Active Multi-Object Tracking for 3D Reconstruction with Hierarchical Reinforcement Learning

Heng Li and Cheng Cai*
* Shanghai DianJi University, China
E-mail: 23600003110510@st.sdju.edu.cn Tel: +86-17263128270
Shanghai DianJi University, China
E-mail: caic@sdju.edu.cn Tel: +86-17349738897

***Abstract*— This study addresses the problem of active multi-object tracking in 3D environment, with a particular focus on application in 3D reconstruction. The objective is to achieve continuous tracking of multiple targets by dynamically controlling the poses of multiple cameras in real time. Unlike existing approaches, this work is conducted in a 3D environment populated with multiple cameras and targets, aiming to maximize the minimum 2-coverage (M2C) and the triangulation angle. Such multi-view coverage is crucial for enhancing the accuracy of 3D reconstruction. To this end, we propose a hierarchical reinforcement learning framework that decomposes the task into a two-level policy structure. A high-level coordinator assigns targets to low-level executors with the goal of optimizing the M2C and the triangulation angle, while each low-level executor is responsible for tracking its assigned targets by minimizing the relative angle to them. Experimental results in a custom-built 3D simulation environment demonstrate that the proposed method substantially improves coverage efficiency compared to baseline approaches.**

I. INTRODUCTION

With the rapid advancements in computer vision and robotics, active object tracking (AOT) has emerged as a critical research area [1]. Unlike traditional passive tracking methods, AOT requires an agent to actively maintain continuous observation of moving targets by dynamically controlling camera positions in real time. However, most of the existing research has been limited to single target tracking scenarios. In recent years, driven by the increasing demand for 3D reconstruction technologies, active multi-object tracking (AMOT) has gained attention as a fundamental prerequisite, demonstrating significant research value and application potential [2].

This paper focuses on the problem of Active Multi-Object Tracking for 3D Reconstruction (AMOT-3DR), wherein a group of cameras collaboratively adjust their poses to continuously track multiple moving targets. The central objective is to maximize the M2C—ensuring that each target is simultaneously covered by at least two cameras. Furthermore, for each target, it is required that maximize the triangulation angle to enhance the geometric accuracy of 3D reconstruction.

AMOT-3DR poses substantial challenges, primarily in the following three aspects. First, in contrast to single target AOT, AMOT-3DR involves the collaborative tracking of multiple targets by multiple cameras. The increased number of agents and targets leads to an exponential growth in the dimensionality of both the state and action spaces, greatly increasing decision complexity. Second, compared to general AMOT tasks, AMOT-3DR introduces stricter coverage constraints: each target must be observed by at least two cameras from diverse viewpoints. These constraints significantly complicate the design of effective coordination strategies. Third, the task is guided by a global objective that is difficult to decompose into local decisions, making it challenging for individual cameras to take optimal actions and achieve efficient collaboration.

To address these challenges, we propose a framework based on hierarchical reinforcement learning, employing a two-level decision-making architecture consisting of a centralized coordinator (high-level policy) and multiple distributed executors (low-level policies). Specifically, the high-level coordinator is responsible for dynamically and efficiently assigning targets to executors, aiming to optimize system-wide M2C. Each executor focuses on tracking its assigned targets by minimizing the relative observation angle between itself and the targets. Through this hierarchical decomposition, the complex AMOT-3DR task is effectively divided into two simpler subtasks, thereby reducing the overall optimization complexity. Both the coordinator and the executors can be trained using standard single-agent reinforcement learning algorithms to achieve their respective objectives.

II. RELATED WORK

A. Single-Target Active Tracking

Single-target active object tracking has made considerable progress in recent years. The earliest deep reinforcement learning-based method was proposed in [3], where raw camera frames are fed into convolutional and recurrent neural networks to output discrete control actions. This end-to-end architecture demonstrated strong performance in terms of accumulated reward and episode length. To enhance robustness, an adversarial framework was introduced in [4], where the tracker and the target form a zero-sum game. The target learns to maximize its chance of escape, thereby forcing the tracker to

979-8-3315-7206-8/25/$31.00 ©2025 IEEE

handle more diverse behaviors. In [5], a multi-camera collaborative tracking strategy was presented, combining pose-based and vision-based control schemes. Each camera autonomously selects the appropriate control mode based on its current observation. A more generalizable and robust tracker was proposed in [6], where a structure-aware motion representation was built by reconstructing the environment and predicting the target's trajectory. Efficiency and coordination in multi-agent systems were addressed in [7], which combined multi-agent deep reinforcement learning with a mixture-of-experts (MoE) model, enabling effective cooperation among agents while reducing computational load. Most of the above methods focus solely on single-target scenarios. In contrast, our work explores the more complex setting of multi-target tracking with multiple active cameras

B. Active Multi-Target Tracking

Research in active multi-object tracking is gaining momentum. A hierarchical coordination framework was introduced in [8], where a high-level coordinator assigns targets and low-level agents control the cameras to maintain tracking. To facilitate coordination, [9] proposed a communication protocol enabling agents to learn with whom and when to exchange state information. In [10], the problem was extended to realistic environments with motion constraints; Monte Carlo Tree Search (MCTS) was applied alongside target motion prediction to improve planning under limited mobility. To address the instability caused by non-stationary policies and inter-agent interference, [11] proposed a hierarchical reinforcement learning strategy with reward shaping and target filtering modules. A more practical approach was demonstrated in [12], where agents receive real-time image inputs in a 3D simulated environment, allowing policies to learn directly from visual observations.

C. Distinction from Previous Studies

While existing AMOT research has led to effective tracking strategies, most focus on maximizing general visibility or coverage. However, in 3D reconstruction, this is not sufficient. Accurate reconstruction requires that each target be simultaneously observed by at least two cameras. Moreover, these observations must come from distinct directions to ensure strong geometric constraints. This work differs from prior efforts in two key aspects. First, instead of maximizing general coverage, we explicitly optimize M2C as the core objective. Second, we impose a multi-view geometric constraint requiring angular disparity between observing cameras. This is based on the triangulation principle [13]: accurate 3D position estimation depends on the intersection of viewing rays from multiple cameras. When cameras observe from similar angles, even minor image or calibration errors can lead to significant deviations in estimated positions. Greater angular separation improves numerical stability and reconstruction accuracy. As illustrated in Fig. 1, although both camera pairs (1,2) and (1,3) satisfy the M2C condition, pair (1,3) provides better reconstruction results due to its wider triangulation angle and superior geometric constraint.

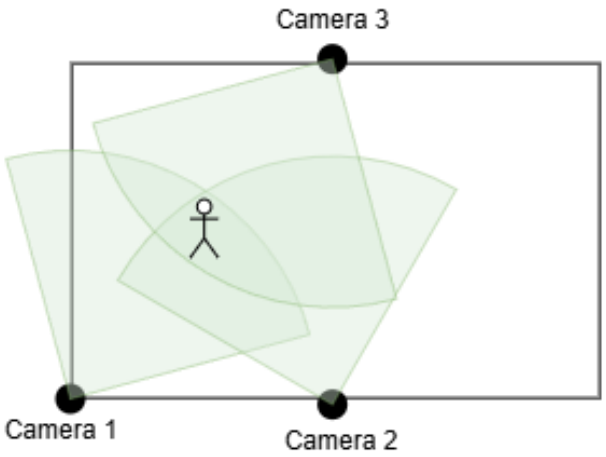

Fig. 1 Diagram explaining why triangulation angle constraints are needed.

III. METHOD

A. Three-Dimensional Simulation Environment

To effectively simulate and solve the problem of AMOT-3DR, we constructed a customized 3D virtual simulation environment for data collection and model training. As illustrated in Fig. 2, the environment consists of a bounded 3D space where multiple cameras are fixed along the boundary. Each camera can rotate horizontally and vertically to follow moving targets. Multiple targets move continuously between randomly generated path points, and after arriving at the current target point, they immediately randomly select the next target point and continue moving.

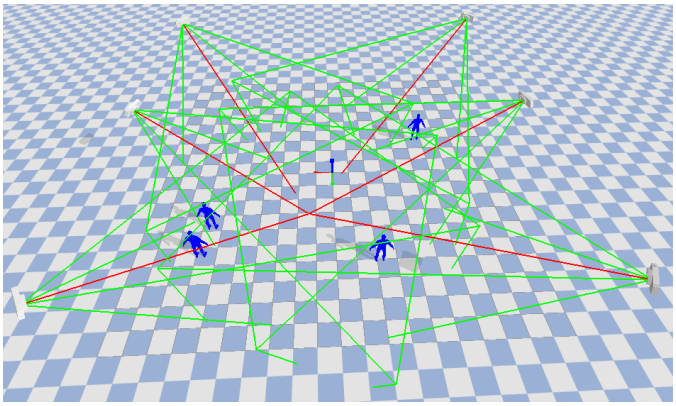

Fig. 2 Simulation environment for training and evaluating the proposed method.

The state space of the environment primarily encodes the relative spatial relationships between cameras and targets. Specifically, the relative position between camera i and target j is defined as $rp_{i,j} = (pitch_{i,j}, yaw_{i,j}, distance_{i,j})$. Here, pitch and yaw denote the vertical and horizontal angular differences between the camera and the target, respectively, and

distance is their Euclidean separation. The global state space is then represented as $S = (rp_{1,1}, \cdots, rp_{1,m}, rp_{2,1}, \cdots, rp_{n,m})$ where n and m are the total number of cameras and targets.

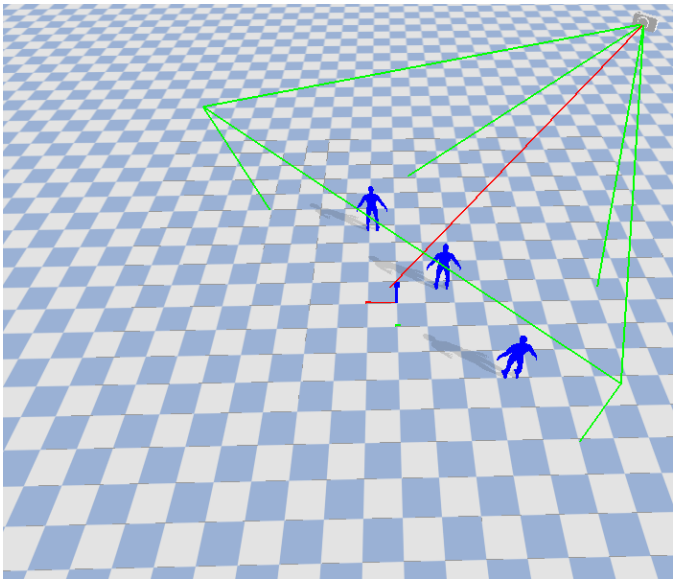

Fig. 3 Simulation environment for training and evaluating executor.

B. Problem Statement

We formulate the AMOT-3DR task as a Partially Observable Markov Decision Process (POMDP) [14], defined by the tuple $\langle S, O, A, R, P, Z, \gamma \rangle$, where:

S is the state space.

O is the observation space.

A is the action space.

R is the reward function.

P denotes state transition probabilities.

Z is the observation model.

γ is the discount factor.

At each time step t, camera i receives a partial observation $o_{t,i} \in O$ with probability $Z(o_{t,i}|s_t)$, where $s_t \in S$ is the global state. The agent selects a joint action $a_t \in A$ according to a policy $\pi(a_t|o_t)$, receives a reward $r_t = R(s_t, a_t)$, and transitions to the next state $s_{t+1} \sim P(s_{t+1}|s_t, a_t)$. The objective is to maximize the expected cumulative reward: $\mathbb{E}[\sum_{t=1}^{T} \gamma^t r_t]$.

C. Overall Architecture

Our method adopts a two-level hierarchical architecture, as shown in Fig. 4. The framework includes a high-level coordinator and multiple low-level executors, where all executors share model parameters. The coordinator allocates targets to each executor based on the global state, and each executor then attempts to optimally track the assigned targets using its local observations.

The executor module maintains multiple tracking models to handle different numbers of assigned targets. First, the coordinator processes the global state S and outputs a target allocation strategy. Based on this allocation, a suitable tracking model is selected for the executor. For example, if camera i is assigned to track targets j and k, a dual-target model is used. The executor's input is defined as $O_i = (rp_{i,j}, rp_{i,k})$, from which the agent outputs an action. All individual camera actions are aggregated into a joint action, which is then executed in the environment to obtain the next state and reward.

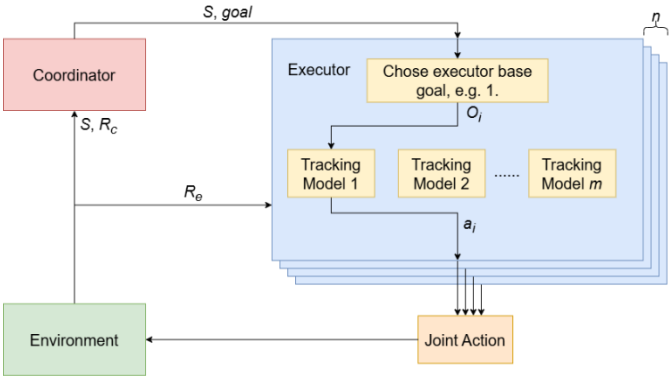

Fig. 4 Overall architecture of the proposed method.

D. Coordinator

The goal of the coordinator is to learn an optimal target allocation strategy that maximizes the M2C of the targets, while also maximizing the triangulation angle for each target to enhance the accuracy of 3D reconstruction. The coordinator needs to master global information to reasonably allocate targets to executors, so its observation space is $O^c = S$. The action space is $goal = (g_{1,1}, g_{1,2}, \cdots, g_{1,m}, g_{2,1}, \cdots, g_{n,m})$, where $g_{i,j}$ indicates whether the target j is allocated to the target i, 0 and 1 represent non-allocation and allocation. Respectively. In order to better complete the allocation task, we designed a suitable reward function. The reward function of the coordinator is Equation (1).

$$r^c = \begin{cases} \frac{1}{m}\sum_{j=1}^{m} I_j - L_j & (a) \\ -0.1 & (b) \end{cases} \tag{1}$$

$$L_j = \begin{cases} \frac{\theta_j - 90}{90} & \theta_j < 90 \\ 0 & \theta_j \geq 90 \end{cases} \tag{2}$$

Among them, $I_j = 1$ if target j is covered by two or more cameras; otherwise, $I_j = 0$. L_j represents the penalty item about the camera angle of view, θ_j represents the maximum triangulation angle among all cameras assigned to the target j. The special case b is that no target is covered, in which case a penalty item is introduced.

E. Executor

The goal of the executor is to cover as many target objects as possible assigned by the coordinator. Since different cameras may be assigned to different numbers of targets, we train a separate tracking model for each number of targets, so that the executor has the ability to handle all numbers of targets. Multiple executors in the environment share the same set of model parameters.

After receiving the target assigned by the coordinator, the executor takes the required observations from $O_i^e = \{rp_{i,k} | k \in goal_i\}$ the global state through a filter module and selects the appropriate tracking model. For example, if the second

executor is assigned to the first, third, and fifth target objects, it will observe $O_2^e = [rp_{2,1}, rp_{2,3}, rp_{2,5}]$and select the tracking model that can handle the three target objects. The executor operates in two degrees of freedom, each with three discrete actions: turn left (or up), stay still, and turn right (or down), resulting in a total of 9 possible discrete actions.

The executor not only needs to cover the target object but also needs to keep the target object in the center of the field of view as much as possible to improve the accuracy of 3D reconstruction. In addition, it is also necessary to reduce energy consumption, that is, reduce the number of camera rotations. In summary, the reward function of the executor is Equation (3):

$$r_i^e = \frac{1}{m_i}\sum_{j=1}^{m_i} r_{i,j}^h + r_{i,j}^v - \beta cost_i \tag{3}$$

$$r_{i,j}^h = \begin{cases} 1 - \frac{|\alpha_{i,j}^h|}{\alpha_{max}^h} & |\alpha_{i,j}^h| < \alpha_{max}^h \\ -1 & |\alpha_{i,j}^h| \geq \alpha_{max}^h \end{cases} \tag{4}$$

$$r_{i,j}^v = \begin{cases} 1 - \frac{|\alpha_{i,j}^v|}{\alpha_{max}^v} & |\alpha_{i,j}^v| < \alpha_{max}^v \\ -1 & |\alpha_{i,j}^v| \geq \alpha_{max}^v \end{cases} \tag{5}$$

Where m_i represents the number of target objects assigned to the i executor, $\alpha_{i,j}^h$ represents the relative angle between the executor and the target object in the horizontal direction, $\alpha_{i,j}^v$ represents the relative angle between the executor and the target object in the vertical direction, α_{max}^h and α_{max}^v represent the maximum viewing angle of the camera in the horizontal and vertical directions respectively, $r_{i,j}^h$ and $r_{i,j}^v$ represent the reward values in the horizontal and vertical directions respectively. $cost_i$ refers to the energy consumption of rotation. When there are actions in both directions, its value is 2. When there is action in only one direction, the value is 1, otherwise it is 0. β is the weight of energy consumption, usually set to 0.01.

IV. Experiments

A. Training Strategy

To enhance the stability and efficiency of the training process, we adopt a phased training strategy. Jointly training the coordinator and executors from the outset can lead to mutual interference, especially in the early stages when both components are still immature. Specifically, an under-trained executor may fail to effectively track the targets assigned by the coordinator, resulting in inaccurate or misleading feedback signals for the coordinator. Conversely, a poorly initialized coordinator may assign targets inappropriately—such as assigning widely separated targets to a single executor or assigning targets outside a camera's effective field of view—which hinders the executor's learning process.

The phased training strategy mitigates this issue by decoupling the training of the two components. In the first phase, we train each executor tracking model independently, as shown in Fig. 3. When training an executor tracking model designed to track k targets, we place k targets into the environment and assign all of them to the corresponding camera by default. The objective of this phase is to ensure that each executor tracking model learns to effectively track and cover the assigned targets under varying target quantities.

Once all executor models reach stable performance, we freeze their parameters and proceed to train the coordinator. At this stage, the coordinator interacts with a reliable set of executors. As a result, the quality of the coordinator's target assignment strategy can be more directly and accurately evaluated based on the resulting tracking coverage, thus avoiding distorted reward signals caused by underperforming executors.

B. Experimental Setup

We utilize the previously described simulation environment for both training and evaluation. The specific experimental settings are as follows: the environment is a 3D space with dimensions 100×100 units. All cameras are mounted at a fixed height of 50 units. Each camera has a maximum viewing distance of 70 units, meaning it cannot observe regions located on the opposite boundary of the environment. The horizontal and vertical fields of view (FoV) of each camera are set to 60° and 33.75°, respectively. These narrower FoV values, compared to standard camera settings, are chosen to mitigate the impact of distortion that occurs when targets appear near the edge of the image, which can significantly degrade the accuracy of 3D reconstruction. As a simplification, edge views are discarded entirely. Each camera can rotate horizontally and vertically in increments of 5° per action.

Existing metrics such as basic coverage rate or minimal two-coverage rate are insufficient to directly and reliably assess the contribution of control strategies to the quality of 3D reconstruction. Therefore, we introduce a novel evaluation metric, termed View-Discriminative Minimal 2-Coverage (VDM2C), which incorporates both the coverage quantity and the triangulation angle of cameras. As shown in the Equation (6) This metric better reflects the practical requirements of accurate and robust 3D reconstruction.

$$VDM2C = \frac{1}{m}\sum_{j=1}^{m} I_j - L_j \tag{6}$$

Among them, $I_j = 1$ if target j is covered by two or more cameras; otherwise, $I_j = 0$. L_j is calculated by Equation (2).

C. Experimental Results

To verify the effectiveness of the proposed hierarchical reinforcement learning framework (referred to as Ours), we compare it with the following baseline methods: Greedy algorithm (Greedy) makes each camera try to cover all targets within its current viewing range and head towards the geometric center of these targets. Multi-agent Deep Deterministic Policy Gradient Algorithm (MADDPG) [15].

Multi-agent version of the proximal policy optimization algorithm (MAPPO) [16]. In order to comprehensively evaluate the performance of the algorithm and verify its robustness, we conducted experiments under various combinations of camera and target numbers. The experimental results are evaluated using VDM2C (%).

TABLE I. VDM2C OF EACH METHOD

n	m	Greedy	MADDPG	MAPPO	Ours
4	5	34.74±5.32	50.65±9.51	55.48±8.03	67.37±5.32
4	6	34.59±5.46	50.49±9.12	53.21±8.05	65.59±5.46
4	7	33.62±5.81	49.62±9.59	52.80±9.53	64.34±5.81
6	5	40.88±4.43	52.54±9.04	56.33±7.29	74.82±4.43
6	6	39.76±4.90	52.32±9.90	56.98±7.90	73.48±4.90

The n and m decompositions in the table I represent the number of cameras and targets. As shown in the table I, Ours method achieved the highest VDM2C value in all experimental configurations, significantly outperforming all baseline methods. This fully demonstrates the effectiveness of the proposed hierarchical framework in processing the AMOT-3DR task. The limitation of the greedy algorithm is that when multiple targets are spatially dispersed, heading towards the geometric center of the target often causes most targets to move out of the camera's field of view, making it impossible to achieve effective coverage. Due to the complexity of the AMOT-3DR task, the global high-level reward signal is difficult to be directly learned by MADDPG and MAPPO and effectively guide the local action decisions of each agent, resulting in low collaboration efficiency.

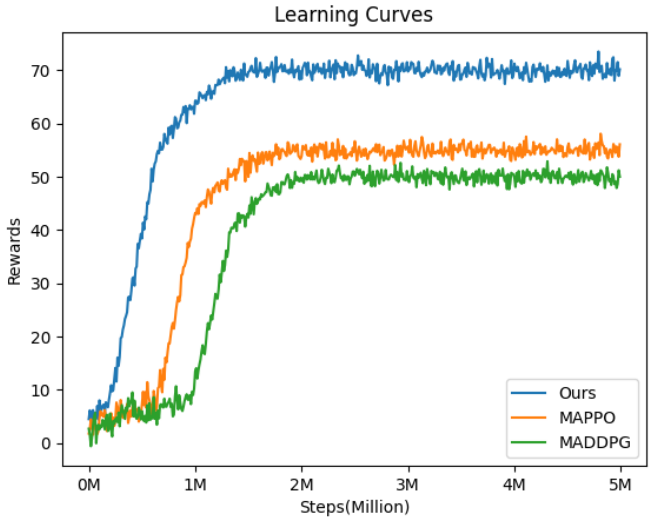

Fig. 5 The learning curve for each method.

As shown in Fig. 5, the learning curves under 6 cameras and 5 targets are shown. It is obvious that our method is superior to other methods in terms of convergence speed and final reward value. This further verifies the effectiveness of our method on the AMOT-3DR task.

V. CONCLUSIONS

This paper is dedicated to solving the problem of active multi-target tracking for 3D reconstruction. In order to effectively improve the coverage quality of the target and meet the minimum double coverage and multi-view geometry constraints required by 3D reconstruction, we propose an innovative framework based on hierarchical reinforcement learning. The core idea of the proposed method is to decompose the complex AMOT-3DR task into two hierarchical subtasks. The high-level coordinator focuses on global optimization by learning an effective target allocation strategy that maximizes the M2C and the triangulation angle for each target. This forms the foundation for precise and robust 3D reconstruction. The low-level executor is responsible for local execution by adjusting the camera pose in real time to efficiently track the assigned targets while keeping them centered in the field of view. Extensive experimental results demonstrate that the proposed framework significantly outperforms baseline multi-agent reinforcement learning methods in terms of both learning efficiency and target coverage quality. Overall, this study offers an effective solution for active multi-object tracking for 3D reconstruction.

REFERENCES

[1] M. Firouznia, J. A. Koupaei, K. Faez, G. A. Trunfio, and H. Amindavar, “Adaptive chaotic sampling particle filter to handle occlusion and fast motion in visual object tracking,” Digit. Signal Process., vol. 134, p. 103933, 2023.

[2] T. I. Amosa, P. Sebastian, L. I. Izhar, et al., “Multi-camera multi-object tracking: A review of current trends and future advances,” Neurocomputing, vol. 552, p. 126558, 2023.

[3] W. Luo, P. Sun, F. Zhong, W. Liu, T. Zhang, and Y. Wang, “End-to-end active object tracking via reinforcement learning,” in Proc. Int. Conf. Mach. Learn. (ICML), Jul. 2018, pp. 3286–3295.

[4] F. Zhong, P. Sun, W. Luo, et al., “AD-VAT: An asymmetric dueling mechanism for learning visual active tracking,” in Proc. Int. Conf. Learn. Representations (ICLR), 2019.

[5] J. Li, J. Xu, F. Zhong, X. Kong, Y. Qiao, and Y. Wang, “Pose-assisted multi-camera collaboration for active object tracking,” in Proc. AAAI Conf. Artif. Intell., vol. 34, no. 1, pp. 759–766, Apr. 2020.

[6] F. Zhong, X. Bi, Y. Zhang, W. Zhang, and Y. Wang, “RSPT: Reconstruct surroundings and predict trajectory for generalizable active object tracking,” in Proc. AAAI Conf. Artif. Intell., vol. 37, no. 3, pp. 3705–3714, Jun. 2023.

[7] H. Nguyen, B. Pham, H. Du, S. Thudumu, R. Vasa, and K. Mouzakis, “CSAOT: Cooperative multi-agent system for active object tracking,” arXiv preprint, arXiv:2501.13994, 2025.

[8] J. Xu, F. Zhong, and Y. Wang, "Learning multi-agent coordination for enhancing target coverage in directional sensor networks," Adv. Neural Inf. Process. Syst., vol. 33, pp. 10053–10064, 2020.

[9] Y. Wang, F. Zhong, J. Xu, and Y. Wang, "ToM2C: Target-oriented multi-agent communication and cooperation with theory of mind," in Proc. Int. Conf. Learn. Representations (ICLR), 2022.

[10] Z. Chen, J. Zhao, M. Yang, W. Zhou, and H. Li, "Optimizing camera motion with MCTS and target motion modeling in multi-target active object tracking," ACM Trans. Multimedia Comput. Commun. Appl., vol. 20, no. 7, pp. 208:1–208:19, Jul. 2024.

[11] Y. Yu, Z. Zhai, W. Li, and J. Ma, "Target-oriented multi-agent coordination with hierarchical reinforcement learning," Appl. Sci., vol. 14, no. 16, 2024.

[12] Z. Fang, J. Zhao, M. Yang, Z. Lu, W. Zhou, and H. Li, "Coordinate-aligned multi-camera collaboration for active multi-object tracking," Multimedia Syst., vol. 30, no. 4, p. 221, 2024.

[13] P. Murdin, Full Meridian of Glory: Perilous Adventures in the Competition to Measure the Earth. Copernicus Books/Springer, 2009.

[14] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," Artif. Intell., vol. 101, no. 1-2, pp. 99–134, 1998.

[15] R. Lowe, Y. I. Wu, A. Tamar, J. Harb, O. Pieter Abbeel, and I. Mordatch, "Multi-agent actor-critic for mixed cooperative-competitive environments," in Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 30, 2017.

[16] C. Yu, A. Velu, E. Vinitsky, J. Gao, Y. Wang, A. Bayen, and Y. Wu, "The surprising effectiveness of PPO in cooperative multi-agent games," in Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 35, pp. 24611–24624, 2022.