interplay enables us to execute target tracking and mapping functionalities masterfully.

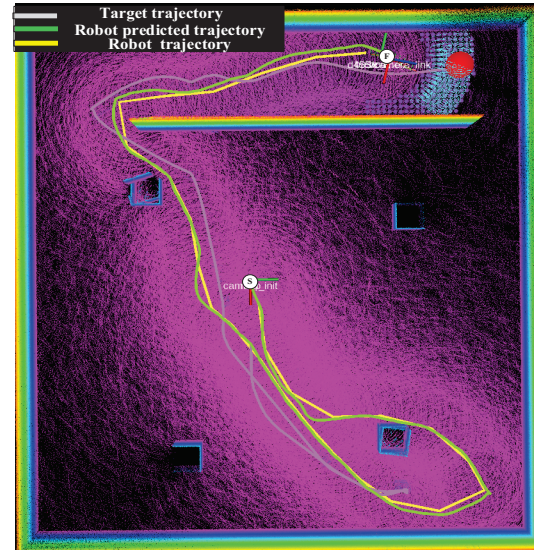


Fig. 10 PCL illustration - quadruped robot target tracking and mapping demonstration

To evaluate the accuracy of TP-SLAM (the proposed SLAM) algorithm and mainstream algorithms, we employed EVO (Evaluation of Odometry and SLAM Systems) and compiled the APE (Absolute Pose Error) comparison in Table 1. The results indicate that the proposed experimental algorithm outperforms other algorithms in terms of accuracy. This signifies the robustness and reliability of the proposed experimental algorithm in achieving precise localization and mapping. The superiority of the proposed experimental algorithm further establishes its position as a leading algorithm in the field, offering enhanced performance and accuracy for various applications.

Table 1: Comparison of SLAM algorithms based on KITTI

Alogrithm	MEAN	MEDIAN	MIN	RMSE
TP-SLAM	7.551	7.790	3.211	7.8039
FAST-LIO2	9.598	9.787	3.058	10.151
ORB-SLAM3	8.171	7.756	2.491	8.580

In challenging environmental conditions, the quadruped robot demonstrates satisfactory tracking capabilities, successfully tracking the target for nearly three minutes. It navigates through potential obstacles with reasonable proficiency, indicating a commendable level of reliability in practical scenarios.

V. CONCLUSION

This paper presents an innovative and safe target tracking system for quadruped robots, utilizing the power of the traversable trajectory exploration method and hierarchical tracking controller. With the aid of Muti-sensor fusion SLAM technology and neural network-based object detection, we accurately estimate the states of the robot and the target. These estimated states and sensor data are carefully processed to draw the optimal trajectory, balancing the considerations of traversability and target localization.

To validate the effectiveness of our proposed system, we conduct rigorous testing in a challenging simulation environment. The comprehensive integration of SLAM, MPC, PD control, DE algorithm, DWA and real time object detection underscores the system robustness in handling complex and dynamic environments. While the practical application of the proposed system in real world scenarios remains a prospect for future exploration, we anticipate promising outcomes, mainly focusing on further refining the integration of a robust global planner to reduce the risk of target loss.

REFERENCES

- [1] E. M. Lee, J. Jeon and H. Myung, "TROT-Q: Traversability and Obstacle Aware Target Tracking System for Quadruped Robot," 2022 13th Asian Control Conference (ASCC), Jeju, Korea, pp.2480-2484, 2022.
- [2] R. Jadhav, R. Patil, A. Diwan, S. M. Rathod and M. Inamdar, "Aerial Object Detection and Tracking Using Yolov4 and Deepsort," 2022 International Conference on Industry 4.0 Technology (I4Tech), Pune, India, pp.1-6, 2022.
- [3] M. H. T, R. P, R. M, and Saniya, "Trajectory Tracking and Lane-Keeping Assistance for Autonomous Systems Using Pid and MPC Controllers," 2023 International Conference on Smart Systems for applications in Electrical Sciences (ICSSSES), Tumakuru, India, pp.1-7, 2023.
- [4] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras," IEEE Transactions on Robotics, vol.33, no.5, pp.1255-1262, 2017.
- [5] T. Shan, B. Englot, C. Ratti and D. Rus, "LVI-SAM: Tightly-coupled Lidar-Visual-Inertial Odometry via Smoothing and Mapping," 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, pp.5692-5698, 2021.
- [6] T. Qin, P. Li, and S. Shen, "VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator," IEEE Transactions on Robotics, vol.34, no.4, pp.1004-1020, 2018.
- [7] Rahman S., Quattrini Li A., Rekleitis I., "Svin2: A Multi-Sensor Fusion-Based Underwater SLAM System," International Journal of Robotics Research, vol.41, no.11, pp.1022-1042, 2022.
- [8] A. P. Gritti, O. Tarabini, J. Guzzi, G. A. Di Caro, V. Caglioti, L. M.Gambardella, and A. Giusti, "Kinect-Based People Detection and Tracking from Small-Footprint Ground Robot," IEEE/RSJ International Conference on Intelligent Robot and Systems (IROS), pp.4096-4103, 2014.
- [9] T. Feng, Y. Yu, L. Wu, Y. Bai, Z. Xiao, and Z. Lu, "A Human Tracking Robot Using Ultra-Wideband Technology," IEEE Access, vol.6, pp.42541-42550, 2018.
- [10] A. Leigh, J. Pineau, N. Olmedo, and H. Zhang, "Person Tracking and Following With 2D Laser Scanners," IEEE International Conference on Robotics and Automation (ICRA), pp.726-733, 2015.
- [11] Li Z., Li B., Liang Q., Liu W., Hou L., and Rong X. "A Quadruped Robot Obstacle Avoidance and Personnel Following Strategy Based on Ultra-Wideband and Three-Dimensional Laser Radar." International Journal of Advanced Robotic Systems, vol.19, no.4, 2022.
- [12] Sutyasadi P., Parnichkun M., "Gait Tracking Control of Quadruped Robot Using Differential Evolution Based Structure Specified Mixed Sensitivity H ∞ Robust Control," Journal of Control Science and Engineering, pp.1-18, 2016.
- [13] E. M. Lee, D. Seo, J. Jeon, and H. Myung, "QR-SCAN: Traversable Region Scan for Quadruped Robot Exploration Using Lightweight Precomputed Trajectory," International Conference on Control, Automation and Systems (ICCAS), pp.957-961, 2021.
- [14] Zhang C., Yang Z., Fang Q., Xu C., Xu H., Xu X., and Zhang J., "FRL-SLAM: A Fast, Robust and Lightweight SLAM System for Quadruped Robot Navigation," 2021 IEEE International Conference on Robotics and Biomimetics (ROBIO), Sanya, China, pp.1165-1170, 2021.
- [15] Dudzik T., Chignoli M., Bledt G., Lim B., Miller A., Kim D., and Kim S., "Robust Autonomous Navigation of A Small-Scale Quadruped Robot In Real-World Environments," 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, pp.3664-3671, 2020.
- [16] W. Xu and F. Zhang, "FAST-LIO: A Fast, Robust LiDAR-Inertial Odometry Package by Tightly-Coupled Iterated Kalman Filter," IEEE Robotics and Automation Letters, vol.6, no.2, pp.3317-3324, 2021.
- [17] Xu G., Du S., Cui D., Zhang S., Chen B., Zhang X., Xue J., Gao Y., "Precise Point Set Registration Using Point-to-Plane Distance and Correntropy for LiDAR Based Localization," 2018 IEEE Intelligent Vehicles Symposium (IV), Changshu, China, pp.734-739, 2018.
- [18] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "FAST-LIO2: Fast Direct LiDAR-Inertial Odometry," IEEE Transactions on Robotics, vol.38, no.4, pp.2053-2073, 2022.
- [19] Y. Cai, W. Xu, and F. Zhang, "Ikd-Tree: An Incremental Kd Tree for Robotic Applications," arXiv preprint arXiv:2102.10808, 2021.
- [20] Calonder, M., Lepetit, V., Strecha, C., Fua, P., "Brief: Binary Robust Independent Elementary Features," Lecture Notes in Computer Science, pp.778-792, 2010.
- [21] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal Speed and Accuracy of Object Detection," arXiv preprint arXiv:2004.10934, 2020.
- [22] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "Octomap: An Efficient Probabilistic 3D Mapping Framework Based on Octrees," Autonomous robot, vol.34, no.3, pp.189-206, 2013.
- [23] X. Li, X. Hu, Z. Wang, and Z. Du, "Path Planning Based on Combination of Improved A-STAR Algorithm and DWA Algorithm," 2020 2nd International Conference on Artificial Intelligence and Advanced Manufacture (AIAM), Manchester, United Kingdom, pp.99-103, 2020.
- [24] D. Kim, D. Carballo, J. Di Carlo, B. Katz, G. Bledt, B. Lim, and S. Kim, "Vision Aided Dynamic Exploration of Unstructured Terrain with A Small-Scale Quadruped Robot," IEEE International Conference on Robotics and Automation (ICRA), pp.2464-2470, 2020.
- [25] E. M. Lee, D. Choi, and H. Myung, "Peacock Exploration: A Lightweight Exploration for Uav Using Control-Efficient Trajectory," Robot Intelligence Technology and Applications (RiTA) 2020. Springer, pp.136-146, 2021.
- [26] Virtanen P., Gommers R., Oliphant T. E., et al., "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," Nature Methods, vol.17, no.3, pp.261-272, 2020.