Mitigating Multi-Module Errors for Reliable Navigation in Dynamic Environments via Online Trajectory Refinement

Anonymous CVPR submission

Paper ID 0008

Abstract

Perception in dynamic obstacle avoidance scenarios is inevitably affected by errors from multiple perception modules, e.g., robot localization, dynamic detection, and trajectory prediction. Existing methods mainly focus on reducing errors in a single module, overlooking their interactions and accumulation, which degrade navigation performance. We propose a systematic framework to mitigate multi-perception module errors and improve the reliability of dynamic obstacle avoidance. The proposed framework integrates two key innovations: an online trajectory refinement network called Residual-DtACI, and a point cloud–based dynamic target localization method. The Residual-DtACI adaptively refines trajectories through multi-expert residual estimation, while the point cloud–based method improves localization accuracy and robustness to occlusions and pose variations. We conducted several experiments to evaluate the proposed framework regarding its effectiveness in reducing accumulated perception errors. Results found that errors across different perception modules were reduced, alleviating error accumulation and enhancing prediction accuracy. In addition, the evaluations on public benchmarks show significant gains, with Residual-DtACI improving robot trajectory accuracy by up to 88.5%, and the point cloud–based method increasing localization accuracy by 59.9%.

1. Introduction

Perception plays a crucial role in the autonomous navigation of mobile robots in dynamic environments. It provides comprehensive information about the surrounding obstacles and the robot's own states. Such information is the foundation of path planning and decision making, determining whether the robot can reliably navigate [36]. However, perception in dynamic obstacle avoidance is unavoidably affected by errors accumulated from different components, which ultimately degrade the performance. Most existing studies have primarily focused on reducing errors within individual components such as localization and trajectory prediction. For localization, several works improved SLAM performance by enhancing feature extraction [23, 25] and tracking robustness under dynamic interference [16, 33, 39]. For trajectory prediction, destination-guided strategies have been adopted to address the indeterminacy in human motion [18, 21, 35].

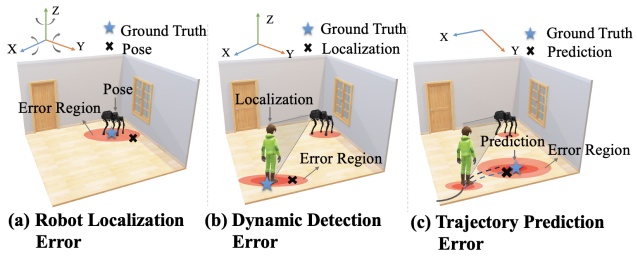

Figure 1. This figure shows perception error accumulating in dynamic obstacle avoidance. (a) Errors in the robot localization module lead to biases in the estimated pose trajectory. (b) Errors in the dynamic detection module cause inaccurate target localization. (c) Errors in the trajectory prediction module result in deviations in predicted motion. As these modules are sequentially coupled, the errors accumulate, ultimately degrading navigation performance.

However, these errors not only exist individually but also interact and accumulate over time due to the coupling among modules, directly impacting the effectiveness of dynamic obstacle avoidance. Existing works have not systematically analyzed and addressed these errors while considering their accumulated effects. Previous approaches that concentrate solely on reducing errors within a single component [7, 9, 16, 18, 21, 30, 33, 35, 39], while overlooking the error propagation from preceding and subsequent modules, cannot effectively reduce the influence of errors on the navigation system.

To overcome the above challenges, we proposed a framework that systematically addresses cumulative perception errors throughout the navigation process and improves the accuracy of dynamic target prediction. Specifically, our

framework focuses on three key components in the perception pipeline, including robot localization, dynamic obstacle detection, and trajectory prediction. It integrates trajectory refinement and target localization optimization to reduce accumulated perception errors and enhance the reliability of dynamic obstacle avoidance. In particular, we achieved the framework through two main innovations. Firstly, we designed the Residual-DtACI network for trajectory refinement. It builds upon the Conformal Inference method DtACI [10], extending with multi-dimensional residual prediction. It does not require prior knowledge, leverages multiple expert estimations, and demonstrates strong adaptability to different types of trajectory errors. In addition, we proposed a point cloud–based target localization method. This method transforms RGB-D inputs into 3D point clouds of the target through depth back-projection. The resulting point cloud is then denoised through a filtering function, from which the target's position is extracted. In the context of dynamic navigation, and in contrast to existing 2D localization [20], our method demonstrates improved localization accuracy in datasets with heavy occlusions and pose variations.

We integrated the above two innovations into a unified framework. Specifically, the Residual-DtACI is applied to refine the robot pose trajectory in the robot localization module as well as the predicted human trajectory in the trajectory prediction module. In parallel, the proposed point cloud–based method is employed in the dynamic detection module to perform target localization. By systematically reducing errors across perception modules and mitigating their accumulation, the framework ultimately enables reliable, collision-free obstacle avoidance. The main contributions are as follows:

- We proposed a framework that systematically analyzes the sources of perception errors and addresses their accumulated effects, effectively improving the accuracy of dynamic target prediction and thereby enhancing the reliability of dynamic obstacle avoidance.
- We proposed an online learning trajectory refinement network called Residual-DtACI. This network adopts trajectory residuals as the loss function to update the weights of multiple residual estimation experts. Using a unified scale, it refines both the robot pose trajectory in the localization module and the person prediction trajectory in the trajectory prediction module.
- We introduced a 3D point cloud-based dynamic target localization method for the dynamic detection module. Quantitative experiments demonstrate that this method achieves higher localization accuracy than conventional 2D approaches, while also providing robustness to occlusions and variations in target pose.

2. Related Works

2.1. Perception Errors in Dynamic Obstacle Avoidance

In visual SLAM, both tracking and mapping are affected by multiple factors, including illumination changes, dynamic objects, and scene texture characteristics, which inevitably introduce localization errors. To address these issues, several works have sought to reduce photometric error to improve the accuracy of camera pose tracking [7, 9, 30]. Other approaches attempt to remove dynamic feature points to mitigate the interference of moving objects [16, 33, 39]. In addition, alternative feature representations, such as 3D Gaussians [23] and neural point clouds [25], have been explored to improve reconstruction quality in complex and highly textured 3D environments. Moreover, in the detection module, YOLOv8 [13] balances speed and accuracy, enabling real-time detection on resource-constrained hardware. However, existing localization methods often neglect the geometric structure of objects, relying on 2D information for position estimation, which leads to errors [20].

Human trajectory prediction is particularly challenging due to the dynamic nature of human motion, such as the unknown long-term goals and the influence of social interactions on individual behavior. Recent studies have focused on forecasting long-term destinations as an intermediate step, first estimating the destination and then generating the trajectory [18, 21, 35]. Recently, PPT [18] adopts a progressive approach that enables the model to capture both short-term dynamics and long-term dependencies. Other works have emphasized the role of group dynamics by explicitly modeling pair-wise and higher-order interactions [1, 15, 34]. MART [15], for instance, employs a hypergraph transformer network to capture both individual behaviors and group-level influences. Overall, existing works primarily focus on addressing errors within individual modules, while overlooking cross-module interactions and accumulation of errors. Therefore, in this work, we aim to systematically address accumulated perception errors.

2.2. Trajectory Refinement

Trajectory refinement has emerged as an effective strategy to reduce prediction errors and has been widely applied in motion prediction for autonomous driving [6, 11, 19, 27, 40, 41]. Shi et al. [27] incorporated local map information as fine-grained trajectory features to refine predictions. Other studies employed residual learning by adopting a simple MLP to predict residuals [11] or introducing angular constraints to enhance refinement [19]. In addition, Zhou et al. [40] proposed a scenario-adaptive framework that dynamically adjusts the refinement process to improve prediction efficiency. Trajectory refinement has also been applied in human trajectory prediction. Yoon et al. [38] employed

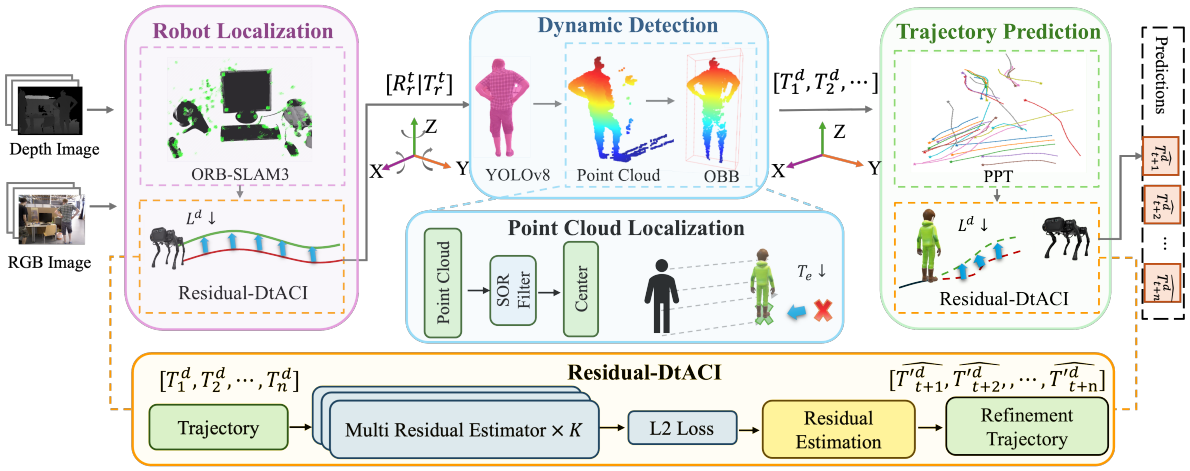

Figure 2. Overview of the proposed framework for systematically addressing perception errors in dynamic obstacle avoidance. The framework consists of three key modules: the robot localization module, which adopts ORB-SLAM3 as the baseline and applies the proposed Residual-DtACI network to refine the robot's pose estimation trajectory; the dynamic detection module, which uses YOLOv8 for person detection and incorporates our point cloud–based method to improve dynamic target localization accuracy; and the trajectory prediction module, which employs PPT as the baseline predictor and integrates the Residual-DtACI with a unified scale to reduce trajectory prediction errors. Overall, the proposed framework systematically reduces accumulated perception errors.

convolutional neural networks to refine predicted trajectories, while Wang et al. [29] focused on modeling future interactions among multiple agents to optimize refinement. However, these refinement methods lack a unified scale applicable across different trajectory tasks in robotic systems.

3. Method

As illustrated in Fig. 2, we proposed a framework to reduce accumulated perception errors in dynamic obstacle avoidance scenarios. This framework focused on reducing errors in three components, which are robot localization, dynamic detection, and trajectory prediction. As a whole, these three modules provide the robot with comprehensive perception information for dynamic navigation, including surrounding obstacles, self-localization, and target trajectory estimation. We developed an online learning network of trajectory refinement called Residual-DtACI, which performed refinement on both robot pose trajectories and dynamic target prediction trajectories within a unified structure. In addition, we introduced a point cloud–based method to improve the accuracy of object localization in the dynamic detection module. First, we adopted ORB-SLAM3 [5] to estimate the robot's pose trajectory from RGB-D image inputs and applied our proposed Residual-DtACI network for trajectory refinement. Next, YOLOv8 [13] was employed to detect pedestrians in RGB images, and depth information was incorporated with our method to generate 3D point clouds of the detected persons. A filtering function was then applied to denoise the point cloud, and oriented bounding boxes were used to extract the person's localization. Finally, we employed Progressive Pretext Task learning (PPT) [18] predictor to predict future trajectories of the persons, which were further refined using the Residual-DtACI. The refined trajectories were subsequently integrated into the navigation path planner to enable reliable dynamic obstacle avoidance.

3.1. Accumulation of Perception Errors in Dynamic Obstacle Avoidance Scenarios

This section describes how perception errors accumulate in the process of dynamic obstacle avoidance scenarios. First, in the SLAM process, dynamic objects, sensor noise, and long-term drift, which affect the capture of feature points, may cause the estimated robot pose $\hat{P}_r^t$ to deviate from its ground truth P_r^t, which inevitably introduces a pose error ε_r^t. This can be expressed as follows:

$$\hat{P}_r^t = P_r^t + \varepsilon_r^t \tag{1}$$

Next, YOLOv8 provides high detection accuracy with relatively fast inference speed to obtain the person mask. By combining the mask of the person with the depth image, the depth values $Z(u, v)$ corresponding to the masked pixels are retrieved. Through the intrinsic matrix K and the extrinsic transformation $\hat{P}_r^t$, the estimated person localization $\hat{T}_p^t$ is obtained. However, due to noise in the point cloud and errors in the extrinsic parameters (the robot pose estimation error), an additional error term $\varepsilon_p^t(\varepsilon_r^t)$ is introduced, which can be formulated as:

$$\hat{T}_p^t = \hat{P}_r^t \cdot K \cdot Z(u, v) = T_p^t + \varepsilon_p^t(\varepsilon_r^t) \tag{2}$$

Finally, a sequence of person localizations is used as the observed trajectory input to the trajectory predictor in order

to forecast the person's future trajectory. However, due to the indeterminate nature of human motion, existing models can hardly achieve highly accurate predictions, which introduces a trajectory prediction error $\varepsilon^{t+1:t+H}$. This error is further influenced by the preceding errors, as formulated below:

$$\hat{\tau}^{t+1:t+H} = g\left(\hat{T}_p^{1:t}\right) = \tau^{t+1:t+H} + \varepsilon_\tau^{t+1:t+H}\left(\varepsilon_p^t, \varepsilon_r^t\right) \tag{3}$$

Therefore, accumulated perception errors need to be addressed systematically. In our framework, we explicitly introduce refinement and correction mechanisms at multiple stages to reduce error propagation across the pipeline.

3.2. Trajectory Refinement

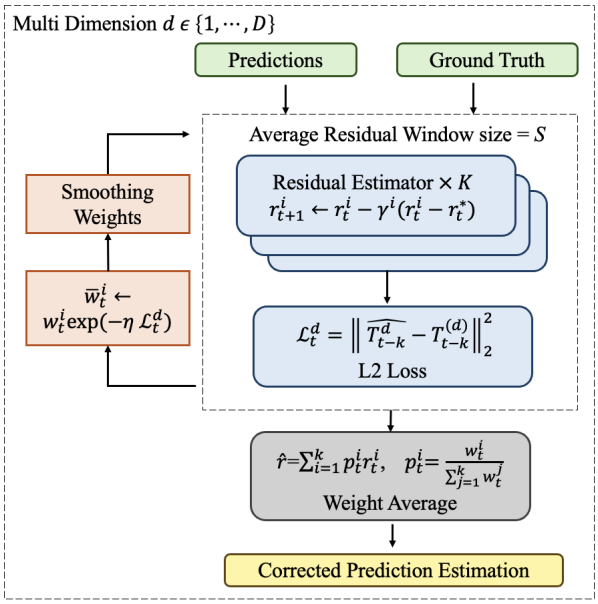

Figure 3. Architecture of the proposed Residual-DtACI network. The network consists of multiple expert models that perform online residual prediction. Their weights are updated using an L_2 loss, with a smoothing mechanism applied to prevent expert collapse and maintain expert diversity.

Trajectory refinement aims to minimize the deviation between the ground-truth trajectory $T^{1:t+H}$ and the predicted trajectory $\hat{T}^{1:t+H}$. In this stage, we leverage both the past ground truth and predicted trajectories $\{T^{1:t}, \hat{T}^{1:t}\}$ as inputs to compute the offset $\Delta T = T - \hat{T}$. This offset is then used to refine the future prediction. We adopt an L_2 loss to optimize the process:

$$\mathcal{L}^d = \left\| \Delta\hat{T}_t^d - \Delta T_t^d \right\|_2^2, \quad \forall\, 1 \le d \le D \tag{4}$$

where D denotes the dimensionality of the trajectory, and the loss is computed independently for each dimension.

3.3. Residual Dynamically-tuned Adaptive Conformal Inference

Building upon the trajectory refinement loss function introduced above, we further modified the Dynamically-Tuned Adaptive Conformal Inference (DtACI)[10] framework with our model and designed the proposed Residual-DtACI network, as shown in Fig 3. This network enabled online correction of predicted trajectories, allowing the system to rapidly adapt to distributional shifts without relying on prior knowledge of the underlying dynamics.

The network consists of multiple experts that perform error estimation $\hat{r}_{t+1}^{i,d}$ in parallel. Each expert i is updated based on the gradient of the residual loss $\nabla\mathcal{L}_t^d$, while adopting a distinct learning rate γ^i to control the update speed. Larger values of γ^i enable faster adaptation to abrupt distributional shifts, whereas smaller values lead to more stable but slower updates. The update formula is as follows:

$$\hat{r}_{t+1}^{i,d} \leftarrow \hat{r}_t^{i,d} - \gamma^i \nabla\mathcal{L}_t^d = \hat{r}_t^{i,d} - \gamma^i(\hat{r}_t^{i,d} - r_t^{i,d}) \tag{5}$$

Each expert's weight $\bar{w}_t^i$ is adaptively adjusted according to its loss $\mathcal{L}_t^d$, where the exponential weighting scheme places greater emphasis on recent performance. The parameter η is a hyperparameter that adjusts the speed at which the weights change and determines the sensitivity of the update:

$$\bar{w}_t^i \leftarrow w_t^i \exp\left(-\eta\, \mathcal{L}_t^d\right), \quad \forall\, 1 \le i \le k \tag{6}$$

After the exponential weighting, a smoothing step is applied to preserve weight diversity by mixing the normalized weights with a uniform distribution, ensuring that no expert loses all of its weight and that all experts retain a minimum probability of being selected. The σ is a smoothing coefficient:

$$w_{t+1}^i \leftarrow (1-\sigma)\frac{\bar{w}_t^i}{\sum_{i=1}^k \bar{w}_t^i} + \frac{\sigma}{k}, \quad \forall\, 1 \le i \le k \tag{7}$$

Based on the probability distribution of each expert p_{t+1}^i, the final prediction correction $\hat{r}_{t+1}^d$ is obtained as the weighted average of all expert predicted residuals $r_{t+1}^{i,d}$:

$$\hat{r}_{t+1}^d = \sum_{i=1}^k p_{t+1}^i,\ \hat{r}_{t+1}^{i,d}, \quad p_{t+1}^i = \frac{w_{t+1}^i}{\sum_{i=1}^k w_{t+1}^i} \tag{8}$$

Finally, the correction values for each dimension are added to the corresponding predicted trajectory, thereby completing the refinement of the trajectory:

$$\hat{T'}_{t+1}^d = \hat{T}_{t+1}^d + \hat{r}_{t+1}^d, \quad \forall\, 1 \le d \le D \tag{9}$$

3.4. Point Cloud- based Dynamic Target Localization

This section introduces a point cloud-based dynamic target localization method, which is designed to reduce the localization errors that occur during the dynamic detection process. Traditionally, localization is performed by applying

the YOLOv8 detection model to obtain bounding boxes of persons from RGB images [20]. The center of each bounding box, together with the corresponding depth value, is then transformed into the world coordinate system and regarded as the center position of the person, as illustrated in Fig. 4(a). However, this approach overlooks the inherent three-dimensional characteristics of a person, which often leads to errors in center localization.

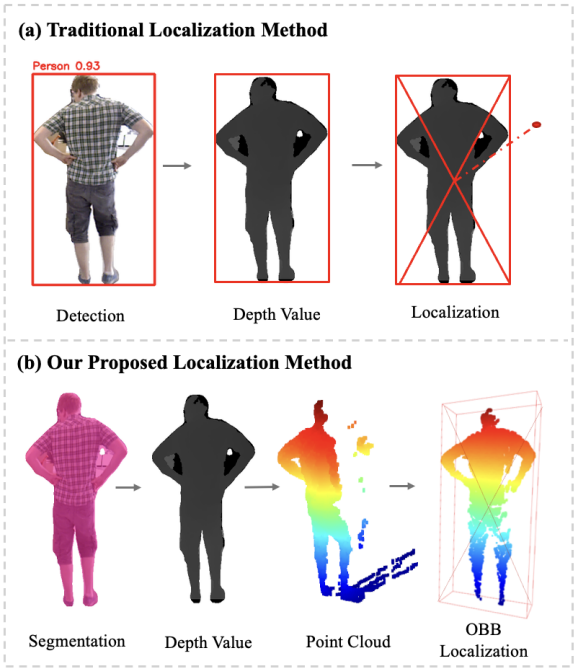

Figure 4. Illustration of dynamic target localization methods. (a) Traditional 2D approach using detection and depth value at the center of the box. (b) Our proposed point cloud-based method integrates segmentation, depth values, and point cloud cleaning with OBB to improve localization accuracy.

Thus, to better capture the 3D characteristics and spatial geometric structure of a person, we propose a point cloud-based dynamic target localization method. The proposed method allows for a more accurate estimation of the person's center position in environments, as illustrated in Fig. 4(b).

We used the YOLOv8 segmentation model to obtain the person mask (u, v) and extracted the corresponding depth values $Z(u, v)$. These pixel depth pairs were then transformed into the world coordinate system using the camera intrinsic matrix K and extrinsic parameters $[R \mid T]$, resulting in the person point cloud T_p:

$$T_p(u, v) = R \cdot \left(Z(u, v)\, K^{-1}[u, v, 1]^T\right) + T \qquad (10)$$

The extrinsic matrix $[R \mid T] = P_r \cdot P_c$ is the composition of the robot's pose in the world coordinate system P_r and the camera's pose relative to the robot P_c. Since the camera is rigidly mounted on the robot, P_c is typically fixed. Next, to eliminate the influence of noise in the person mask, we applied the Statistical Outlier Removal (SOR) filter function to clean the raw point cloud. For each point, the mean distance to its k nearest neighbors was computed. Any point with a mean distance greater than threshold $D_{\max}$ was considered an outlier and removed:

$$D_{\max} = \mu + \lambda \times \sigma \qquad (11)$$

Following cleaning the point cloud, an Oriented Bounding Box (OBB) was fitted to the target person's point cloud, and its center was taken as the person's localization.

4. Experiment

In this section, we elevated the effectiveness of our proposed framework in systematically reducing accumulated perception errors in dynamic obstacle avoidance scenarios. First, we validate the quantitative benefits of our methods in reducing errors on the three key perception modules, which are robot localization, dynamic detection, and trajectory prediction. Subsequently, we evaluate the performance of the complete framework on the THUD++ [17] dataset to demonstrate its ability to systematically reduce accumulated perception errors. In particular, our experiments are designed to answer the following four research questions:

- What is the quantitative benefit of trajectory refinement in reducing errors of robot pose trajectory estimation?
- What is the quantitative benefit of the point cloud–based method in reducing errors of dynamic target localization?
- What is the quantitative benefit of trajectory refinement in reducing errors of person trajectory prediction?
- What are the quantitative benefits of the complete framework in reducing accumulated perception errors across the perception pipeline?

4.1. Trajectory Refinement for Robot Pose Trajectory

We adopted ORB-SLAM3 as the baseline SLAM system and incorporated our Residual-DtACI network for robot pose trajectory refinement. Experiments were conducted on the TUM RGB-D [28] dataset, including 5 static sequences and 4 dynamic sequences, with the dynamic sequences containing highly dynamic objects interference. In addition to the baseline, we compared our approach with several state-of-the-art methods. For the dynamic sequences, we selected methods specifically designed to handle the interference of dynamic objects, Dynamic-VINS [33] and ViQu-SLAM [16]. For the static sequences, we included methods that emphasize accuracy in static environments, Point-SLAM [25] and GS-SLAM [23].

For evaluation, we used the Root Mean Square Error (RMSE) from the Absolute Trajectory Error (ATE). Prior

to error computation, all estimated trajectories were aligned to the ground truth using the SE(3) Umeyama alignment method. The Residual-DtACI network employs three error estimators with learning rates γ of 0.1, 0.2, and 0.4. The weight update parameter η is set to 2.72, and the smoothing coefficient σ is fixed at 0.001.

TUM RGB-D Static Dataset Table 1 reports the tracking results on the TUM RGB-D static dataset. Our method achieved consistently better robot pose tracking performance than the baseline ORB-SLAM3. In most sequences, the Residual-DtACI refinement yielded the best RMSE performance. For example, on the *fr2_xyz* sequence, our method achieved an RMSE of 0.0034 meters, corresponding to a 10.53% improvement over ORB-SLAM3. Similarly, on the *fr3_office* sequence, we obtained an RMSE of 0.0066 meters, resulting in a 36.54% improvement over the baseline. Beyond improving upon ORB-SLAM3, the refined trajectories also demonstrated superiority over several state-of-the-art methods, highlighting the effectiveness of our Residual-DtACI network in enhancing localization accuracy under static environments.

TUM RGB-D Dynamic Dataset Table 2 presents the tracking results on the TUM RGB-D dynamic sequences. These sequences are particularly challenging due to the presence of moving objects, which often cause drift in the tracking process. Our method achieved notable improvements in RMSE over ORB-SLAM3 across all dynamic sequences. For instance, on the *fr3_walking_half* sequence, our method achieved an 85.38% improvement over the baseline, while on the *fr3_walking_rpy* sequence, the improvement reached 88.53%. Furthermore, compared with state-of-the-art methods that rely on detection-and-removal strategies for dynamic objects, our Residual-DtACI network achieved a remarkable RMSE of 0.0307 meters on the *fr3_walking_rpy* sequence.

The above results demonstrate that our Residual-DtACI network is effective in both static and dynamic environments. In static sequences, pose tracking errors mainly arise from sensor noise and long-term drift, whereas in dynamic sequences, the interference of moving objects often leads to severe tracking drift. Our refinement approach effectively reduces these errors in both cases, achieving remarkable improvements in robot pose trajectory accuracy over the baseline and current methods.

4.2. Dynamic Target Localization

We evaluated our proposed point cloud–based method for dynamic target localization on the YCB-Video [32] and BEHAVE [4] datasets. As the baseline, we adopted a 2D detection–based method from prior work [20]. The YCB-Video dataset is a widely used benchmark for object pose estimation, providing accurate 6D poses of objects. The BEHAVE dataset focuses on tracking human–object interactions. Although it does not directly provide the human center position, it includes the 3D coordinates of the left hip joint H_{hip_l} and right hip joint H_{hip_r}. We derived the ground-truth human center position H_c as the average of the two hip joints:

$$H_c = \frac{H_{hip\_l} + H_{hip\_r}}{2} \tag{11}$$

The evaluation metrics were defined with reference to 6D pose estimation [32], including the position error T_e and the Average Distance (ADD) metric. The position error T_e is defined as the Euclidean distance between the estimated and ground-truth center points. The ADD metric uses position error T_e and considers a prediction correct if it is smaller than a predefined threshold, set to 10% of the 3D model diameter.

Table 1. RESULTS OF RMSE OF ATE [m] ON TUM RGB-D STATIC DATASETS. **BEST** IS BOLD; SECOND-BEST IS UNDERLINED.

Sequence	ORB-SLAM3	Point-SLAM	GS-SLAM	**Ours**	Improvement
fr1_desk	0.0186	0.0434	0.0150	0.0193	–
fr1_desk2	0.0296	0.0454	–	0.0246	16.89%
fr1_room	0.0695	0.3092	–	0.0296	57.41%
fr2_xyz	0.0038	0.0131	0.0144	0.0034	10.53%
fr3_office	0.0104	0.0348	0.0149	0.0066	36.54%

Table 2. RESULTS OF RMSE OF ATE [m] ON TUM RGB-D DYNAMIC DATASETS. **BEST** IS BOLD; SECOND-BEST IS UNDERLINED.

Sequence	ORB-SLAM3	Dynamic-VINS	ViQu-SLAM	**Ours**	Improvement
fr3_walking_xyz	0.4048	0.0486	0.1258	0.0885	78.14%
fr3_walking_static	0.0176	0.0077	0.0124	0.0135	23.30%
fr3_walking_rpy	0.2676	0.0629	0.0720	0.0307	88.53%
fr3_walking_half	0.3920	0.0608	0.0261	0.0573	85.38%

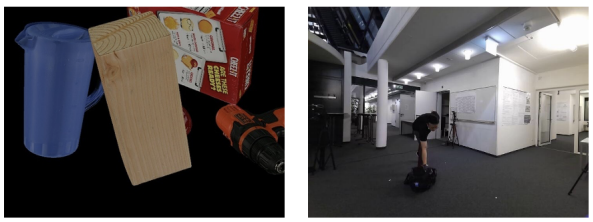

Figure 5. Examples from benchmark datasets of target localization. (Left) YCB-Video dataset, which contains objects with significant occlusions. (Right) BEHAVE dataset, which includes humans in diverse poses.

Table 3. RESULTS OF ADE [m] AND FDE [m] ON ETH/UYC DATASETS. **BEST** IS BOLD; SECOND-BEST IS UNDERLINED.

Method	ETH		HOTEL		UNIV		ZARA1		ZARA2		AVG	
	ADE↓	FDE↓	ADE↓	FDE↓	ADE↓	FDE↓	ADE↓	FDE↓	ADE↓	FDE↓	ADE↓	FDE↓
PPT [18]	0.36	0.49	0.11	0.14	0.22	0.40	0.17	0.30	0.18	0.33	0.21	0.33
NPSN [2]	0.36	0.59	0.16	0.25	0.23	0.39	0.18	0.32	0.14	0.25	0.21	0.36
MID [12]	0.39	0.66	0.13	0.22	0.22	0.45	0.17	0.30	0.13	0.27	0.21	0.38
ET [3]	0.36	0.53	0.12	0.19	0.24	0.43	0.19	0.33	0.14	0.24	0.21	0.34
LED [22]	0.39	0.58	0.11	0.17	0.26	0.43	0.18	0.26	0.13	0.22	0.21	0.33
MART [15]	0.35	0.47	0.14	0.22	0.25	0.45	0.17	0.29	0.13	0.22	0.21	0.33
TUTR [26]	0.40	0.61	0.11	0.18	0.23	0.42	0.18	0.34	0.13	0.25	0.21	0.36
TrajFine[29]	0.35	0.60	0.11	0.18	0.22	0.48	0.15	0.30	**0.12**	0.25	0.19	0.36
V^2-Net [31]	**0.23**	**0.37**	0.11	0.16	0.21	0.35	0.19	0.30	0.14	0.24	0.18	0.28
V^2-Net+TR [38]	0.26	**0.37**	0.11	0.15	0.19	0.34	0.17	0.30	0.13	0.23	0.17	0.28
Ours	0.27	0.40	**0.08**	**0.13**	**0.15**	**0.23**	**0.12**	**0.17**	0.13	**0.21**	**0.15**	**0.23**
Improvement	25.00%	18.37%	27.27%	7.14%	31.82%	42.50%	29.41%	43.33%	27.78%	36.36%	28.57%	30.30%

Table 4. RESULTS OF T_e [m] AND ADD [%] ON YCB-Video DATASETS. **BEST** RESULTS ARE BOLD; SECOND-BEST IS UNDERLINED.

Object	Baseline		**Ours**		Improvement	
	$T_e \downarrow$	ADD ↑	$T_e \downarrow$	ADD ↑	$T_e \downarrow$	ADD↑
003_cracker_box	0.0672	4.83	**0.0350**	**63.13**	47.92%+58.30%	
004_sugar_box	0.0459	0.48	**0.0204**	**80.73**	55.56%+80.25%	
008_pudding_box	0.0454	2.26	**0.0182**	**71.32**	59.91%+60.06%	
009_gelatin_box	0.0324	3.56	**0.0165**	**81.05**	49.07%+77.49%	
036_wood_block	0.0673	0.67	**0.0414**	**42.59**	38.48%+41.92%	
061_foam_brick	0.0372	**99.50**	**0.0281**	96.72	24.46%	–

Table 5. RESULTS OF T_e [m] AND ADD [%] ON BEHAVE DATASETS. **BEST** RESULTS ARE BOLD; SECOND-BEST IS UNDERLINED.

Sequence	Baseline		**Ours**		Improvement	
	$T_e \downarrow$	ADD ↑	$T_e \downarrow$	ADD ↑	$T_e \downarrow$	ADD↑
01_backpack_back	0.1993	65.85	**0.1383**	**78.57**	30.61%+12.72%	
01_basketball	0.2250	68.89	**0.1324**	**80.85**	41.16%+11.96%	
01_boxtiny_hand	0.2023	60.00	**0.1648**	**70.83**	18.54%+10.83%	
01_toolbox	0.2009	73.33	**0.1471**	**74.47**	26.78%+1.14%	
01_yogamat_hand	0.2470	68.75	**0.1497**	**72.92**	39.39%+1.17%	

YCB-Video Dataset Quantitative results of object localization on the YCB-Video dataset are provided in Table 4. Our method achieved lower position errors T_e and higher ADD scores than the baseline on most objects. For example, on *004_sugar_box*, our method improved ADD by 80.25%, while on *009_gelatin_box*, the position error was reduced by nearly half and the ADD increased by 77.49%. On the other hand, for *061_foam_brick*, the ADD metric did not show improvement, and the baseline method already achieved a position error of 0.0372 meters. We note that due to its simple geometry and limited occlusion, the 2D baseline was sufficient for handling such simple cases. However, in more complex scenarios with occlusions, our point cloud–based method demonstrates notable advantages, as shown in Fig. 5.

BEHAVE Dataset The person localization performance on the BEHAVE dataset is reported in Table 5. In this dataset, which focuses on human-object interaction scenarios, our method consistently achieved improvements across all sequences. For example, on the *01_basketball* sequence, the position error T_e was 0.1324 meters, representing a 41.16% reduction compared to the baseline, while the ADD improved by 11.96%. On the *01_yogamat_hand* sequence, the position error T_e decreased by 39.39%. The BEHAVE dataset involves humans performing diverse postures, such as bending, leaning, and crouching, as shown in Fig. 5. These results demonstrate the adaptability of our method to various human poses in complex interaction scenarios.

The above results confirm the effectiveness of our proposed point cloud–based method for dynamic target localization. In addition to human-centered scenarios, we also evaluated on an object dataset to extend the diversity of dynamic targets. These results highlight the potential of our point cloud–based approach to generalize across diverse dynamic targets and to remain robust in realistic scenarios involving occlusions and complex human object interactions.

4.3. Trajectory Refinement for Person Prediction Trajectory

For the trajectory prediction module, we used PPT [18] as the baseline predictor and applied our Residual-DtACI net-

work for trajectory refinement. The network structure and parameters were kept the same as in the robot pose trajectory refinement experiment. We evaluated the quantitative benefits of our method on the ETH/UCY dataset, in comparison with a wide range of current state-of-the-art approaches. The ETH [24]/UCY [14] dataset is a widely used benchmark for human trajectory prediction, consisting of two datasets with five different scenes, ETH and HOTEL from the ETH dataset, and UNIV, ZARA1, and ZARA2 from the UCY dataset.

For evaluation, we followed the previous trajectory prediction metrics: Average Displacement Error (ADE) and Final Displacement Error (FDE). ADE computes the mean Euclidean distance between the predicted and ground-truth positions across all time steps in the trajectory, thereby reflecting the overall prediction accuracy. In contrast, FDE measures the Euclidean distance between the predicted final position and the ground-truth final destination, capturing the accuracy of the final destination prediction. We followed the standard setting by taking the past 8 timestamps (3.2s) as the observed trajectory and predicting the subsequent 12 timestamps (4.8s) as the future trajectory.

ETH/UYC Dataset As shown in Table 3, we compared our method with ten state-of-the-art trajectory prediction methods. Our method achieved the best prediction performance, with an average ADE of 0.15 meters and an average FDE of 0.23 meters across all scenes. Per secens results remain competitive, and are best in several cases. For example, in the *HOTEL* scene, our method achieved an ADE of 0.08 meters and an FDE of 0.13 meters; in the *UNIV* scene, an ADE of 0.15 meters and an FDE of 0.23 meters; and in the *ZARA* scenes, an ADE of 0.12 meters and an FDE of 0.17 meters. As a trajectory refinement network, our approach demonstrated significant improvements over the baseline PPT model, reducing ADE from 0.21 to 0.15 meters (a 28.57% improvement) and FDE from 0.33 to 0.23 meters (a 30.30% improvement).

The experiment of trajectory prediction further demonstrates the effectiveness of our Residual-DtACI network for trajectory refinement. When applied to trajectories affected by different sources of error, our network was able to significantly reduce these errors. These results validate that the Residual-DtACI network provides a robust refinement mechanism for trajectory prediction tasks, achieving consistent improvements over the baseline as well as competitive or superior performance compared to state-of-the-art methods.

4.4. Ablation Study

To further validate the overall effectiveness of our framework, we conducted an ablation study on THUD++ [17] dataset. In this study, we removed the error reduction methods from each module of our framework to evaluate their individual contributions. (a. Pose w/o Residual-DtACI) The robot pose trajectory was directly obtained from ORB-SLAM3 without refinement by Residual-DtACI; (b. Loc. w/o Point Cloud) Dynamic target localization was performed using the baseline method; (c. Pred. w/o Residual-DtACI) Future trajectories of human were predicted using PPT without refinement by Residual-DtACI. As shown in Table 6, the full framework achieves the best performance, while removing any component leads to a noticeable degradation.

Table 6. RESULTS OF ABLATION STUDY ON THUD++ DATASET. **BEST** RESULTS ARE BOLD; SECOND-BEST IS UNDERLINED.

Model Ablation	ADE↓	Degrad.	FDE↓	Degrad.
Pose w/o Residual-DtACI	0.2595	16.26%	0.2163	-0.1645
Loc. w/o point cloud	0.2904	30.11%	0.1762	-0.1244
Pred. w/o Residual-DtACI	0.3228	44.62%	0.4042	-0.3524
Ours	**0.2232**	–	**0.0518**	–

4.5. Evaluation on a Quadruped Robot

We implemented our framework on a Unitree Go2 robot with an onboard NVIDIA Jetson Orin computer and an Intel RealSense D435i depth camera. For the perception module, both the baseline components and our proposed optimization methods were directly deployed. For the path planning module, we adopted a cost map–based approach with the Fast Marching Method (FMM) as the foundation [8, 37], and further enhanced the navigation capability by incorporating local search in the command generator. The final linear and angular velocity commands were then transmitted to the Unitree Go2 through its high-level SDK. We conducted experiments in a laboratory environment, where the robot was assigned multiple fixed targets while a person approached it at different speeds.

5. Conclusion and Future work

This paper proposed a framework that systematically addresses accumulated perception errors in dynamic obstacle avoidance scenarios. By integrating our proposed Residual-DtACI network for trajectory refinement and the proposed point cloud–based method for target localization, our approach reduces errors across different perception modules. It enhances the performance of dynamic autonomous navigation. Future work will extend the framework to outdoor environments while considering the impact of illumination changes. Evaluations on benchmark datasets and real-world tests on the Unitree Go2 robot verify the effectiveness of the proposed framework.

References

[1] Inhwan Bae, Jin-Hwi Park, and Hae-Gon Jeon. Learning pedestrian group representations for multi-modal trajectory prediction. In *European Conference on Computer Vision*, pages 270–289. Springer, 2022. 2

[2] Inhwan Bae, Jin-Hwi Park, and Hae-Gon Jeon. Non-probability sampling network for stochastic human trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6477–6487, 2022. 7

[3] Inhwan Bae, Jean Oh, and Hae-Gon Jeon. Eigentrajectory: Low-rank descriptors for multi-modal trajectory forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10017–10029, 2023. 7

[4] Bharat Lal Bhatnagar, Xianghui Xie, Ilya A Petrov, Cristian Sminchisescu, Christian Theobalt, and Gerard Pons-Moll. Behave: Dataset and method for tracking human object interactions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15935–15946, 2022. 6

[5] Carlos Campos, Richard Elvira, Juan J Gómez Rodríguez, José MM Montiel, and Juan D Tardós. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE transactions on robotics*, 37(6):1874–1890, 2021. 3

[6] Sehwan Choi, Jungho Kim, Junyong Yun, and Jun Won Choi. R-pred: Two-stage motion prediction via tube-query attention-based trajectory refinement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8525–8535, 2023. 2

[7] Jakob Engel, Thomas Schöps, and Daniel Cremers. Lsd-slam: Large-scale direct monocular slam. In *European conference on computer vision*, pages 834–849. Springer, 2014. 1, 2

[8] Zipeng Fu, Ashish Kumar, Ananye Agarwal, Haozhi Qi, Jitendra Malik, and Deepak Pathak. Coupling vision and proprioception for navigation of legged robots. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17273–17283, 2022. 8

[9] Xiang Gao, Rui Wang, Nikolaus Demmel, and Daniel Cremers. Ldso: Direct sparse odometry with loop closure. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2198–2204. IEEE, 2018. 1, 2

[10] Isaac Gibbs and Emmanuel J Candès. Conformal inference for online prediction with arbitrary distribution shifts. *Journal of Machine Learning Research*, 25(162):1–36, 2024. 2, 4

[11] Carlos Gómez-Huélamo, Marcos V Conde, Rafael Barea, and Luis M Bergasa. Improving multi-agent motion prediction with heuristic goals and motion refinement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5323–5332, 2023. 2

[12] Tianpei Gu, Guangyi Chen, Junlong Li, Chunze Lin, Yongming Rao, Jie Zhou, and Jiwen Lu. Stochastic trajectory prediction via motion indeterminacy diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17113–17122, 2022. 7

[13] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. Ultralytics yolo, 2023. [Online; accessed 11-Sep-2025]. 2, 3

[14] Laura Leal-Taixé, Michele Fenzi, Alina Kuznetsova, Bodo Rosenhahn, and Silvio Savarese. Learning an image-based motion context for multiple people tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3542–3549, 2014. 8

[15] Seongju Lee, Junseok Lee, Yeonguk Yu, Taeri Kim, and Kyoobin Lee. Mart: Multiscale relational transformer networks for multi-agent trajectory prediction. In *European Conference on Computer Vision*, pages 89–107. Springer, 2024. 2, 7

[16] Chengyang Li, Yulai Zhang, Zhiqiang Yu, Xinming Liu, and Qing Shi. A robust visual slam system for small-scale quadruped robots in dynamic environments. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 321–326. IEEE, 2024. 1, 2, 5

[17] Zeshun Li, Fuhao Li, Wanting Zhang, Zijie Zheng, Xueping Liu, Yongjin Liu, and Long Zeng. Thud++: Large-scale dynamic indoor scene dataset and benchmark for mobile robots. *arXiv preprint arXiv:2412.08096*, 2024. 5, 8

[18] Xiaotong Lin, Tianming Liang, Jianhuang Lai, and Jian-Fang Hu. Progressive pretext task learning for human trajectory prediction. In *European Conference on Computer Vision*, pages 197–214. Springer, 2024. 1, 2, 3, 7

[19] Mengmeng Liu, Hao Cheng, Lin Chen, Hellward Broszio, Jiangtao Li, Runjiang Zhao, Monika Sester, and Michael Ying Yang. Laformer: Trajectory prediction for autonomous driving with lane-aware scene constraints. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2039–2049, 2024. 2

[20] Yunfei Luan, Muhang He, Yudong Tian, Chengjie Lin, Yunhan Fang, Zihao Zhao, Jianxin Yang, and Yao Guo. Intelligent disinfection robot with high-touch surface detection and dynamic pedestrian avoidance. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3595–3601. IEEE, 2024. 2, 5, 6

[21] Karttikeya Mangalam, Harshayu Girase, Shreyas Agarwal, Kuan-Hui Lee, Ehsan Adeli, Jitendra Malik, and Adrien Gaidon. It is not the journey but the destination: Endpoint conditioned trajectory prediction. In *European conference on computer vision*, pages 759–776. Springer, 2020. 1, 2

[22] Weibo Mao, Chenxin Xu, Qi Zhu, Siheng Chen, and Yanfeng Wang. Leapfrog diffusion model for stochastic trajectory prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5517–5526, 2023. 7

[23] Hidenobu Matsuki, Riku Murai, Paul HJ Kelly, and Andrew J Davison. Gaussian splatting slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18039–18048, 2024. 1, 2, 5

[24] Stefano Pellegrini, Andreas Ess, Konrad Schindler, and Luc Van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *2009 IEEE 12th international conference on computer vision*, pages 261–268. IEEE, 2009. 8

[25] Erik Sandström, Yue Li, Luc Van Gool, and Martin R Oswald. Point-slam: Dense neural point cloud-based slam. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18433–18444, 2023. 1, 2, 5

[26] Liushuai Shi, Le Wang, Sanping Zhou, and Gang Hua. Trajectory unified transformer for pedestrian trajectory prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9675–9684, 2023. 7

[27] Shaoshuai Shi, Li Jiang, Dengxin Dai, and Bernt Schiele. Motion transformer with global intention localization and local movement refinement. *Advances in Neural Information Processing Systems*, 35:6531–6543, 2022. 2

[28] Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of rgb-d slam systems. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 573–580. IEEE, 2012. 5

[29] Kuan-Lin Wang, Li-Wu Tsao, Jhih-Ciang Wu, Hong-Han Shuai, and Wen-Huang Cheng. Trajfine: Predicted trajectory refinement for pedestrian trajectory forecasting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4483–4492, 2024. 3, 7

[30] Rui Wang, Martin Schworer, and Daniel Cremers. Stereo dso: Large-scale direct sparse visual odometry with stereo cameras. In *Proceedings of the IEEE international conference on computer vision*, pages 3903–3911, 2017. 1, 2

[31] Conghao Wong, Beihao Xia, Ziming Hong, Qinmu Peng, Wei Yuan, Qiong Cao, Yibo Yang, and Xinge You. View vertically: A hierarchical network for trajectory prediction via fourier spectrums. In *European Conference on Computer Vision*, pages 682–700. Springer, 2022. 7

[32] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *arXiv preprint arXiv:1711.00199*, 2017. 6

[33] Linhui Xiao, Jinge Wang, Xiaosong Qiu, Zheng Rong, and Xudong Zou. Dynamic-slam: Semantic monocular visual localization and mapping based on deep learning in dynamic environment. *Robotics and Autonomous Systems*, 117:1–16, 2019. 1, 2, 5

[34] Chenxin Xu, Maosen Li, Zhenyang Ni, Ya Zhang, and Siheng Chen. Groupnet: Multiscale hypergraph neural networks for trajectory prediction with relational reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6498–6507, 2022. 2

[35] Chenxin Xu, Weibo Mao, Wenjun Zhang, and Siheng Chen. Remember intentions: Retrospective-memory-based trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6488–6497, 2022. 1, 2

[36] Qi Yin, Yang Ye, Meida Chen, and Yangming Shi. *A Teacher–Student Learning Approach to Improve Quadruped Robots' Autonomous Locomotion and Obstacle Avoidance on Construction Sites*, pages 572–581. 1

[37] Qi Yin, Ziming Liu, Meida Chen, Yang Ye, Jing Du, and Yangming Shi. A navigation system to improve quadruped robot's autonomous path planning and locomotion on outdoor construction sites. *Available at SSRN 5907605*, 2026. 8

[38] Hanbit Yoon, Usman Ali, Joonhee Choi, and Eunbyung Park. Rethinking convolutional neural networks for trajectory refinement. *Pattern Recognition*, 157:110883, 2025. 2, 7

[39] Chao Yu, Zuxin Liu, Xin-Jun Liu, Fugui Xie, Yi Yang, Qi Wei, and Qiao Fei. Ds-slam: A semantic visual slam towards dynamic environments. In *2018 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 1168–1174. IEEE, 2018. 1, 2

[40] Yang Zhou, Hao Shao, Letian Wang, Steven L Waslander, Hongsheng Li, and Yu Liu. Smartrefine: A scenario-adaptive refinement framework for efficient motion prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15281–15290, 2024. 2

[41] Zikang Zhou, Jianping Wang, Yung-Hui Li, and Yu-Kai Huang. Query-centric trajectory prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17863–17873, 2023. 2