

Adap-RPF: Adaptive Trajectory Sampling for Robot Person Following in Dynamic Crowded Environments

Weixi Situ^{1*}, Hanjing Ye^{1*}, Jianwei Peng², Yu Zhan¹, and Hong Zhang^{1†}, *Life Fellow, IEEE*

Abstract—Robot person following (RPF) is a core capability in human–robot interaction, enabling robots to assist users in daily activities, collaborative work, and other service scenarios. However, achieving practical RPF remains challenging due to frequent occlusions, particularly in dynamic and crowded environments. Existing approaches often rely on fixed-point following or sparse candidate-point selection with oversimplified heuristics, which cannot adequately handle complex occlusions caused by moving obstacles such as pedestrians. To address these limitations, we propose an adaptive trajectory sampling method that generates dense candidate points within socially aware zones and evaluates them using a multi-objective cost function. Based on the optimal point, a person-following trajectory is estimated relative to the predicted motion of the target. We further design a prediction-aware model predictive path integral (MPPI) controller that simultaneously tracks this trajectory and proactively avoids collisions using predicted pedestrian motions. Extensive experiments show that our method outperforms state-of-the-art baselines in smoothness, safety, robustness, and human comfort, with its effectiveness further demonstrated on a mobile robot in real-world scenarios.

I. INTRODUCTION

Robot person following (RPF) is a fundamental capability in human–robot interaction (HRI), enabling a wide range of applications [1]–[3]. However, existing methods based on fixed-point following or sparse candidate selection with simplified heuristics struggle to cope with frequent occlusions of the target by static objects (e.g., walls or furniture) or dynamic obstacles (e.g., other pedestrians), posing significant challenges for deploying RPF in real-world environments.

Conventional RPF methods adopt a fixed-point following strategy, tracking a target point that maintains a constant relative distance and orientation to the target. The relative orientation is typically behind [4]–[6], ahead [7], [8], or side by side [9], [10], under the assumption that the target remains continuously visible. In complex and crowded environments, this assumption often breaks down, as rigidly maintaining a fixed relative position can cause the robot to lose sight of the target person when nearby people or objects temporarily obstruct the view, or render the target point unreachable when

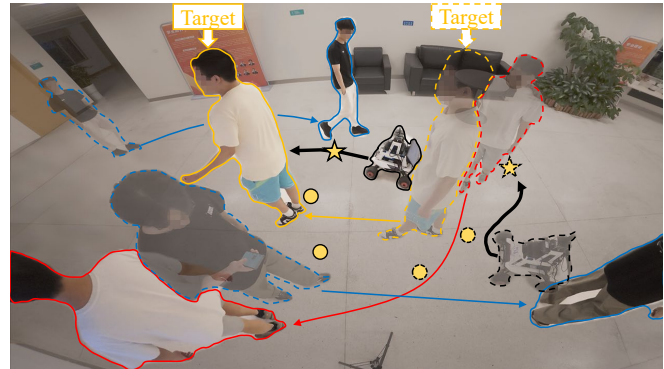


Fig. 1. Adap-RPF in a real-world dynamic crowded environment. The robot proactively avoids dynamic occlusions (pedestrian outlined in red) by adaptively sampling trajectory based on predicted human motion (colored arrows). Dashed outlines denote agent positions at time T_0 , while solid outlines denote their positions at time T_1 . The golden star marks the selected optimal following point, and golden circles denote the candidate points.

it is occupied, leading to frequent occlusions and tracking failures. To address occlusions, some methods [11], [12] predict potential search regions to reacquire the target person. However, these reactive strategies can induce oscillatory behavior when the search space is constrained by surrounding pedestrians.

An alternative is proactive planning to prevent occlusion and potential collisions. A common strategy [13], [14] involves: i) pre-defining several candidate following points around the target, and ii) dynamically switching among them based on the situation. These approaches establish a small set of candidate points or regions around the target and dynamically switch among them according to heuristic rules, enabling proactive planning to avoid obstacles and potential collisions while maintaining continuous tracking. Although these methods exemplify the general *sample–evaluate–plan* paradigm, they tend to oversimplify both candidate point selection and evaluation, often assuming back-following as the safest option and neglecting dynamic occlusions. Consequently, sparse sampling with simplified heuristics limits their effectiveness in dynamic, crowded environments where occlusions and collisions are prevalent.

Recently, several methods [15]–[18] have been proposed to directly estimate the next robot action through a unified model, based on either deep-learning [16], [19] or optimization [15], [17], [18]. Deep-learning-based approaches often suffer from poor generalization and limited interpretability, reducing their effectiveness in unseen scenarios. In contrast, optimization-based methods formulate the problem by integrating multiple constraints, such as target visibility and collision avoidance, into a unified optimization framework

*Equal contribution.

†Corresponding author: Hong Zhang hzhang@sustech.edu.cn.

¹W. Situ, H. Ye, Y. Zhan and H. Zhang are with the Shenzhen Key Laboratory of Robotics and Computer Vision and the Department of Electronic and Electrical Engineering, Southern University of Science and Technology (SUSTech), Shenzhen, China. situwx2024@mail.sustech.edu.cn.

²J. Peng is with SUSTech.

This work was supported in part by Shenzhen Science and Technology Program (No. SGDX20240115111759002), in part by Meituan Academy of Robotics Shenzhen, in part by the Shenzhen Association for Science and Technology (No. XHXS2025-003), and in part by High level of special funds (G03034K003) from SUSTech, Shenzhen, China.

[18]. However, this often sacrifices proxemic comfort, leading to overly distant following behaviors. Moreover, their formulation leverages only the predicted target trajectory for visibility optimization, while neglecting the predicted motions of surrounding pedestrians for proactive collision avoidance, limiting adaptability in dynamic environments.

To address these limitations, we propose **Adap-RPF**, a hierarchical planning framework that proactively integrates target visibility, collision avoidance, human comfort and motion smoothness. Unlike fixed-point following or sparse candidate-point selection methods [13], [14], our approach performs target-centric adaptive trajectory sampling over a dense set of candidate points generated along the predicted target motion within a semi-annular region respecting the target’s personal and social zones. Candidates are evaluated through a multi-objective cost function considering both prediction-aware factors (e.g., occlusion, proximity) and observation-based factors (e.g., distance, travel, stickiness). The optimal point is then extrapolated into a following trajectory, conditioned on the target’s predicted motion. Building on this trajectory, we further develop a prediction-aware model predictive path integral (MPPI) controller that tracks the reference trajectory while proactively avoiding collisions based on predicted pedestrian motions. This predictive integration into an MPC-style controller is inspired by related work in crowd navigation [20], [21], but represents the first attempt in the context of RPF. Experimental results validate that our framework is both practical and efficient, enabling smoother and safer person-following behaviors compared with state-of-the-art (SOTA) baselines, even in highly dynamic environments. Moreover, we demonstrate the real-world applicability of our method through experiments on a mobile robot, as shown in Fig. 1.

II. RELATED WORK

This section presents a literature review on existing RPF approaches, with particular attention to their proactive strategies for dealing with occlusion and their potential incorporation of collision avoidance functions.

A. Robot Person Following with Multiple Predefined Points

Some methods define multiple candidate following points around the target and switch among them based on heuristic rules. For example, Hoeller *et al.* [13] discretize a semi-circular region behind the target into five candidate points and prefer the point without collision risk, treating back-following as the preferred strategy. However, this method simplifies point sampling and evaluation, leaving the method ill-suited for crowded environments where dynamic occlusions and collision risks occur frequently. Yao *et al.* [14] switches between rear and side positions based on static obstacle detection and corridor geometry, while Vu *et al.* [22] models an elliptical region around the target’s motion and selects feasible points according to obstacle presence and distance. These methods address static occlusion based on static obstacle or environmental structure recognition, which

is not generalized to dynamic environments where frequent and dynamic occlusions occur.

To address these challenges, we employ a denser point-sampling strategy combined with a multi-objective evaluation framework that jointly considers proximity constraints [23], target visibility, collision avoidance, and motion smoothness. The selected optimal point is then extended into a person-following trajectory by incorporating the predicted motion of the target. This trajectory-sampling scheme is both computationally efficient and adaptive to dynamic environments.

B. Robot Person Following with A Unified Model

Recently, several approaches have explored unified models that directly generate the robot’s next action. Such methods can be broadly categorized into learning-based methods [16], [19] and optimization-based methods [15], [17], [18]. Learning-based methods, such as [16], train reinforcement learning agents on randomized human trajectories with reward functions encoding preferred distance and orientation, thereby enabling adaptive action selection. However, these approaches require extensive training, exhibit poor generalization, and offer limited interpretability, which restricts their effectiveness in unseen scenarios.

In contrast, optimization-based approaches explicitly incorporate task constraints into the planning formulation. For example, Repiso *et al.* [15] formulate multiple person-following costs within the anticipate kinodynamic planning framework, switching between rear and side positions in narrow passages. Yet, this approach relies on a global map and assumes the target’s global destination is known in advance, limiting its applicability in dynamic, unknown environments. Wang *et al.* [18] embed objectives such as target detectability, tracking accuracy, motion smoothness, and obstacle avoidance into a nonlinear MPC framework for multi-target following. Yet, their approach struggles to balance visibility with tracking and often overlooks proxemic comfort. Moreover, it generalizes poorly in highly dynamic settings, as it does not leverage future trajectories of surrounding pedestrians, leading to limited proactive collision avoidance.

By contrast, our Adap-RPF framework explicitly incorporates predicted trajectories of all observed pedestrians into the controller, enabling proactive and reliable collision avoidance. Furthermore, rather than embedding all RPF-related objectives directly into the controller optimization, we delegate most objectives to the adaptive trajectory sampling module. This decomposition reduces the computational burden of the controller and allows faster response to dynamic pedestrians.

III. METHODOLOGY

In this section, we present Adap-RPF, a hierarchical planning framework that leverages human trajectory prediction (Sec. III-A) to proactively address target visibility, proxemic comfort, collision avoidance, and motion smoothness. The key innovation is the adaptive trajectory sampling module

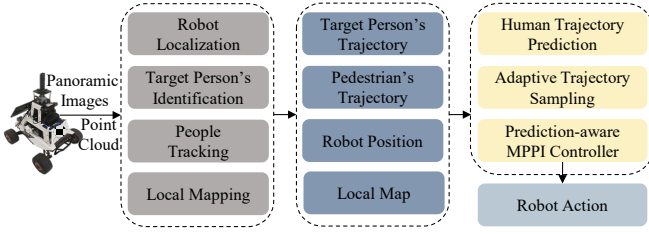


Fig. 2. **RPF system pipeline.** We integrate the proposed Adap-RPF framework into the system, where the light yellow modules indicate our contributions. Our framework consists of three components: human trajectory prediction, adaptive trajectory sampling and prediction-aware MPPI controller. Additional RPF system modules are adopted from work [11] and are not the focus of this paper. Overall, the proposed RPF system can locate, track, and follow a target person while proactively avoiding occlusions in dynamic, crowded environments.

(Sec. III-B), which generates a dense set of candidate following points within the target-centric social space, rather than relying on fixed or sparsely predefined samples. These candidates are then evaluated against predicted pedestrian trajectories to anticipate dynamic occlusions and potential collisions, with the optimal point selected to form a person-following trajectory aligned with the target’s predicted motion. We subsequently design a prediction-aware MPPI controller to track this reference trajectory (Sec. III-C), which incorporates predicted trajectories of surrounding pedestrians into its optimization process, thereby ensuring proactive and reliable collision avoidance.

The complete RPF framework is illustrated in Fig. 2. Our approach tracks people and identifies the target person based on [24], [25]. Additionally, we employ FAST-LIO [26] for robot localization and construct a local map by filtering the 3D LiDAR scans to retain points within a narrow band around the robot’s body height, which are then projected onto a 2D plane. Adap-RPF leverages this perception information to proactively follow the target person.

A. Human Trajectory Prediction

Building on the people detection, tracking, and target identification modules, we obtain the global positions of the target and surrounding pedestrians. At each timestep, each pedestrian’s position is appended to a FIFO queue. The resulting historical trajectories are used to predict an N -step future trajectory with a constant velocity model (CVM), which offers high computational efficiency. Our framework is agnostic to the choice of prediction model, allowing seamless integration of alternative trajectory predictors.

B. Adaptive Trajectory Sampling

Unlike fixed-point following schemes, our method performs target-centric adaptive trajectory sampling. As illustrated in Fig. 3, this approach emphasizes the multi-objective evaluation over a dense set of candidate points, thereby expanding the feasible solution space and enhancing flexibility in dynamic, crowded environments. The process consists of three steps: 1) sampling candidate following points, 2) evaluating them using a multi-objective cost function, and 3) estimating the robot’s person-following trajectory based on the optimal point and predicted target trajectory.

1) **Candidate Point Sampling:** We employ a low-discrepancy point-sampling strategy (Sobol sampling [27]) to generate candidate points along the target’s motion direction. These points uniformly populate the sampling space, enabling a comprehensive evaluation of potential following positions. The sampling space is defined as a backward-facing semi-annular region aligned with the target’s motion direction. Its inner and outer radii correspond to the target’s personal and social zones, respectively, thereby satisfying proxemic constraints [23] and promoting comfortable, socially compliant following behavior. Specifically, given two consecutive target positions $\mathbf{p}_t^{\text{Tar}} = [x_t^{\text{Tar}}, y_t^{\text{Tar}}]^T$ and $\mathbf{p}_{t+1}^{\text{Tar}} = [x_{t+1}^{\text{Tar}}, y_{t+1}^{\text{Tar}}]^T$, the target person’s heading is:

$$\phi = \text{atan2}(y_{t+1}^{\text{Tar}} - y_t^{\text{Tar}}, x_{t+1}^{\text{Tar}} - x_t^{\text{Tar}}). \quad (1)$$

Accordingly, Sobol’ points $u_i = (u_{i,1}, u_{i,2}) \in [0, 1]^2$ are mapped to a semi-annulus whose inner radius r_{\min} corresponds to the target’s personal zone and outer radius r_{\max} to the social zone, thereby ensuring socially comfortable following. The mapping uses an area-uniform radius and a back-half angle, and we sample 50 points to adequately cover the region while maintaining computational efficiency:

$$r_i = \sqrt{r_{\min}^2 + (r_{\max}^2 - r_{\min}^2)u_{i,1}}, \quad \theta_i = \phi + \frac{\pi}{2} + \pi u_{i,2}. \quad (2)$$

Thus, each candidate following point can be expressed in Cartesian coordinates as:

$$\mathbf{c}_i = \begin{bmatrix} x_t^{\text{Tar}} + r_i \cos \theta_i \\ y_t^{\text{Tar}} + r_i \sin \theta_i \\ \theta_i \end{bmatrix}, \quad (3)$$

which serve as candidate following positions for subsequent multi-objective evaluation.

2) **Multi-objective Evaluation:** Subsequently, the sampled target-centric candidate points \mathbf{c}_i are processed by first filtering out those occupied by pedestrians using a local map. The remaining candidates are then assessed through a multi-objective cost function:

$$\min J(\mathbf{c}_i) = w_{\text{occ}}J_{\text{occ}}(\mathbf{c}_i) + w_{\text{prox}}J_{\text{prox}}(\mathbf{c}_i) + w_{\text{dist}}J_{\text{dist}}(\mathbf{c}_i) + w_{\text{trav}}J_{\text{trav}}(\mathbf{c}_i) + w_{\text{stick}}J_{\text{stick}}(\mathbf{c}_i). \quad (4)$$

where these weights are empirically set to $w_{\text{occ}} = 10.0$, $w_{\text{prox}} = 1.0$, $w_{\text{dist}} = 10.0$, $w_{\text{trav}} = 1.0$, and $w_{\text{stick}} = 0.5$.

Occlusion Cost. To evaluate whether the target will be occluded when the robot arrives at a candidate position \mathbf{c}_i , we assume the robot will reach \mathbf{c}_i after N steps from the current time t using constant velocity. We then construct a virtual camera extrinsic matrix from \mathbf{c}_i and project the predicted positions of the target \mathbf{p}_{t+N}^T and each pedestrian \mathbf{p}_{t+N}^h at time $t + N$ into the panoramic image. The occlusion cost is defined as:

$$J_{\text{occ}}(\mathbf{c}_i) = \frac{1}{|\mathcal{H}_N|} \sum_{h \in \mathcal{H}_N} \text{IoU}(\mathcal{B}_T(\mathbf{p}_{t+N}^T, \mathbf{c}_i), \mathcal{B}_h(\mathbf{p}_{t+N}^h, \mathbf{c}_i)), \quad (5)$$

where \mathcal{H}_N is the set of predicted pedestrians at time $t + N$. $\mathcal{B}_T(\cdot, \mathbf{c}_i)$ and $\mathcal{B}_h(\cdot, \mathbf{c}_i)$ are the projected bounding boxes of

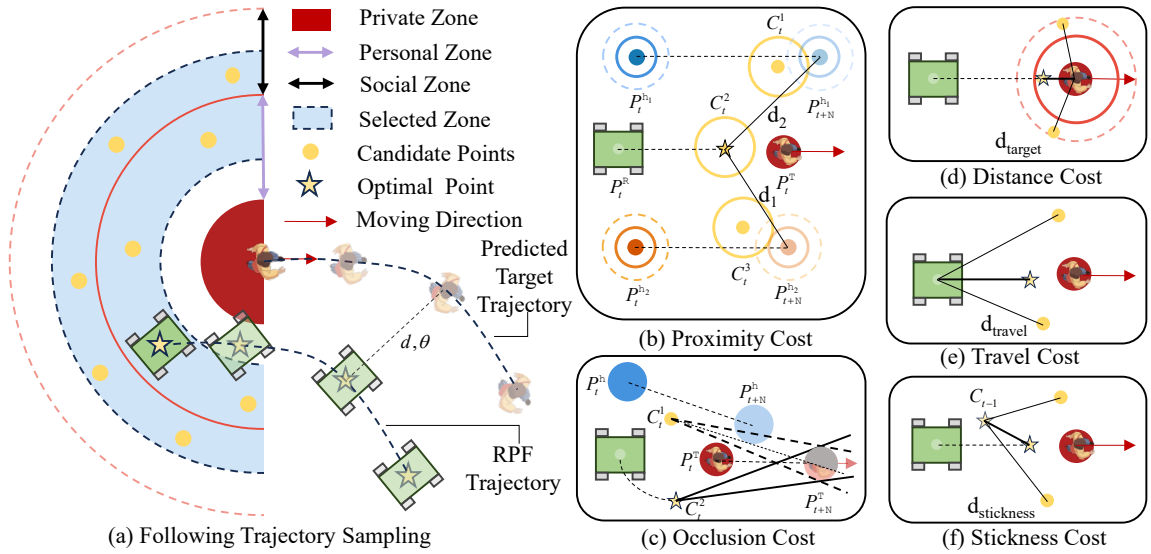


Fig. 3. **Target-centric Adaptive Following Trajectory Sampling.** (a) **Following Trajectory Sampling.** Candidate following points are generated using Sobol sampling within a target-centric semi-annular region defined by the target’s personal and social zones. The candidates are evaluated using a multi-objective cost function that accounts for target visibility, collision risk, social compliance, and smoothness. A following trajectory is then constructed relative to the predicted target trajectory using an offset (d, θ) based on the selected optimal point. (b) **Proximity Cost (Eq. 6).** Minimum distance d_1 to surrounding pedestrians is computed from predicted trajectories and social zones; candidates intruding into personal space (e.g., C_i^3, C_i^1) are discarded, and C_i^2 is selected. (c) **Occlusion Cost (Eq. 5).** Occlusion is estimated using the IoU between pedestrian and target projections (P_{t+N}^h and P_{t+N}^{Tar}) using predicted trajectories; C_i^1 is rejected due to partial occlusion (light gray), whereas C_i^2 remains visible. (d) **Distance Cost (Eq. 8).** Penalizes deviation from the desired distance to the target. (e) **Travel Cost (Eq. 9).** Encourages minimal path effort from the robot’s current position (green rectangle P_t^R). (f) **Stickiness Cost (Eq. 10).** Promotes continuity by favoring points near the previously selected following point.

the target and pedestrian h . $\text{IoU}(\cdot, \cdot)$ denotes the intersection-over-union between two projected bounding boxes.

Proximity Cost. To ensure the robot avoids intruding into the personal zones of surrounding pedestrians, we define the proximity cost:

$$J_{\text{prox}}(\mathbf{c}_i) = \begin{cases} 0, & d_{\min}(\mathbf{c}_i) \geq d_{\max}, \\ \left(\frac{d_{\max} - d_{\min}(\mathbf{c}_i)}{d_{\max} \cdot d_{\min}(\mathbf{c}_i)} \right)^2, & \varepsilon \leq d_{\min}(\mathbf{c}_i) < d_{\max}, \\ M, & d_{\min}(\mathbf{c}_i) < \varepsilon, \end{cases} \quad (6)$$

where $d_{\min}(\mathbf{c}_i)$ represents the minimum distance between a candidate point \mathbf{c}_i and the predicted pedestrian positions at the corresponding future time step N . This distance is calculated as:

$$d_{\min}(\mathbf{c}_i) = \min_{h \in \mathcal{H}_{\ell}(\mathbf{c}_i)} \left\| \mathbf{c}_i - \mathbf{p}_{\ell}^h(\mathbf{c}_i) \right\|_2. \quad (7)$$

Here, d_{\max} denotes the safety threshold, set to 1.2 m to align with the boundary of the pedestrian’s personal zone [23]; ε represents the lower bound to avoid collisions; and M is a large penalty value of 1000, used as a filtering threshold.

Distance Cost. To maintain a socially acceptable distance within the target-centric personal zone while accounting for the robot’s physical footprint, we penalize deviations from a desired following distance d^* , set to 1.5 m in our experiments. Let $\mathbf{c}_i^p = [c_i^x, c_i^y]^\top$ denote the 2D positional component of candidate \mathbf{c}_i . The distance cost is defined as:

$$J_{\text{dist}}(\mathbf{c}_i) = \left(\left\| \mathbf{c}_i^p - \mathbf{p}_t^{\text{Tar}} \right\|_2 - d^* \right)^2. \quad (8)$$

Travel Cost. The travel cost measures the Euclidean distance between a candidate point and the robot’s current

position $\mathbf{p}_t^R = [x_t^R, y_t^R]^\top$, promoting navigational efficiency by favoring closer goals:

$$J_{\text{trav}}(\mathbf{c}_i) = \left| \mathbf{c}_i^p - \mathbf{p}_t^R \right|_2. \quad (9)$$

Stickiness Cost. To promote motion smoothness and discourage abrupt changes in the selected point, the stickiness cost is defined as:

$$J_{\text{stick}}(\mathbf{c}_i) = \begin{cases} \left\| \mathbf{c}_i - \mathbf{c}_{\text{prev}} \right\|_2, & \text{if } \mathbf{c}_{\text{prev}} \text{ exists,} \\ 0, & \text{otherwise.} \end{cases} \quad (10)$$

Through the above sampling and evaluation process, we select the optimal following point that minimizes the multi-objective cost function (Eq. 4).

3) Person-Following Trajectory Estimation: Upon determining the optimal following point, instead of merely tracking the point as in other methods, we calculate the relative distance d and orientation θ with respect to the target. Based on this, a goal trajectory G is estimated to align with the predicted target trajectory, enabling the lower MPPI local controller to achieve smoother motion.

C. Prediction-Aware MPPI Controller

Given the estimated person-following trajectory, we employ a prediction-aware MPPI controller for trajectory tracking, collision avoidance and motion smoothness. We adopt MPPI [28] because this sampling-based MPC ensures fast computation and avoids solver failures, making it well-suited for real-time navigation in dynamic environments. For completeness, we briefly introduce our prediction-aware MPPI in the following section.

For the robot dynamic model, we assume a differential model defined as:

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k) = \begin{bmatrix} x_k^R + v_k \cos \theta_k \Delta t \\ y_k^R + v_k \sin \theta_k \Delta t \\ \theta_k + \omega_k \Delta t \end{bmatrix}, \quad (11)$$

where $\mathbf{x}_k = [x_k^R, y_k^R, \theta_k]^\top$ denotes the robot state, serving as the foundation for forward simulation of the MPPI controller. As for the cost function, we introduce three cost terms:

$$\begin{aligned} \min J(k) &= \sum_{k=0}^{K-1} (w_{\text{goal}} J_{\text{goal}}(k) + w_{\text{enc}} J_{\text{enc}}(k) + w_u J_u(k)), \\ \text{s.t. } & \mathbf{u}_{\min} \leq \mathbf{u}_k \leq \mathbf{u}_{\max}, \end{aligned} \quad (12)$$

where $k = 0, \dots, K-1$ indexes the horizon, w are the cost weights, and J are the cost functions, defined as follows.

Goal Tracking Cost. To ensure smooth and continuous motion, the robot follows the estimated person-following trajectory aligned with the MPPI horizon. At each step, the robot's position $\mathbf{p}_k^R = [x_k^R, y_k^R]^\top$ is encouraged to follow the corresponding goal point $\mathbf{g}_k \in G$:

$$J_{\text{goal}}(k) = \|\mathbf{p}_k^R - \mathbf{g}_k\|_2^2. \quad (13)$$

Encroachment Cost. To avoid entering pedestrians' personal zones, we penalize candidate states that fall within a safety margin of $d_{\text{safe}} = 1.2$ m. This margin accounts for both the robot's and pedestrians' physical dimensions around their predicted positions:

$$J_{\text{enc}}(k) = \sum_{j=1}^{N_h} \max\left\{0, d_{\text{safe}}^2 - \|\mathbf{p}_k^R - \mathbf{p}_k^{h_j}\|_2^2\right\}. \quad (14)$$

Control Effort Cost. To further ensure motion smoothness, we penalize the variation in control inputs between consecutive time steps:

$$J_u(k) = \|\mathbf{u}_k - \mathbf{u}_{k-1}\|_2, \quad (15)$$

where $\mathbf{u}_k = [v_k, \omega_k]^\top$ and \mathbf{u}_{k-1} denote the control inputs at the current and previous time steps, respectively.

These cost functions further refine the person-following trajectory by integrating dynamic constraints and prediction-aware collision avoidance, enabling the robot to proactively follow the target person in dynamic crowded environments.

IV. EXPERIMENTS

To evaluate our Adap-RPF, we conduct experiments on a public benchmark. This section first introduces the benchmark (IV-A), baselines (IV-B), and implementation details (IV-C). We then compare our approach with fixed-point, multiple predefined-point and optimization-based baselines (IV-D.1), followed by ablation studies analyzing the role of human trajectory prediction in explicit dynamic point selection and local planning (IV-D.2). Finally, we demonstrate the system's real-world effectiveness in dynamic crowded environments with a mobile robot, as shown in Fig. 1.

A. Benchmark

We use a public benchmark [29] to quantitatively evaluate RPF performance in dynamic environments. The benchmark provides four realistic dynamic-crowd scenarios: perpendicular, parallel, circular, and random crowd. These scenarios are primarily distinguished by pedestrian motion patterns. In the perpendicular scenario, pedestrians cross orthogonally to the target's path, simulating intersections. In the parallel scenario, pedestrians move in the same direction as the target, resembling corridors or sidewalks. In the circular crowds scenario, pedestrians traverse along straight lines connecting opposite points on a circle, mimicking diagonal crossing in plazas or open squares. In the random crowds scenario, pedestrian motions are randomized and disordered, reflecting the chaotic flow in densely populated environments such as shopping malls or train stations.

In this benchmark, each pedestrian is modeled as a circle with a radius of 0.3 m. Their interaction behaviors are simulated using ORCA [30]. The benchmark presents diverse and challenging conditions, including dynamic occlusions, unpredictable pedestrian trajectories, and frequent crossings. These challenges make it difficult to maintain target visibility, providing a solid evaluation for RPF planning methods. Representative examples are shown in Fig. 4.

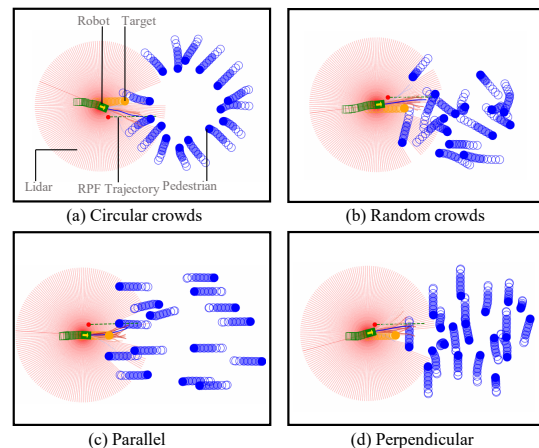


Fig. 4. **Representative dynamic-crowd scenarios in the public RPF benchmark:** (a) circular crowds, (b) random crowds, (c) parallel and (d) perpendicular. Red beams indicate Lidar scans, while the green dashed line represents the RPF trajectory. Blue circles denote moving pedestrians, the green rectangle represents the robot, and the yellow circle marks the target being followed.

B. Baselines

To evaluate target visibility and following performance on the benchmark, we compare our method Adap-RPF with several baselines categorized as follows:

1) *Multiple Predefined Following Points:* **Vu's Method** [22] adopts a similar sample-evaluate-plan approach to ours but leverages current scene observations for following-point selection and local planning.

2) *Optimization-based:* **Wang's Method** [18] is the first to maintain target visibility in RPF by formulating an optimization function within the NMPC framework, incorporating multiple constraints.

TABLE I. Comparison of task success rates (%) across four benchmark scenarios with varying pedestrian numbers (5–30). The task success rate represents the proportion of successfully completed following tasks under different pedestrian numbers in the benchmark scenarios. **Bold** indicates the best result, while underlined marks the second-best.

Method	Circular						Crowd						Parallel						Perpendicular					
	5	10	15	20	25	30	5	10	15	20	25	30	5	10	15	20	25	30	5	10	15	20	25	30
Wang’s [18]	43.75	16.25	6.25	2.50	3.75	11.25	85.00	37.50	30.00	13.75	0.00	0.00	95.00	95.00	90.00	75.00	70.00	40.00	83.75	38.75	17.50	10.00	3.75	0.00
Vu’s [22]	67.50	50.00	21.25	23.75	<u>23.75</u>	<u>20.00</u>	78.75	57.50	25.00	13.75	16.25	2.50	86.25	73.75	61.25	62.50	35.00	22.50	73.75	32.50	22.50	12.50	8.75	10.00
RDA-Traj [29]	<u>77.50</u>	<u>57.50</u>	<u>35.00</u>	35.00	22.50	12.50	82.50	82.50	<u>57.50</u>	55.50	35.00	35.00	<u>95.00</u>	<u>95.00</u>	82.50	<u>85.00</u>	<u>72.50</u>	<u>77.50</u>	82.50	<u>72.50</u>	<u>47.50</u>	<u>42.50</u>	32.50	25.00
Ours	83.62	64.34	49.45	<u>33.68</u>	47.39	37.43	93.75	<u>72.50</u>	61.25	<u>42.50</u>	<u>32.50</u>	<u>12.50</u>	100.00	98.75	98.75	90.00	92.50	85.00	90.00	78.75	65.00	56.25	<u>31.25</u>	<u>17.50</u>

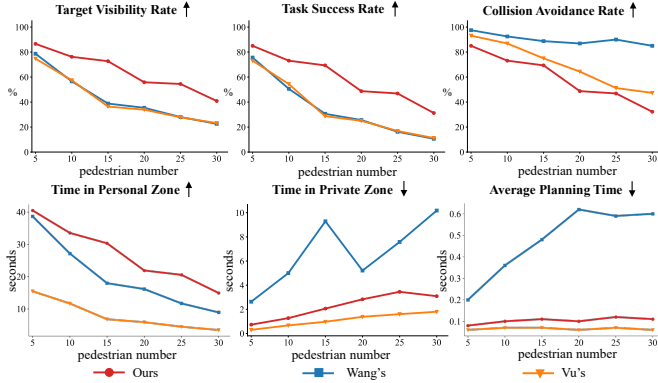


Fig. 5. Comparison of our method with optimization-based and multiple predefined points baselines across six metrics on the public benchmark. The horizontal axis represents pedestrian number (5–30). Each method is tested across four scenarios: circular crowds, random crowds, parallel, and perpendicular. Results are averaged over 20 random instances per scenario and then across all scenarios.

3) *Fixed Following Point: RDA Planner Trajectory Following (RDA-Traj)* achieves SOTA performance in the public benchmark [29]. It employs a Kalman-filter-based prediction model to estimate the future trajectory of the target. A person-following trajectory is then extrapolated based on the predicted target trajectory and a fixed following position. The RDA planner [31] is used to track this trajectory.

To ensure a fair comparison of proactive planning capabilities, no person-search techniques [11] are employed; therefore, the robot does not attempt to re-identify the target after occlusion.

C. Implementation Details

We first compute the time required for the target to traverse each dynamic environment without robot following. This duration is scaled by a factor of three, and the maximum time across all environments is used as the evaluation horizon for all algorithms. To ensure diverse and comprehensive testing, we randomly generate 50 scenarios per environment by sampling pedestrian start and end positions within the target’s traversed region. Each scenario includes 5, 10, 15, 20, 25, and 30 pedestrians. All evaluations are performed on a computer with an Intel(R) Core(TM) i9-14900K CPU and NVIDIA GeForce RTX 3090 GPU.

For real robot experiments, we use YOLOX [32] for person detection and a deep re-identification model [24] to track the target. The platform is a ScoutMini differential-drive robot equipped with an Intel NUC 11 (i7-1165G7

CPU @ 2.80GHz, NVIDIA RTX 2060). A Livox Mid-360 LiDAR ($360^\circ \times 59^\circ$ FOV, 10 Hz) and a Ricoh Theta panoramic camera (1280×720 , 15 Hz) are mounted on the robot (Fig. 1). All planners and controllers are configured identically to the simulation setup for consistency.

D. Experiment Results

1) *Evaluation Metrics:* We use six common metrics for evaluation. (1) Target visibility rate (TVR) [18], the proportion of steps where the target remains visible; (2) Collision avoidance rate (CVR), the proportion of scenarios where the robot avoids any collision; (3) Task success rate (TSR), the percentage of trials where the target is still visible at the final step; (4) Time in personal zone (TPZ), the number of steps the robot stays within the target’s personal zone (0.45–1.2 m) [23]; (5) Time in private zone (TPrZ), the number of steps the robot intrudes into any pedestrian’s intimate zone (0–0.45 m) [23]; and (6) Average planning time (APT), the average planning runtime per scenario.

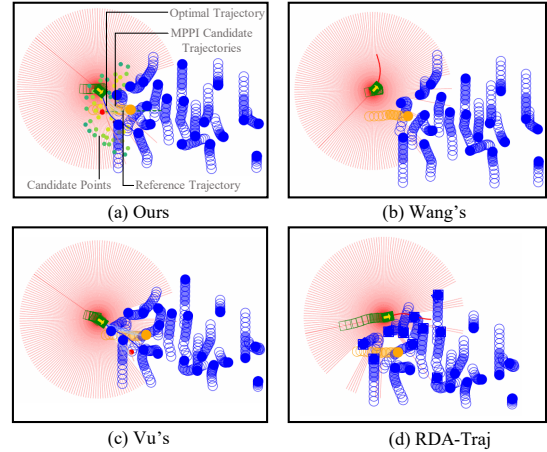


Fig. 6. Comparison of visualization results in the perpendicular scenario with 20 pedestrians. (a) Ours: adaptively switches to the optimal red point to avoid dynamic occlusion; (b) Wang’s: produces incorrect solutions; (c) Vu’s: collides with pedestrians; (d) RDA-Traj: uses a fixed point, directly runs into the crowd, and loses the target.

As shown in Fig. 5, Adap-RPF significantly outperforms existing SOTA proactive RPF approaches, including predefined-point and optimization-based strategies, across key metrics: TSR, TVR, and TPZ. For TSR, Adap-RPF achieves approximately 10% higher performance than the baselines with five pedestrians. Under the 15-pedestrian setting, it reaches 69.38%, outperforming Wang’s method by 38.76% and Vu’s by 40.63%. Even in high-density

TABLE II. **Ablation study on TVR and TSR.** TVR is the proportion of steps where the target remains visible, and TSR is the percentage of trials where the target is visible at the final step. Both metrics (%) are evaluated under varying pedestrian counts (5 to 30) to assess the impact of adaptive sampling (Adap.) and trajectory prediction (Traj. Pred.).

Ablation		TVR / %						TSR / %					
Adaptive Trajectory Sampling	MPPI Controller	5	10	15	20	25	30	5	10	15	20	25	30
w/o Adap.	w/ Traj. Pred.	86.15	77.18	61.92	52.98	38.89	41.03	85.00	75.94	59.06	48.44	30.32	31.56
w/o Traj. Pred.	w/o Traj. Pred.	62.56	46.33	29.61	25.41	21.86	19.56	58.44	38.44	19.38	14.06	8.12	5.94
w/o Traj. Pred.	w/ Traj. Pred.	81.52	74.50	59.36	50.01	40.64	36.06	80.62	74.06	55.94	44.68	33.44	24.06
w/ Traj. Pred.	w/o Traj. Pred.	78.32	61.90	47.22	40.17	34.02	32.78	77.50	58.44	42.50	31.88	25.31	21.25
w/ Traj. Pred.	w/ Traj. Pred.	91.96	79.68	70.40	58.76	55.68	44.26	91.25	77.18	65.94	51.88	47.82	34.38

scenarios with 25 pedestrians, it remains robust at 46.88%, exceeding both baselines by over 30%. For TVR, Adap-RPF maintains a 10% advantage at five pedestrians and reaches 72.70% with 15 pedestrians, surpassing Wang by 33.90% and Vu by 36.28%. At a density of 25 pedestrians, its TVR stays above 50%, while both baselines fall below 30%, yielding a margin of over 26%. For TPZ, Adap-RPF achieves 40.47 s with five pedestrians, 1.5 s higher than Wang and 50% higher than Vu. With 15 pedestrians, it records 30.31 s, representing 1.7 and 4.5 times the performance of Wang and Vu, respectively. These results demonstrate that Adap-RPF ensures superior tracking, target visibility, and social compliance across varying pedestrian densities. Fig. 6 provides intuitive visualizations supporting these findings. In a perpendicular scenario with 20 pedestrians, Adap-RPF selects optimal trajectories, while Wang’s method produces incorrect solutions. Vu’s method leads to collisions, and the fixed-point RDA-Traj approach loses the target in dense crowds. While Adap-RPF demonstrates strong performance across key metrics, its APT and TPrZ are slightly higher than those of Vu’s method. Nevertheless, the method remains real-time and offers a more comprehensive evaluation of target visibility and social comfort. Although the CVR is lower due to the relatively aggressive nature of our approach, Adap-RPF consistently maintains high TVR and TSR.

As summarized in Table. I, we compare Adap-RPF with Wang’s method, Vu’s method, and the fixed-point baseline RDA-Traj, averaging TSR across fixed-back and side-following modes in various scenarios with different pedestrian densities. While our method outperforms the others in most cases, RDA-Traj occasionally performs better in some crowded and perpendicular scenarios. This advantage comes from the fixed-back strategy, where the robot follows directly behind the target, leveraging the target’s natural obstacle avoidance to create a clear path without active planning. Our method does not explicitly account for environmental patterns, which limits its performance in such cases. Future work will incorporate environmental patterns into adaptive trajectory sampling for further improvement.

2) *Ablation Study of Adap-RPF:* To assess the contributions of adaptive trajectory sampling and human trajectory prediction in both the sampling module and the MPPI controller, we conduct ablation studies using TVR and TSR as evaluation metrics. The results are summarized in Table II.

Ablation 1 replaces adaptive trajectory sampling with a fixed relative following point, and constructs a trajectory by maintaining constant distance and orientation along the predicted target path; MPPI still uses human trajectory prediction. Ablation 2 removes trajectory prediction entirely from both adaptive trajectory sampling and MPPI. Ablation 3 disables prediction in adaptive trajectory sampling but retains it in MPPI. Ablation 4 enables prediction for adaptive trajectory sampling but disables it in MPPI. These variants isolate the effects of adaptive trajectory sampling and the role of human trajectory prediction.

We observe that ablation 1 achieves 86.15% TVR and 85.00% TSR, below our method (91.96% TVR and 91.25% TSR) under the low-density setting with five pedestrians. As the number of pedestrians increases to 15–20, both metrics drop more sharply, indicating that our method remains effective under frequent occlusions. However, performance declines at 30 pedestrians due to overcrowding, suggesting the need to incorporate crowd flow information. These results emphasize the importance of adaptive trajectory sampling.

In the five-pedestrian scenario, ablation 2 performs the worst, with TVR and TSR dropping to 62.56% and 58.44%, more than 25–30% below our method. Ablation 3 and ablation 4 perform reasonably well under five pedestrians, with TVR at 81.52% and 78.32%, and TSR at 80.62% and 77.50%, respectively. However, their performance drops sharply as density increases, with TVR falling below 60% and TSR below 56%, which is 10–20% lower than our method. These results from ablation 2 to ablation 4 highlight that human trajectory prediction is critical for both adaptive trajectory sampling and the MPPI controller. The best performance is achieved when these components are effectively combined.

E. Discussion

As shown in Fig. 5, the proposed Adap-RPF outperforms optimization-based and predefined-point baselines in TVR, TSR, and TPZ across varying pedestrian densities. This is achieved by sampling candidate points within the target’s social zones and evaluating them using a multi-objective cost function that considers target visibility, collision avoidance, human comfort, and motion smoothness. The optimal point generates a smooth, socially compliant trajectory, tracked by a prediction-aware MPPI controller.

Ablation results (Table. II) further highlight that trajectory prediction is essential for both adaptive sampling and MPPI, playing a key role in maintaining stable visibility and robust following. Nonetheless, as shown in Table. I, Adap-RPF does not fully exploit the natural shielding effect of following directly behind a person. This limitation leads to performance degradation in highly crowded scenarios, highlighting an area for improvement. Future work will explore leveraging the crowd flow information to further improve generalization and robustness.

V. CONCLUSIONS

In this paper, we present Adap-RPF, a hierarchical planning framework that proactively integrates multiple objectives—target visibility, collision avoidance, motion smoothness, and human comfort. By densely sampling candidate points within the target-centric social space and evaluating them using predicted trajectories of both the target and surrounding pedestrians, Adap-RPF generates a proactive following trajectory that guides a prediction-aware MPPI controller, enhancing both comfort and safety. A key advantage of our method is its ability to maintain target visibility for as long as possible, even under dynamic occlusions, which is crucial for robust and consistent person following in crowded environments. Compared with baseline methods, Adap-RPF achieves SOTA person-following performance across multiple metrics in diverse dynamic scenarios on public benchmarks and real-world experiments.

REFERENCES

- [1] M. J. Islam, J. Hong, and J. Sattar, “Person-following by autonomous robots: A categorical overview,” *Int. J. Robot. Res. (IJRR)*, vol. 38, no. 14, pp. 1581–1618, 2019.
- [2] M. A. Goodrich, A. C. Schultz *et al.*, “Human–robot interaction: a survey,” *Foundations and trends® in human–computer interaction*, vol. 1, no. 3, pp. 203–275, 2008.
- [3] A. Eirale, M. Martini, and M. Chiaberge, “Human following and guidance by autonomous mobile robots: A comprehensive review,” *IEEE Access*, 2025.
- [4] A. Leigh, J. Pineau, N. Olmedo, and H. Zhang, “Person tracking and following with 2d laser scanners,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*. IEEE, 2015, pp. 726–733.
- [5] J. Peng, Z. Liao, Z. Su, H. Yao, Y. Zeng, and H. Dai, “A dual closed-loop control strategy for human-following robots respecting social space,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*. IEEE, 2024, pp. 11 252–11 258.
- [6] J. Zhao, H. Ye, Y. Zhan, H. Luan, and H. Zhang, “Human orientation estimation under partial observation,” in *Proc. IEEE/RJS Int. Conf. Intell. Robots Syst. (IROS)*. IEEE, 2024, pp. 11 544–11 551.
- [7] P. Nikdel, R. Shrestha, and R. Vaughan, “The hands-free push-cart: Autonomous following in front by predicting user trajectory around obstacles,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*. IEEE, 2018, pp. 4548–4554.
- [8] P. Nikdel, R. Vaughan, and M. Chen, “Lbgp: Learning based goal planning for autonomous following in front,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*. IEEE, 2021, pp. 3140–3146.
- [9] L. Y. Morales Saiki, S. Satake, R. Huq, D. Glas, T. Kanda, and N. Hagita, “How do people walk side-by-side? using a computational model of human behavior for a social robot,” in *Proc. ACM/IEEE Int. Conf. Human-Robot Interact. (HRI)*, 2012, pp. 301–308.
- [10] J. Peng, Z. Liao, H. Yao, Z. Su, Y. Zeng, and H. Dai, “Mpc-based human-accompanying control strategy for improving the motion coordination between the target person and the robot,” in *Proc. IEEE/RJS Int. Conf. Intell. Robots Syst. (IROS)*. IEEE, 2023, pp. 7969–7975.
- [11] H. Ye, K. Cai, Y. Zhan, B. Xia, A. Ajoudani, and H. Zhang, “Rpf-search: Field-based search for robot person following in unknown dynamic environments,” *IEEE/ASME Trans. Mechatronics*, pp. 1–12, 2025.
- [12] B.-J. Lee, J. Choi, C. Baek, and B.-T. Zhang, “Robust human following by deep bayesian trajectory prediction for home service robots,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2018, pp. 7189–7195.
- [13] F. Hoeller, D. Schulz, M. Moors, and F. E. Schneider, “Accompanying persons with a mobile robot using motion prediction and probabilistic roadmaps,” in *Proc. IEEE/RJS Int. Conf. Intell. Robots Syst. (IROS)*. IEEE, 2007, pp. 1260–1265.
- [14] H. Yao, H. Dai, E. Zhao, P. Liu, and R. Zhao, “Laser-based side-by-side following for human-following robots,” in *Proc. IEEE/RJS Int. Conf. Intell. Robots Syst. (IROS)*. IEEE, 2021, pp. 2651–2656.
- [15] E. Repiso, G. Ferrer, and A. Sanfeliu, “On-line adaptive side-by-side human robot companion in dynamic urban environments,” in *Proc. IEEE/RJS Int. Conf. Intell. Robots Syst. (IROS)*. IEEE, 2017, pp. 872–877.
- [16] S. Leisiazar, S. R. R. Rohani, E. J. Park, A. Lim, and M. Chen, “Adapting to frequent human direction changes in autonomous frontal following robots,” *IEEE Robot. Autom. Lett.*, vol. 10, no. 3, pp. 2934–2941, 2025.
- [17] Y. Song, Q. Zhang, Z. Hu, and J. Liu, “Safe and robust human following for mobile robots based on self-avoidance mpc in crowded corridor scenarios,” in *Proc. IEEE Int. Conf. Robot. Biomim. (ROBIO)*. IEEE, 2023, pp. 1–6.
- [18] Z. Wang, S. Huang, M. Li, J. Ouyang, Y. Wang, and H. Chen, “Continuous robotic tracking of dynamic targets in complex environments based on detectability,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*. IEEE, 2024, pp. 16 338–16 344.
- [19] S. Leisiazar, E. J. Park, A. Lim, and M. Chen, “An mcts-drl based obstacle and occlusion avoidance methodology in robotic follow-ahead applications,” in *Proc. IEEE/RJS Int. Conf. Intell. Robots Syst. (IROS)*. IEEE, 2023, pp. 221–228.
- [20] V.-A. Le, B. Chalaki, V. Tadiparthi, H. N. Mahjoub, J. D’sa, and E. Moradi-Pari, “Social navigation in crowded environments with model predictive control and deep learning-based human trajectory prediction,” in *Proc. IEEE/RJS Int. Conf. Intell. Robots Syst. (IROS)*. IEEE, 2024, pp. 4793–4799.
- [21] S. Poddar, C. Mavrogiannis, and S. S. Srinivasa, “From crowd motion prediction to robot navigation in crowds,” in *Proc. IEEE/RJS Int. Conf. Intell. Robots Syst. (IROS)*. IEEE, 2023, pp. 6765–6772.
- [22] C.-T. Vu, H.-H. Huang, and Y.-C. Liu, “Autonomous navigation for human-following robots based on optimized position tracking,” in *Proc. IEEE Int. Conf. Control Robot. Eng. (ICCRE)*. IEEE, 2025, pp. 23–27.
- [23] E. T. Hall, “A system for the notation of proxemic behavior,” *American anthropologist*, vol. 65, no. 5, pp. 1003–1026, 1963.
- [24] H. Ye, J. Zhao, Y. Zhan, W. Chen, L. He, and H. Zhang, “Person re-identification for robot person following with online continual learning,” *IEEE Robot. Autom. Lett.*, 2024.
- [25] H. Ye, J. Zhao, Y. Pan, W. Cherr, L. He, and H. Zhang, “Robot person following under partial occlusion,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2023, pp. 7591–7597.
- [26] W. Xu and F. Zhang, “Fast-lio: A fast, robust lidar-inertial odometry package by tightly-coupled iterated kalman filter,” *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 3317–3324, 2021.
- [27] I. M. Sobol, “Distribution of points in a cube and approximate evaluation of integrals,” *USSR Comput. Math. Math. Phys.*, vol. 7, pp. 86–112, 1967.
- [28] G. Williams, N. Wagener, B. Goldfain, P. Drews, J. M. Rehg, B. Boots, and E. A. Theodorou, “Information theoretic mpc for model-based reinforcement learning,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*. IEEE, 2017, pp. 1714–1721.
- [29] H. Ye, W. Situ, J. Peng, Y. Zhan, B. Xia, K. Cai, and H. Zhang, “Follow-bench: A unified motion planning benchmark for socially-aware robot person following,” *arXiv preprint arXiv:2509.10796*, 2025.
- [30] J. Van Den Berg, S. J. Guy, M. Lin, and D. Manocha, “Reciprocal n-body collision avoidance,” in *Robotics Research: The 14th International Symposium ISRR*. Springer, 2011, pp. 3–19.
- [31] R. Han, S. Wang, S. Wang, Z. Zhang, Q. Zhang, Y. C. Eldar, Q. Hao, and J. Pan, “Rda: An accelerated collision free motion planner for autonomous navigation in cluttered environments,” *IEEE Robot. Autom. Lett.*, vol. 8, no. 3, pp. 1715–1722, 2023.
- [32] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, “Yolox: Exceeding yolo series in 2021,” *arXiv preprint arXiv:2107.08430*, 2021.