Cognitive UAV Tracking: Leveraging DRL and Hybrid Curriculum Learning for Target Reacquisition

Jiaqing Wang[1], Baichuan Zeng[3], Lan Deng[4], Ze Ji[5], Changyun Wei[4,*], Zheng Zeng[1,2,*]

***Abstract*—Tracking a moving unmanned ground vehicle (UGV) with an autonomous Unmanned Aerial Vehicle (UAV) is challenging, particularly in GNSS-denied indoor environments where reacquiring the UGV after losing track poses a significant obstacle. This paper presents a novel learning framework designed to address these challenges, enabling a quadrotor UAV to effectively chase a moving UGV and regain tracking in an indoor environment. The proposed framework encompasses two primary components: the Track-HCL and the Tracking Vision System (TVS). The TVS leverages a lightweight tracker to offer real-time recognition and localization of the UGV. Additionally, the Chronological Ghosting (CG) method is employed to describe the UGV's motion trend within a single frame. The Track-HCL component introduces a hybrid curriculum strategy to guide policy learning for the Deep Reinforcement Learning (DRL) agent. The Track-HCL enables the agent to learn the tracking policy conducive to target chasing and proficient reacquisition. We demonstrate the effectiveness of the proposed method in both simulation and field experiments.**

***Index Terms*—UAV, UAV tracking, Deep Reinforcement Learning, Curriculum learning**

I. Note to Practitioners

This research addresses the challenge of autonomously tracking a moving UGV with an UAV in environments where GPS signals may not be available, such as indoors. The solution is particularly valuable for applications requiring precise location tracking, such as logistics in large warehouses, surveillance, or search and rescue missions where real-time data and reactivity are crucial. Our approach combines a novel learning framework using DRL with a hybrid curriculum strategy, enhancing the UAV's ability to adapt and respond to dynamic changes in the UGV's path even when visual contact is temporarily lost. The TVS we developed is key to this adaptability, providing real-time recognition and localization of the UGV, which allows the UAV to maintain tracking or quickly reacquire the target after disruptions caused by obstacles or signal interference.

The practical implementation of our system shows that it can significantly improve the efficiency and reliability of UAV tracking operations. It reduces the need for manual intervention in tracking processes and increases operational uptime in environments where traditional tracking methods may fail. However, there are limitations, such as the dependency on visual cues, which might be obstructed in highly cluttered environments. Future enhancements could involve integrating multisensory data to mitigate this.

The potential applications of this technology extend beyond the tested scenarios. Industries that operate in complex and dynamic environments, like urban logistics or emergency response in disaster-stricken areas, could benefit from deploying UAVs equipped with our tracking system. This would not only improve operational efficiency but also enhance safety and response times.

II. INTRODUCTION

UNMANNED aerial vehicles (UAVs) are essential in numerous domains such as agriculture [1], logistics [2], reconnaissance [3], and search-and-rescue operations [4]. In agriculture and logistics, UAVs with enhanced tracking capabilities can autonomously follow and monitor large herds or equipment over vast areas. These sectors benefit significantly from UAVs' capabilities in complex tracking tasks, utilizing their aerial vantage point to offer unique advantages. UAVs equipped with sophisticated tracking systems can monitor and follow dynamic targets across diverse terrains, and their autonomous operation allows them to track continuously even in environments where prior geographic information is not available.

Autonomous tracking play a vital role in optimizing UAV efficacy. Techniques like PID (Proportional, Integral, Derivative) control have been widely employed to achieve consistent tracking of a moving target [5], [6]. Various researchers address trajectory optimization challenges by conceptualizing tracking as a problem of optimizing flight paths to ensure continuous target engagement [7]–[9]. Additionally, Gu et al. [10] utilize Deep Reinforcement Learning (DRL) to enhance UAVs' capabilities in autonomously following moving ground targets. These methods heavily depend on uninterrupted target perception. However, vision-based tracking scenarios often experience interruptions, especially when obsta-

[1]School of Oceanography, Shanghai Jiao Tong University, Shanghai, 200240, China

[2]State Key Laboratory of Submarine Geoscience, Shanghai Jiao Tong University, China

[3]Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong, China.

[4]College of Mechanical and Electrical Engineering, Hohai University, Changzhou, 213022, China.

[5]School of Engineering, Cardiff University, Cardiff CF24 3AA, United Kingdom.

This Research is supported by the Oceanic Interdisciplinary Program of Shanghai Jiao Tong University (project number SL2022ZD106 and SL2023ZD206) .

*Corresponding authors: Changyun Wei (e-mail: c.wei@hhu.edu.cn) and Zheng Zeng (e-mail: zheng.zeng@sjtu.edu.cn)

cles or terrain changes cause the target to move outside the UAV's camera field of view (FOV) [11]. Techniques involving gimbal-controlled cameras have been proposed to mitigate these challenges by dynamically adjusting the FOV [12], yet the problem of obstacle-induced tracking interruptions persists.

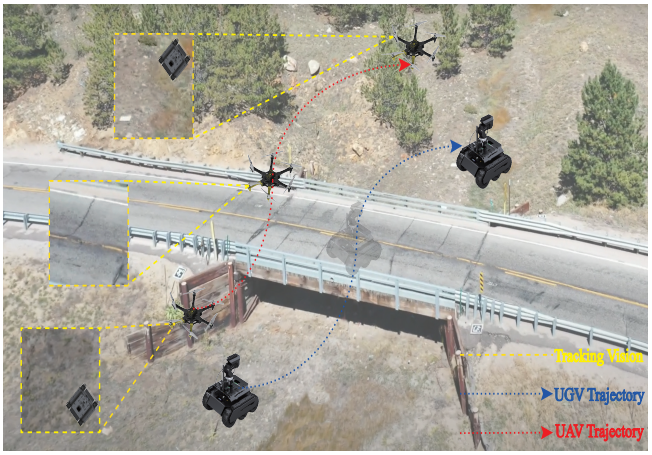


Fig. 1. Demonstration of the target reacquisition mission with the visualization of the generated trajectory.

To address the aforementioned issues, we implement a Hybrid Curriculum Learning strategy to guide DRL agent in learning tracking strategies. Additionally, we introduce the Tracking Vision System (TVS) to provide our DRL-based controller with comprehensive state information regarding the UGV. In instances where the UAV loses track, the TVS is employed to reestablish tracking of the UGV. Based on the feedback obtained by the TVS, the UAV can chase the UGV when the UGV is observable, and can perform target reacquisition when the UAV loses track. A series of experiments are conducted in which the UAV endeavors to track a moving UGV. Fig. 1 depicts the air-ground target reacquisition mission. The main contributions are summarized as follows:

- We introduce an innovative approach utilizing hybrid curriculum learning to efficiently guide the learning trajectory for UAV cognitive tracking.
- We present a streamlined pipeline designed to achieve rapid visual tracking while simultaneously providing accurate pose estimation. The pipeline also incorporates a retracking mechanism to recover tracking in case of losess.
- The integrated tracking framework enables UAV to chase the possible trajectory of the UGV even when it loses track. This advancement contributes to the robustness and reliability of the tracking process.

III. RELATED WORK

In the past decade, there has been a rapid surge in advancements in UAV autonomous tracking. Vision-based methods for UAV autonomous tracking control have gained considerable attention, particularly due to the progress made in lightweight visual tracking algorithms [13]–[15]. Previous works such as those by Redmon et al. [16] have proposed vision-based methods utilizing traditional control techniques to minimize tracking errors at each step. Li et al. [17] uses a Vision Transformer-based technique for real-time UAV vision tracking. Lin et al. [18] have taken obstacle constraints into consideration during tracking, formulating tracking problems as trajectory optimizations within a pre-built flight corridor. However, these approaches often require prior knowledge of the environment or the tracking target, limiting their deployability.

Zhao et al. [19] employed an end-to-end DRL control method for quadrotor tracking. By taking perceptual and environmental constraints into account, this intelligent approach may usher in a new era of UAV autonomous tracking. The effectiveness of Average-Reward TD Learning has been demonstrated in the work of Zhang et al. [20]. However, TD Learning's experiment results are primarily derived for tabular or linear function approximation, leaving open questions about extending to more complex function approximators or high-dimensional problems. Huang et al. [21] have proposed enhancements in Soft Actor-Critic methodologies. Sadly, its' SAC-based methods typically require extensive tuning and large batch sizes for best performance. Xia et al. [22] propose a parallel optimal learning control strategy to achieve UAV cooperative tracking control. But the further adaptations may be necessary before applying the framework to a broader class of multi-agent systems. Recent work uses a monocular-camera-only approach for dynamic target chasing by drones in dense environments [23]. It employs a cross-modal variational autoencoder to encode RGB images into latent vectors to include the target information. Gupta et al [24] propose a novel drag-aware model combined with Model Predictive Control to enabling UAVs to autonomously track high-speed USVs. However, this method is sensitive to visibility limitations due to reliance on visual pose estimation alone.

IV. PROBLEM FORMULATION

A. Tracking Process with Limited Perception

This paper aims to enable the UAV to maintain tracking of the UGV even when visual perception is lost. The UAV is required to re-establish tracking once the UGV reappears, as illustrated in Fig. 2.

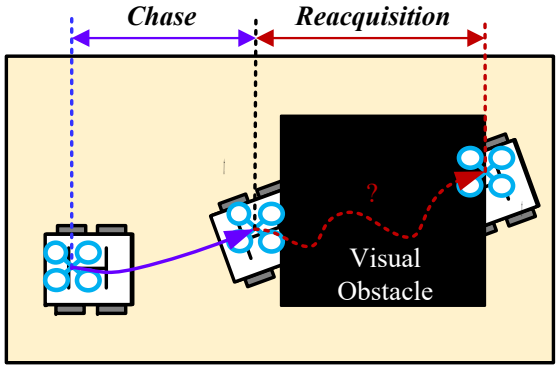


Fig. 2. Illustration of the UAV target tracking process.

Firstly, target reacquisition requires the ability to re-identify the target, demanding a high-level understanding of modified UGV images (e.g., rotation or segmentation). Secondly, it is difficult to pilot the UAV to track a non-linearly moving UGV under visual obstacles (VOs). In manually-controlled cases, human pilots often predict the movement trajectories by observing past motion trends of the UGV when it loses track [25]. This approach entails cognitive interpretation to anticipate the UGV's motion pattern. Therefore, designing an autonomous UAV tracking system with limited perceptual input presents a challenge in creating an effective cognitive controller.

B. Curriculum Learning for DRL Training

A key challenge in DRL is selecting tasks whose difficulty is well-matched to the agent's current skill level. If tasks are too easy or too difficult, training can stagnate or fail to produce meaningful improvements. To tackle this, Curriculum Learning (CL) methods propose sequencing tasks in a manner that accelerates DRL training, often emphasizing tasks of intermediate difficulty as more informative for policy improvement [26].

However, existing Adaptive Curriculum Learning (ACL) approaches often fall short due to reliance on a Difficulty Discriminator (DD) that is prone to misclassification [27], [28]. When the DD is not well-trained—especially at early training stages—it can prematurely assign overly complex tasks to the agent, destabilizing or slowing learning [29]. Moreover, many ACL methods pre-train the DD using manually constructed task databases rather than in an end-to-end fashion, undermining automation and adaptivity [30].

V. METHODOLOGY

A. System Overview

We initiate tracking using a continuous stream of 720×720 RGB images at a rate of 20 frames per second. Utilizing the mature capabilities of state-of-the-art trackers specifically designed for UAV visual tracking, a monocular camera snapshot is adequate for identifying and locating the designated UGV. We use the Tracking Vision System (TVS) to offer the DRL agent with a sequence of relative orientation maps between the UAV and the UGV at each time step. The RL agent is then tasked to continuously track the UGV through the velocity commands from the DRL controller.

Moreover, our Chronological Ghosting (CG) method provides the controller with information regarding the motion trends of the UGV during instances when the UAV loses track. In the absence of UGV perception information, we rely on our cognitive controller to pursue potential UGV trajectories. After the UGV reappears, we use the TVS to relocate and retrack the UGV visually, providing the controller with the renewed information to perform target reacquisition.

B. Track-HCL

1) Task Setup: Our task encompasses four environmental variables, as presented in Tab. I. We categorize our task into two main categories: *Chase* tasks, and the *Reacquisition* tasks.

TABLE I
TASK ITEM OF TRACK-HCL

Tasks	Parameters
Maximum linear velocity of UGV	v_{UGV}
Maximum angular velocity of UGV	ω_{UGV}
Number of visual obstacles	N
Maximum wind speed	v_{wind}

- *Chase*: The *Chase* tasks are introduced to provide the agent with adequate signals to learn low-level principles for tracking the UGV. Specifically, the *Chase* tasks encompass only two variables: the UGV's linear velocity v_{UGV} and angular velocity ω_{UGV}. The UGV's velocity increases progressively during training.
- *Reacquisition*: The *Reacquisition* tasks are designed to train the agent in tracking the UGV when the perception of the target is temporarily blocked . Additionally, the *Reacquisition* tasks necessitate the agent to reestablish tracking after the UGV becomes visually detectable again. The *Reacquisition* tasks incorporate two supplementary variables based on the *Chase* tasks: the number of visual obstacles denoted by N and the maximum wind speed v_{wind}. This Gaussian noise v_{wind} is restricted to a magnitude of 0.3 m/s and persists for a maximum duration of 2 seconds.

2) Hybrid Curriculum Learning: This paper introduces the Track-Hybrid Curriculum Learning (Track-HCL) approach, where the agent is presented with tasks of carefully-selected difficulty gradients. The Track-HCL comprises two main components: Progressive Curriculum Learning (PCL) and Autonomous Curriculum Learning (ACL), as illustrated in Fig. 3.

- Progressive Curriculum Learning (PCL): Initially, the PCL is employed to regulate the advancement of learning in the task of *Chase*, achieved by incrementally introducing more complex tasks based on the agent's ongoing performance. Therefore, we introduce a cumulative-reward-based variable denoted as μ_n to govern the learning progression in our PCL. The updating of μ_n is as follows:

$$\mu_{k+1} = \omega_\mu \sum_{t=0}^{\infty} \gamma^k R_t + \mu_k, \tag{1}$$

  where R_t signifies the reward received by the agent at each time step t. Subsequently, the PCL can adapt the output *Chase* tasks in trial $k+1$ based on the received accumlated reward in trial k. The coefficient ω_μ is employed to establish a mapping between the training performances of the DRL agent and the progression of our curriculum.
- Autonomous Curriculum Learning (ACL): Inspired by the work of Morad et.al. [28], a Difficulty Discriminator (DD) is employed to generate task success probability as a metric to gauge the complexity of the output *Reacquisition*

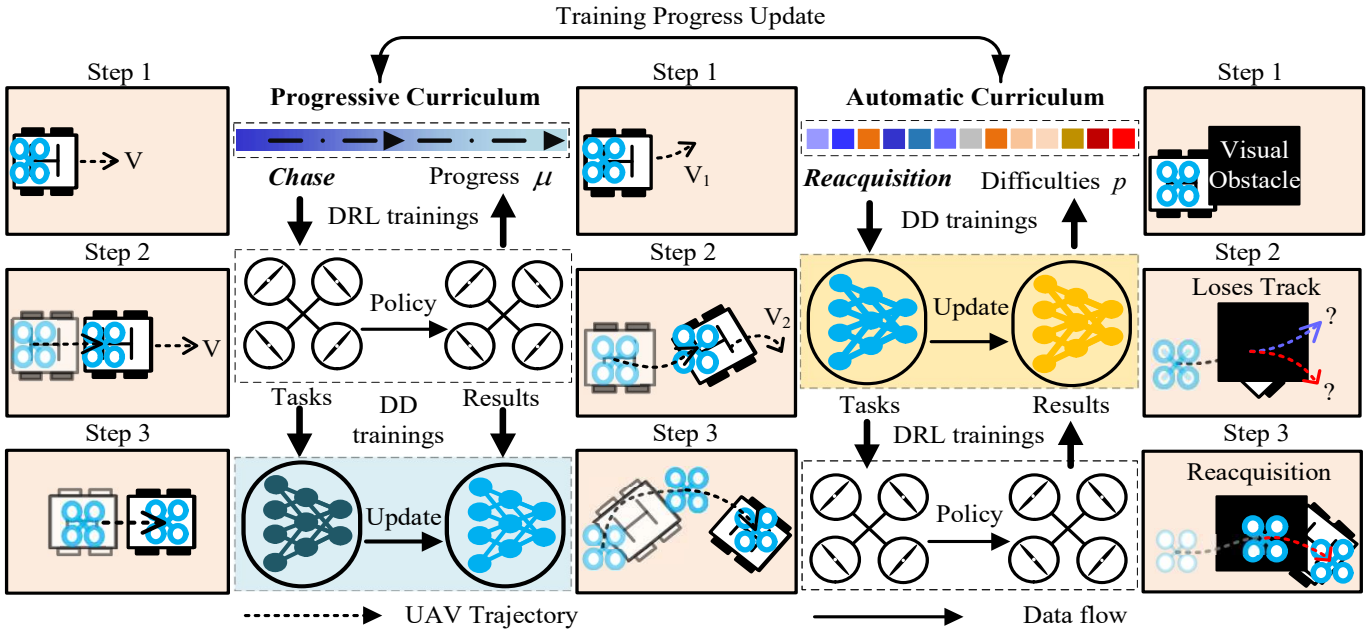


Fig. 3. The Track-HCL pipeline combines the benefits of both PCL and ACL. The training of the DD in Track-HCL starts with end-to-end training, involving learning from scratch, using the data collected from PCL.

task in the ACL. Our DD training employs the end-to-end training method. Deliberately configuring a replay buffer with limited capacity, our curriculum learning update facilitates Track-HCL's concentration on guiding recent task learning. The output value of the task elements is subsequently modified based on the success probability, ensuring a persistent challenge for the agent with tasks of balanced complexity.

Algorithm 1 illustrates our Track-HCL framework, where U_k represents the output task in trial k. Δ_{vary} denotes the value variation for each task element between the tasks of *Chase* and *Reacquisition*.

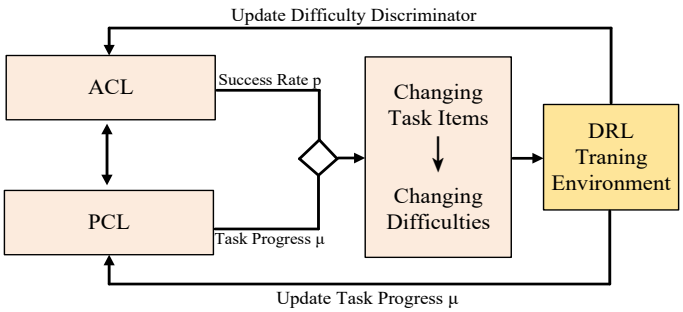


Fig. 4. Track-HCL implementation flowchart. PCL adjusts UGV speed based on cumulative-reward progress μ; ACL uses a Difficulty Discriminator to tune obstacle count; both feed into the DRL training loop until convergence.

Fig. 4 presents a concise flowchart of the Track-HCL framework, illustrating how PCL and ACL components interact with the DRL training loop.

The UAV starts a tracking mission, encountering either a Chase or Reacquisition scenario. The DRL system updates

Algorithm 1 Track-HCL

Input: Progress μ_k;
Output: Task U_k;

for $t = 1$ to T **do**:
  $TaskType \leftarrow GetTaskType(\mu_k)$;
  if $TaskType = Chase$ **then**:
    Get U_k from $U_k = \mu_k \Delta_{vary} + U_0$;
    return U_k.
  end if;
  if $TaskType = Reacquisition$ **then**:
    $p \leftarrow GetSuccessProbability(TaskType)$;
    if $p < \eta$ **then**:
      return $U_k = U_{Reacquisition} - p\Delta_{vary}$;
    end if;
    if $\eta < p < \xi$ **then**:
      return U_k;
    end if;
    if $\xi < p$ **then**:
      return $U_k = U_{Chase} + p\Delta_{vary}$;
    end if;
  end if;
  Update progress μ_K using Eq. 1;
end for.

the UAV's tracking policy based on real-time data, adjusting its approach to effectively chase or reacquire the target. The PCL module receives updates on the UAV's performance and task complexity. Simultaneously, the ACL module adapts the difficulty of tasks (represented as difficulties p in the figure) using a DD. The DD assesses the probability of task success and adjusts future tasks to maintain a balanced challenge for

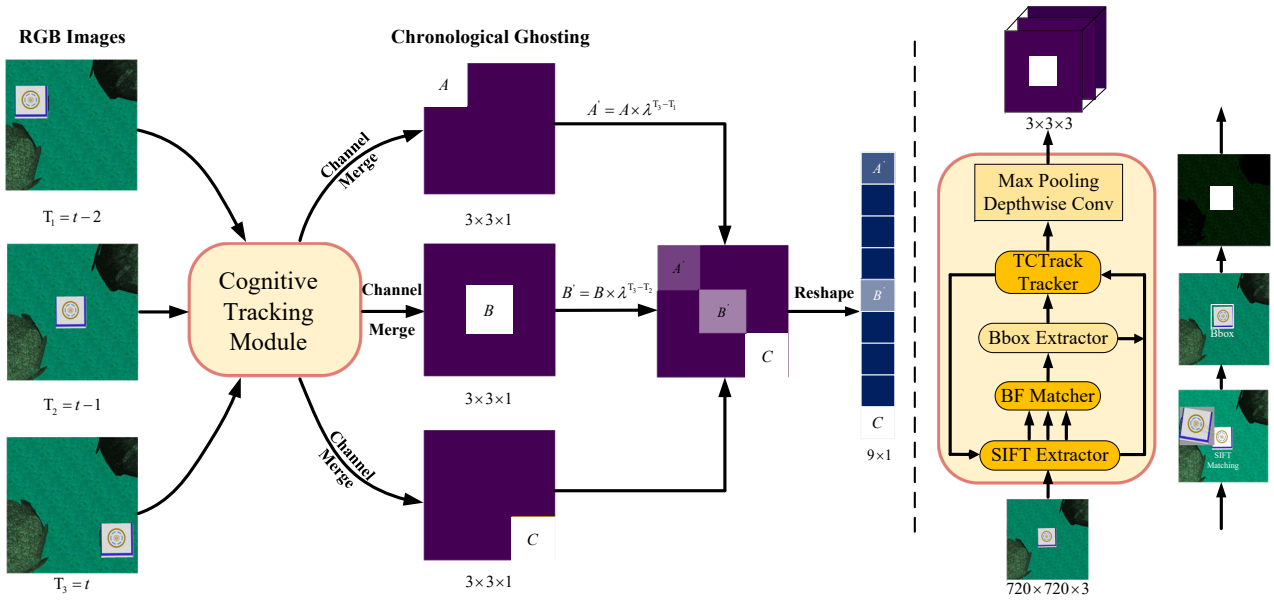


Fig. 5. Pipeline for the Tracking Vision System. By employing the TCtrack tracker to discern the designated UGV, our Tracking Vision System (TVS) exhibits enhanced resilience to environmental disturbances. Additionally, we utilize a temporally concatenated, gray-scaled image consisting of only 3 × 3 pixels as the primary component of the state input. This approach reduces training expenses and accelerates policy convergence.

the UAV. The outcomes of tasks, whether successful or not, along with the adjusted task difficulties, are fed back into the DRL system. Data from the DRL's performance on these tasks is also used to train the DD, ensuring that the task difficulty is always optimally challenging for the UAV's current skill level.

C. Tracking Vision System

Rapid visual tracking is implemented through the Tracking Vision System (TVS), which utilizes lightweight and efficient algorithms designed for real-time application. The TVS aims to provide the UAV with the capability to recognize and locate the target UGV, as shown in Fig. 5. The TVS leverages algorithms such as Scale-Invariant Feature Transform (SIFT) and Temporal Contexts for Aerial Tracking (TCTrack), enabling it to quickly identify and track UGV even in dynamically changing environments. Specifically, we employ the Cognitive Tracking Module (CTM) to effectively track, highlight, and subsequently retrack the intended ground-based target. This component is primarily responsible for the initial capture and continuous monitoring of the UGV, ensuring that the UAV maintains a visual lock on the target throughout the operation. We then normalize the output image and use the Chronological Ghosting (CG) to provide the agent with the motion trend of the UGV.The CG technique plays a pivotal role in enhancing the pose estimation capabilities of the system. By integrating temporal data from sequential frames, CG constructs a richer context of the UGV's movement, providing a composite image that reflects both the current and preceding locations. Finally, the image is reshaped into a 9 × 1 array, constituting the state vector at time step t.

1) Coginitive Tracking Module: We have developed the Cognitive Tracking Module (CTM) through the utilization of the SIFT (Scale-Invariant Feature Transform) algorithm [31] and the TCTrack tracker [32], facilitating target recognition and tracking. Our visual pipeline commences with the application of SIFT to extract feature points from the provided template. Subsequently, the identified keypoint matches are employed to deduce the homography matrix. This matrix is then employed to transform the corners of the template, thereby delineating a Bounding-box (Bbox) that encompasses the designated UGV. Moreover, we use the derived Bbox to provide the TCTrack tracker with the initial frame. Once the tracker is initiated, we constantly highlight the TCTrack-generated Bbox in the image while darken the rest of the image. We then use the Depthwise Separable Convolution (DSC) method [33] and the Max Pooling techniques to condense the image information into an image of 3 × 3 pixels. Finally, we normalize the pixel value in the image. When the visual tracking of the UGV fails, the CTM takes charge by reinstating SIFT for target localization and reidentification. This dynamic process ensures the reacquisition of visual tracking.

2) Chronological Ghosting: Ghosting, a term commonly employed in photography, refers to the creation of multiple exposures of an object within a single frame. However, ghosted images are occasionally utilized to consolidate correlated motion information from multiple frames into a singular frame. The provision of the target's motion trend to the agent is of utmost importance for effective cognitive tracking acquisition. In light of this, we introduce the Chronological Ghosting (CG) methodology, which extracts the UGV's motion trend through the analysis of a sequence of consecutive frames. First, the location of the UGV is extracted by the SIFT algorithm and a Brute Force Matcher, we use previously prepared feature

points extracted from the template of the marker that guides the tracking process. We then highlight the area that contains the UGV and use DSC to divide the approximate position of the UGV into the corresponding area of the final 3×3 image The process commences by assembling CTM-processed images spanning from past time step t_n to the current time step t. Subsequently, we manipulate the pixel value V_{image} within the collected images as outlined below:

$$V'_{image} = V_{image} \cdot \lambda^{t-t_n}, \tag{2}$$

where a discounting variable λ ($\lambda \in (0, 1)$) is introduced to concatenate continuous frames into a single frame. As the frame's temporal proximity to the current time step t increases, so does its capacity to preserve higher pixel values. Consequently, the composite result of aggregating these processed images elegantly encapsulates the evolving motion trends of the UGV. This not only provides the UAV with a comprehensive understanding of the UGV's motion state, but also leverages this knowledge to guide subsequent decisions and actions.

The pose estimation within our system is provided by the CG method, which is integrated into the CTM. When the UAV loses visual tracking of the UGV, the CG method steps in by analyzing consecutive frames to provide a motion trend of the UGV. This is achieved by ghosting images that consolidate correlated motion information into a singular frame, providing the UAV with a comprehensive view of the UGV's likely trajectory and movement patterns.

D. TD3 Controller

1) Deep RL Architecture: We use the Twin Delayed Deep Deterministic Policy Gradient (TD3) algorithm [34] for our policy training. The state vector used in our approach consists of images generated by the Tracking Vision System (TVS). At time step t, the state vector s_t is defined as $s_t = (\mathbb{R}^{3\times3\times1}, D_t)$. The pixel values within $\mathbb{R}^{3\times3\times1}$ are normalized to the interval $[0, 1]$. Additionally, a warning variable D_t is introduced to indicate whether the UAV has lost track of the UGV at time step t. D_t is set to 1 if the UAV has the destinated UGV in sight, and is set to 0 when the TVS dose not find the ground target. In comparison to the direct utilization of 120×120 RGB images as the state representation in the work of Zhao et.al. [19], the use of 3×3 maps offers increased generality and has been shown to lead to expedited training and enhanced policy performance [35], [36]. The action space A is defined as a two-dimensional continuous space. Specifically, an action a is denoted as $a = (\mathrm{v_x}, \mathrm{v_y}) \in A$, where $\mathrm{v_x}$ and $\mathrm{v_y}$ represents the reference linear velocities along the x and y axes in the world coordinate system, respectively.

2) Reward function R: It is essential to establish a well-defined reward function R to formulate an effective tracking policy P. The realization of R comprises three components: chase rewards r_{chase}, cognitive tracking reward $r_{cognite}$, and the termination reward r_T. At each time step t, the total reward is composed of different terms:

$$r_t = \begin{cases} \omega_1 \cdot r_{chase} + \omega_3 \cdot r_T & \text{if } \textit{Chase} \\ \omega_1 \cdot r_{chase} + \omega_2 \cdot r_{reacquire} + \omega_3 \cdot r_T & \text{if } \textit{Reacquisition} \end{cases} \tag{3}$$

where $\omega_{i(i=1-3)}$ are the weight coefficients assigned to each reward component. A step reward r_{chase} is assigned at constant intervals until a training episode ends. We encourage the agent to minimize the relative distance at each step through using the r_{chase} as follows:

$$r_{chase} = (d_{in} - d_t) \cdot n_{approach}, \tag{4}$$

where $d_t = \sqrt{\Delta d_{x_t}^2 + \Delta d_{y_t}^2}$, and d_{in} represents the maximum distance for successful tracking. $n_{approach}$ increments by 1 (and has a maximum value of 5) when d_t is within the range of d_{in} and resets to 1 when the limit is exceeded.

While the objective of r_{chase} is to provoke chasing when the agent spots and identify the UGV using TVS, the cognitive tracking reward $r_{reacquire}$ deals with the problem of tracking when the UAV loses track of the UGV. Therefore, we reward the agent when it succeeds in matching the possible trajectories of the UGV. We collect the ground truth of the UGV below the VO and return the Euclidean distance between them. The $r_{reacquire}$ is calculated as:

$$r_{reacquire} = 5(d_{in} - d_t) + 5(\theta_{in} - \theta_t), \tag{5}$$

θ_t denotes the relative angle between a_t and the velocity v_{UGV} of the UGV at time step t. Moreover, we incorporate a threshold θ_{in} to help the agent to require tracking once the target reappears, which assists the agent in selecting actions with greater alignment towards tracking. Additionally, the $r_{reacquire}$ is design to help the agent to cross the learning gap between *Chase* and *Reacquisition* by returning a large reward when the agent manage to catch up with the ground target once it reappears. The terminal reward r_T is assigned at the end of each episode. An episode terminates when the UAV either exceeds the allowable distance for continued tracking or reaches the maximum number of steps allowed for tracking. We assign a positive terminal reward to the agent if it completes the tracking task within the maximum steps T, or a penalty is imposed if the UAV has completely lost track of the UGV. The terminal reward r_T is represented as follows:

$$r_T = \begin{cases} 1, & \text{success} \\ -1, & \text{else} \end{cases} \tag{6}$$

3) Short Term Memory: Due to variations in camera pose, flight vibrations, or obstructions from visual obstacles (VOs), the perception of the tracking target frequently experiences disruptions during the tracking process. In scenarios where manual control is employed, human pilots often predict the movement trajectories of ground targets by recalling the last observed movements of the Unmanned Ground Vehicle (UGV) after visual contact is lost. Inspired by this human capability, we propose the Short Term Memory (STM) module, designed

to endow the DRL controller with a self-updating memory function to handle losses in perception. By emulating the short-term memory strategy used by pilots, the STM aims to enable the DRL agent to make well-informed tracking decisions. This is achieved by leveraging both the latest and recently observed environmental information.

The STM module initiates its process by capturing visual tracking signals and storing the current state s_{t-n-1} from the Local Vision System (LVS), as depicted in Figure 6. If visual tracking persists into the next time step $t-n$, the STM uses s_{t-n} to refresh the previously stored state data. In cases where $D_t = 0$ (indicating a loss of UGV tracking), the STM replaces the current image $Image_n$ with the previously stored image $Image_{t-n}$, and this image is then fed to the TD3 controller. Should the Unmanned Aerial Vehicle (UAV) re-establish perceptual tracking of the ground target after n time steps, the STM updates its memory to incorporate the new image information $Image_t$, thereby preparing for any potential future disruptions in tracking.

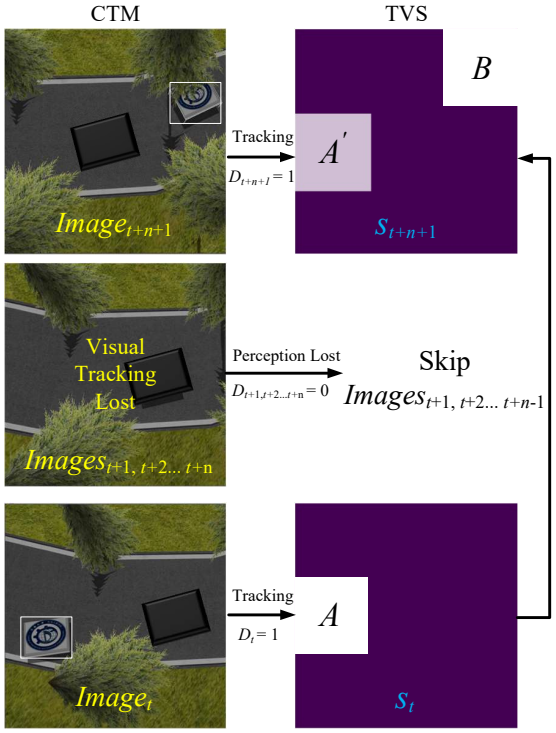


Fig. 6. Pipeline for the Short Term Memory module.

This method depends crucially on the STM module to maintain consistent tracking. The STM leverages data from prior successful tracking episodes to predict the potential movement of the UGV. Upon the re-appearance of the UGV within the field of view, the Tracking Vision System (TVS) is reactivated to refresh the tracking data, ensuring rapid and precise reacquisition of the target. The combined use of trend analysis through ghosting and memory-based prediction ensures robust and continuous tracking, even when direct visual confirmation is momentarily lost.

VI. EXPERIMENT

A. Experimental Settings

We conducted our experiment under both simulations and real-world scenarios. The simulated training and testing procedures for all the controllers were carried out within the Gazebo simulation environment, as depicted in Fig. 7.

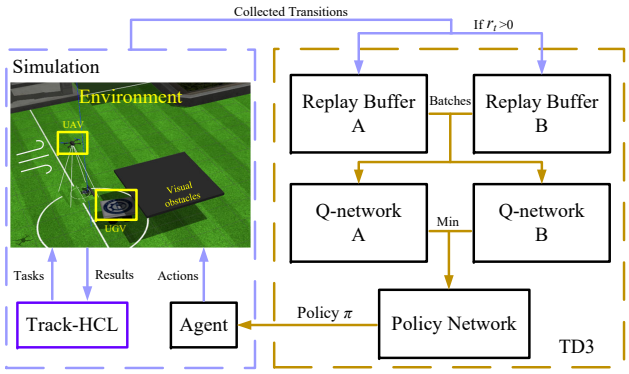


Fig. 7. An overall of our proposed training framework.

The PX4 Autopilot flight controller was linked using MAVLink [1] to the onboard computer. During our simulated experiments, we utilized a modified Iris UAV equipped with a downward-looking camera, perpendicular to the UAV frame. Importantly, the connection between the UAV and the on-board camera remained fixed for these experiments. Additionally, adaptations were made to the upper plate of a Husky UGV [2] model to install a detection marker.

B. Policy Training

During training and testing on a PC, our TVS runs at 107 FPS. For the training process, we conducted 10 trials, each consisting of 500 episodes. The entire training process took a total of 12 hours. This translates to an average time per trial of 1.2 hours. During each episode, both the UAV and the UGV are repositioned to their initial locations, with the UAV assigned the task of tracking the UGV. Additionally, the heading of the UGV is randomized at the outset of every episode. We set the VO to appear in front of the moving UGV suddenly in *Reacquisition* tasks, introducing diverse scenarios and preventing overfitting. Notably, for the baseline training using the TD3 algorithm and the Soft Actor-Critic (SAC) algorithm [37], the agent is trained directly with *Frontier* tasks. Throughout the training process, we utilize a mini-batch size denoted as N_{batch} and conduct N_{update} iterations at each time step. Additionally, we employ the Partitioned Buffer Replay method [38] to facilitate our DRL training. In the training sessions for Track-HCL and the ACL, we perform updates to the DD using a mini-batch size of N_{ACL} for N_{update} iterations subsequent to the conclusion of a training episode. The neural network architecture for our DD is illustrated in

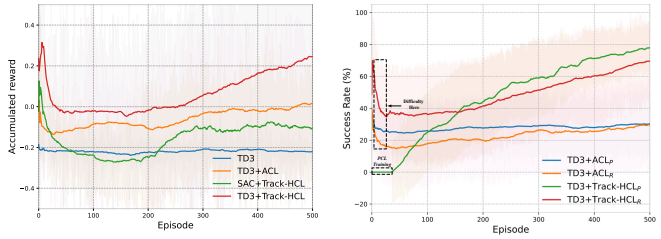
[1]https://px4.io/
[2]https://robots.ros.org/husky/

Tab. II. All hidden layers in the neural networks used the Leaky Relu [39]. The learning curves of our proposed models

TABLE II
DD Network Achitecture

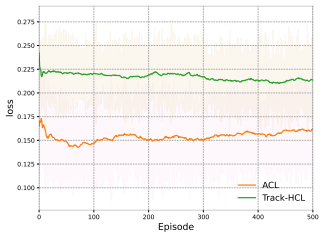
Input	Network Dimensions	Output	Operator
Tasks	4×20	FC1	leaky relu
FC1	20×20	FC2	leaky relu
FC2	20×1	P	sigmoid

TD3+Track-HCL, SAC+Track-HCL, the variant using ACL along with the TD3 baseline, are presented in Fig. 8.Each algorithm was used for 10 trials, with 500 episodes per trial. The rapid surge in the accumulated reward and success rate

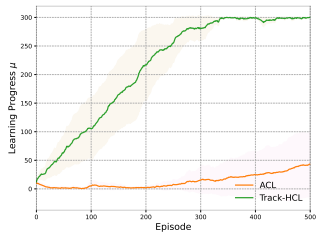


(a) Average cumulative reward over 10 trials with 500 episodes each; the UGV's max speed is raised from 0.4 m/s to 0.8 m/s and a second visual obstacle is introduced.

(b) Corresponding success rate; the transient dip at episode 30 reflects the difficulty jump, after which performance recovers in the TD3+Track-HCL.



(c) Mean loss of the DD in TD3 across 10 runs, showing stable convergence by 10 episodes.



(d) Average curriculum progress μ over training, illustrating gradual task difficulty increase and convergence under ACL and Track-HCL.

Fig. 8. Comparison of averaged accumulated reward and averaged success rate. We observe a significant increase in both the accumulated reward and the success rate of TD3+Track-HCL, indicating the effectiveness of our Track-HCL in learning cognitive UAV tracking principles.

of TD3+Track-HCL during the initial 30 rounds serves as a testament to the RL agent swiftly grasping the mechanics of the *Chase* task. This validates the efficacy of employing PCL in mastering basic tracking principles (see Fig.8(b)). Notably, the performance of the baseline TD3 and the SAC+Track-HCL algorithm remained consistently subpar, demonstrating the challenges in training a DRL agent for the *Reacquisition* task. Comparing to the performance of the TD3 and TD3+ACL in both Fig. 8(a) and Fig. 8(b), the learning curve of the Track-HCL rises sharply, demonstrating the effectiveness of the Track-HCL for guiding the agent in policy learning. The DDs of Track-HCL and ACL need to adapt to the learning outcomes of the agent at the end of each episode, thus leading to less descent observed in the loss curve (see Fig. 8). Additionally, the performance drop observed after the 30th episode in Fig. 8(b) results from intentional rise in task difficulties and environment complexity. Initially,the Track-HCL instructs the UGV to move slowly, allowing the model to collect effective samples and achieve a higher success rate. The curriculum increases the UGV's maximum speed from 0.4 m/s to 0.8 m/s and adds an extra visual obstacle. This difficulty jump induces a transient performance drop, which then recovers as the agent adapts under our Track-HCL scheme.

A conspicuous decline for TD3+Track-HCL and the TD3+ACL was observed after the 30th episode in both Fig. 8(a) and Fig. 8(b). This phenomenon offers two potential explanations: either the agent successfully learned *Chase* and subsequently encountered increased difficulty with the ACL-generated *Reacquisition*, or the agent never fully mastered the *Chase* task and got stuck in learning it. The steep incline depicted in Fig. 8(d) support the former explanation for Track-HCL. Nonetheless, since the ACL shows little increase throughout the training trails (see Fig. 8(d)), its decrease after the 30th episode aligns more closely with the latter hypothesis. Moreover, our Track-HCL approached the predefined limits of μ, confirming that TD3+Track-HCL effectively accomplished both *Chase* and *Reacquisition* tasks, thereby demonstrating the effectiveness of our hybrid curriculum strategy. The comparison between the learning progress of ACL and Track-HCL in Fig. 8(d) proves that employing a progressive curriculum is particularly efficacious in steering the early-stage training of basic tracking principles.

C. Simulated Tests

To test the effectiveness of our policies under various test scenarios, we performed the experiments using a UGV speed of 0.8 m/s. Since the rest of the trained policies are beyond optimal, indicating they are unable to fulfill the basic tracking tasks (see Sec. VI-B), we only implemented the tracking policy trained under TD3+Track-HCL. We repeated the tests under different scenarios for 100 trials. A test run is successful if the UAV keeps chasing the UGV within a distance limit of 3 m relative to the UGV.

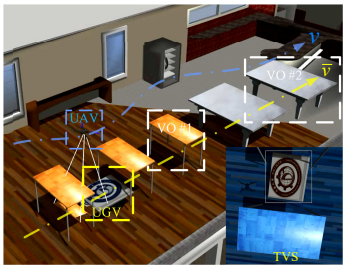
1) Test Scenario A: UGV movement along a straight trajectory: We set the UGV to move along a straight trajectory through several small-sized VOs with varying shapes before reaching the destination, as shown in Fig. 9(a). In Scenario A, we were more focused on testing the capability of TD3+Track-HCL in fulfilling *Chase* rather than fulfilling *Reacquisition*. Therefore, the VOs were set to have a relatively big space interval to enable the TVS to keep retracking and updating the position of the UGV for the controller.

Fig. 9(b) illustrates the averaged 3D trajectory of the UAV and the UGV. We observe that the UAV can chase the trajectory of the target UGV while planning smooth trajectories. It is noteworthy that when the UAV regains tracking of the UGV in the end, its trajectory quickly adjusts back to the UGV. Our method reached a 99% success rate in 100 testing trials, indicating the trained policy reached satisfying

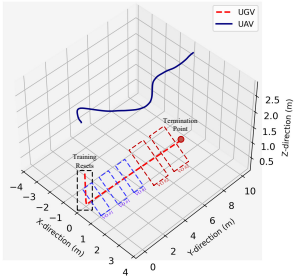
TABLE III
STATISTICS FOR THE SIMULATED TESTS

Scenario	Average distance error(m)				Average velocity error(m/s)				Success Rate
	x-axis	y-axis	σ	Total	x-axis	y-axis	σ	Total	
Scenario A	0.67 ± 0.02	0.41 ± 0.01	0.02	0.79	1.00 ± 0.02	0.50 ± 0.01	0.02	1.12	99 %
Scenario B	0.66 ± 0.02	0.45 ± 0.01	0.02	0.80	1.01 ± 0.02	0.49 ± 0.01	0.02	1.12	84 %
Scenario C	0.50 ± 0.02	0.82 ± 0.02	0.02	0.96	0.97 ± 0.03	1.01 ± 0.03	0.04	1.40	63 %

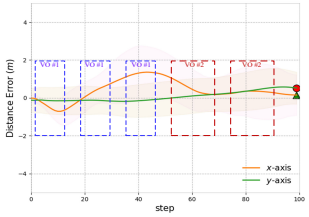
stability (see Tab. III). Both Fig. 9(c) and Fig. 9(d) demonstrate TD3+Track-HCL's superior performance in dealing with task *Chase* and constant track-retrack problems, thereby improving motion stability and safety throughout the tracking process.



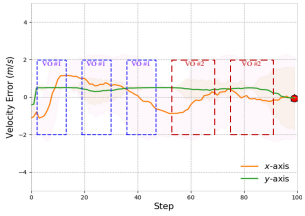
(a) Testing diagram of Scenario A in the Gazebo simulator.



(b) Averaged trajectories of tracking a moving UGV.



(c) Averaged distance error.

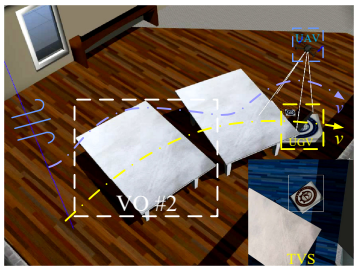


(d) Averaged velocity error.

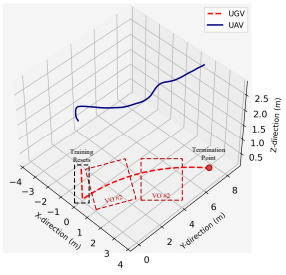
Fig. 9. Illustration of the averaged trajectory, distance, and velocity error for our TD3+Track-HCL in Test Scenario A.

Conversely, the UAV's most common failure was its inability to locate the UGV. Therefore, we conducted more testing under Scenario B to test the capability boundary of our method to fulfill task *Reacquisition*.

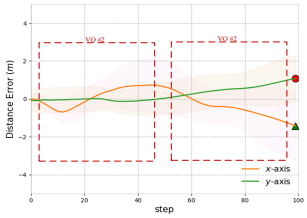
2) Test Scenario B: UGV movement along a curved trajectory: To further validate our method's cognitive ability to fulfill *Reacquisition*, we set the UGV to move along a curved trajectory through two VOs held together. The setup for our Test Scenario B is depicted in Fig. 10(a). The 3D trajectory of the UAV and the UGV is illustrated in Fig. 10(b). The UAV showed less elegant diving behavior when losing track of the UGV, which indicates the UAV has learned to search for the UGV when losing track. The UAV's movements are more gentle in Fig. 10(b) compared to Fig. 9(b), and the averaged distance error and averaged velocity error are both smaller than in Test Scenario A. This improved performance can be attributed to the UAV having learned to manoeuvre cautiously during long-periodical periods of losing track. However, the distance error in the x-axis is relatively bigger at the end of the testing trials. There are two possible explanations for this



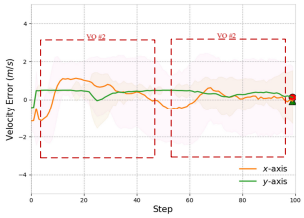
(a) Testing diagram of Scenario B in the Gazebo simulator.



(b) Averaged trajectories of tracking a moving UGV.



(c) Averaged distance error.

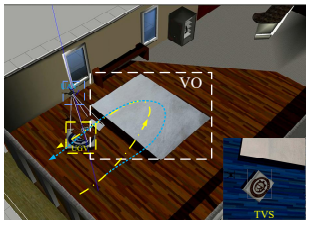


(d) Averaged velocity error.

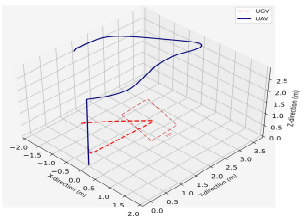
Fig. 10. Illustration of the averaged trajectory, distance and velocity error for our TD3+Track-HCL in Test Scenario B.

unsatisfying result. First, it is possible that using a 3×3 image is not sufficient to express the small turning trend of the UGV, which could be misinterpreted by the UAV that the designated UGV was moving in a straight line. Second, it is possible that the trained policy lacks the sophistication to predict and track a curved-moving UGV while the UAV has visually lost track of it, which might be improved by modifying the reward design. Nevertheless, we reached an 84% success rate in the testing trials for *Reacquisition*, demonstrating the TD3+Track-HCL is capable of fulfilling task *Reacquisition*.

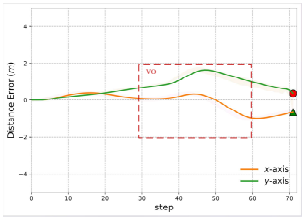
3) Test Scenario C: UGV movement along a lateral and backward trajectory: To evaluate the boundary of our method's cognitive capabilities in achieving task *Reacquisition*, we designed Test Scenario C. We set the UGV to move laterally and then backward through a VO. The experimental setup for this scenario is illustrated in Fig.11(a), and the averaged 3D trajectories of the UAV and UGV are shown in Fig.11(b). The 3D trajectory of the UAV and the UGV is illustrated in Fig. 11(b). As depicted in Fig. 11(b), the UAV initially succeeds in following the UGV's trajectory but continues forward even after the UGV enters the VO. To further challenge our method's reacquisition ability under evasive maneuvers, the UGV was intentionally maneuvered backward within the VO. Our method successfully reacquired



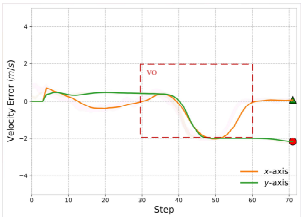
(a) Testing diagram of Scenario C in the Gazebo simulator.



(b) Averaged trajectories of tracking a moving UGV.



(c) Averaged distance error.



(d) Averaged velocity error.

Fig. 11. Illustration of the averaged trajectory, distance and velocity error for our TD3+Track-HCL in Test Scenario C.

tracking only if the UGV reappeared in the UAV's FOV before the UAV departed from the tracking region.

Figures 11(c) and 11(d) present the averaged distance and velocity errors between the UAV and UGV during tracking. The reacquisition task achieved a 63% success rate, highlighting the TD3+Track-HCL model's potential for tracking an evasively maneuvering hostile UGV. However, this success rate is significantly lower compared to Test Scenario B, primarily due to the UAV's limited FOV and the narrow training scope of the reward function.

D. Real-world Experiments

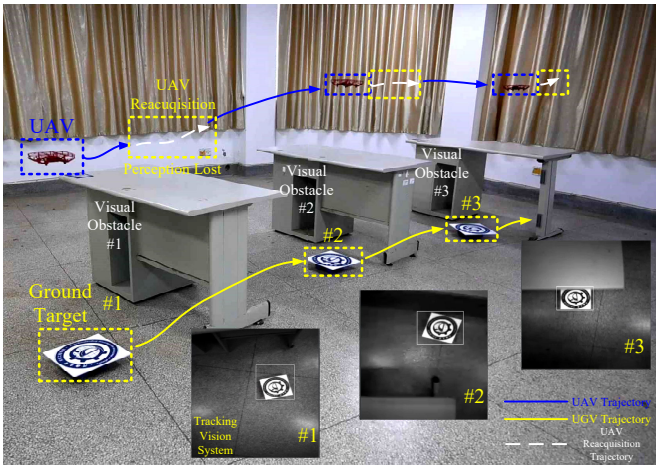
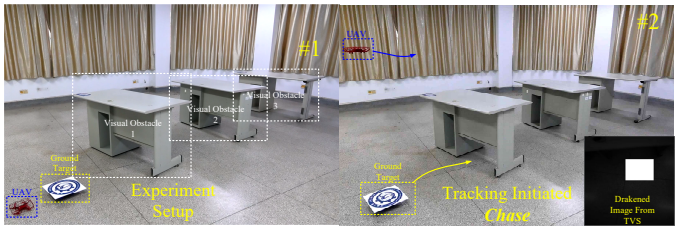


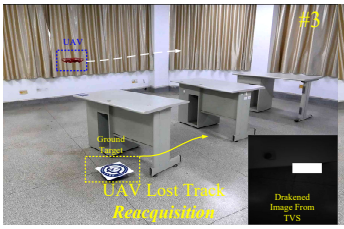
Fig. 12. Field experiment demonstration with visualization of generated trajectories in a indoor environment containing three static obstacles. Blue and yellow lines denote the UAV and UGV paths, respectively. UAV: DJI Robomaster TT (25 Hz gray-scale camera; max speed 1.2 m/s); UGV: RoboDK RDK-X3 (max speed 1 m/s). Insets show TVS detection bounding boxes under varied lighting conditions.

To evaluate the performance of our Track-HCL and TVS, we applied the optimal tracking policies learned by TD3+Track-HCL in real-world environments. In our field experiments, we use a Robomaster TT [3] as the validation UAV. A gray-scale camera is rigidly mounted below the UAV, publishing images at 25 Hz. We use a D-Robotics Developer Kit X3[4] (RDK X3) onboard computer as the ground station for UAV control. A visual marker is mounted on the top plate of our UGV. We first initiated the tracking by receiving images of the ground target, and then we used the ground station for TVS processing and UAV control. Our method achieves a real-time running speed of 17 FPS on an onboard computer using the TVS approach. We conducted 5 tracking trails in indoor environments; one of the trails is shown in Fig. 13. We set up 3 static obstacles as the VO. The UGV is set to pass under the VOs constantly, forcing the UAV to fulfill both *Chase* and *Reacquisition*.



(a)

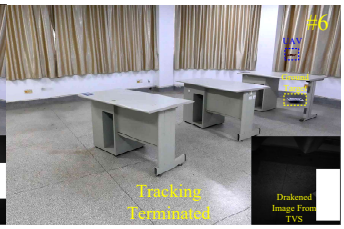
(b)



(c)

(d)

(e)



(f)

Fig. 13. Tracking experiments in indoor environments with obstacles.

Fig. 13 illustrates our tracking experiment. We observe that first, the TVS was able to identify the designated ground target using the SIFT Extractor (see Fig. 9(b)). As the UGV was tracked by the TVS and moved forward, the UAV could catch up with it at $t = 2s$, thus fulfilling task *Chase*. We then observed that at $t = 5s$, as the target UGV passes under the obstacles 1 and the UAV loses track, the UAV quickly flew toward where the target had disappeared, as shown in Fig. 9(c). Furthermore, Fig. 9(d) demonstrates that at $t =$

[3]https://www.dji.com/uk/robomaster-tt

[4]https://developer.d-robotics.cc/

10s, the UAV successfully regained track of the UGV both visually and motionally to fulfill task *Reacuqisition*. The UGV then drove through VO 2 and VO 3 to validate the trained policy's capability for our UAV to fulfill both task *Chase* and *Reacquisition*. The results show that the UAV can cognitively manage the tracking task process with or temporarily without the perception of the UGV. Furthermore, we reached an 80% success rate in the conducted experiments. The learned strategies were successfully implemented in real-world environments without requiring any parameter tuning, which verifies the stability of our method.

Despite the overall success, we encountered a significant failure during one of the trials, which highlighted potential areas for improvement. In this particular instance, as shown in Fig. 14, the UAV failed to reacquire the UGV after losing track at $t = 16s$.

E. Discussion

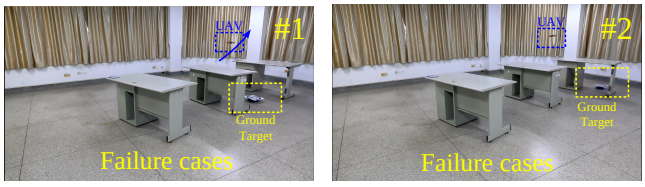
Although the UAV can learn satisfying tracking policies using the proposed method, several issues that arise from the simulation and real-world experiments are still worth discussing.

1) DRL-based Control Approach: In our study, we selected the Twin Delayed Deep Deterministic Policy Gradient (TD3) algorithm for UAV tracking in environments with continuous action spaces. This choice was predicated on TD3's demonstrated ability to effectively handle the complexities associated with such scenarios. However, implementing other DRL algorithm may achieve better result. TD-Lambda [40] is a feasible approach to leverages eligibility traces to balance between immediate and long-term reward assignments, which can be critical for the UAV to learning different policies from the task *Chase* and *Reacquisition*. The action space was two-dimensional, considering only the 3D relative position between the UAV and the UGV. It would be more informative to include the relative heading direction of the UGV from the perspective of the UAV. If communication was allowed between the UAV and the UGV, not only would the heading direction be easily available, but also its speed for constructing the system state representation. Furthermore, we introduced 3 weight coefficiencies to our reward function.(see Sec. V-D2), drawing inspiration from the works of Liang et.al. [41] and Zhao et.al. [42]. From a theoretical standpoint, larger weights on chase and reacquisition terms drive the agent to stay close to the target and quickly recover from tracking failures, respectively. By contrast, the termination reward establishes a strong incentive or penalty at episode completion, ensuring the policy takes into account success or failure over a full trajectory.

Additionally, several DRL algorithms including SAC have been explored in a previous works [43]. However, the SAC algorithm consistently exhibited instability during training. Consistent with these prior findings, we also tested an SAC-based baseline (SAC+Track-HCL) in this study, but similarly poor performance was observed. A plausible reason for SAC's suboptimal performance in our context is its inherent sensitivity to entropy temperature settings, which complicates the optimization process in highly dynamic UAV tracking tasks.

2) UAV Perception: In the advancement of our method alongside state-of-the-art (SOAT) algorithms for UAV tracking, we have extended our experimental framework to include SiamAPN++ [44]. However, our findings reaffirm that TCTrack remains the most robust and precise tracking algorithm for our specific requirements. The underlying reasons are multifaceted: Firstly, TCTrack excels at providing real-time tracking capabilities, utilizing a synergistic blend of robust algorithms and streamlined models adept at processing visual data efficiently. Such efficiency is paramount in UAV applications, where timely and accurate responses are essential to navigate dynamic environments effectively. Furthermore, its adaptability is particularly suited for UAV tracking, where unpredictability in environmental conditions is a routine challenge.

3) Limitations and Failure Scenarios: The failure occurred when the UGV passed beneath VO 3, causing the UAV to lose visual contact despite moving to the UGV's predicted location. Several factors might have caused these failures. First, the visual tracking algorithm exhibited insufficient robustness under abrupt lighting changes and complex occlusions, leading to an incomplete or delayed reacquisition of the target. Second, the UAV's prediction mechanism did not sufficiently account for abrupt changes in the UGV's speed and direction, resulting in a mismatch between predicted and actual positions. Third, the UAV's limited field of view (FOV) constrained the target's capture through the TVS, especially under dynamic scenarios with sudden maneuvers. Moreover, the DRL-trained post-tracking loss search strategy was not fully effective in recovering the UGV once it had disappeared from view. These findings underscore the need for enhanced resilience of both the TVS and CG methods when operating under challenging lighting conditions and among dense obstacles.



(a) (b)

Fig. 14. Failure case where the UAV loses track of the UGV and failed to reacquire it.

We use the combination of SIFT and TCTrack to achieve balance performance and computational cost in our TVS. Both methods demand relatively low computational resources compared to other deep learning-based lightweight target identification and tracking approaches. However, we have observed that TCTrack exhibited limited robustness under varying conditions—particularly when facing severe lighting changes or occlusions during our field experience. We notice that the spatial-channel Transformer-based low-light enhancer (SCT) method [45] can bolster the tracker's capability to

maintain reliable identification and tracking even when illumination levels are poor or highly dynamic. Therefore, one potential remedy is to replace TCTrack with the SCT. This enhancement may help mitigate the limitations observed in extreme conditions during field experiment, thereby improving the overall system robustness.

VII. CONCLUSION

In this paper, we have proposed a cognitive approach for autonomous UAV tracking of a moving UGV. The Track-HCL has been introduced to efficiently generate adaptive curriculums to guide the agent to learn the optimal target chasing and reacquisition policy. A vision system has been designed to track and retrack the target marker and extract motion trends of the UGV. Our proposed methods have exhibited satisfying performance in tracking and reacquisition tasks in both simulations and the field experiments.

For future work, we aim to enhance the perceptual ability of our tracking framework by applying semantic and anomaly detection algorithms to enable the agent to understand the relationships between obstacles. Furthermore, we plan to extend our tracking methods to multi-agent collaborations, and to incorporate TD-Lambda with our curriculum learning method.

REFERENCES

[1] Pratap Tokekar, Joshua Vander Hook, David Mulla, and Volkan Isler. Sensor planning for a symbiotic uav and ugv system for precision agriculture. *IEEE Transactions on Robotics*, 32(6):1498–1511, 2016.

[2] Pengfei Du, Yueqiang Shi, Haotong Cao, Sahil Garg, Mubarak Alrashoud, and Piyush Kumar Shukla. Ai-enabled trajectory optimization of logistics uavs with wind impacts in smart cities. *IEEE Transactions on Consumer Electronics*, 70(1):3885–3897, 2024.

[3] Hamid Menouar, Ismail Guvenc, Kemal Akkaya, A Selcuk Uluagac, Abdullah Kadri, and Adem Tuncer. Uav-enabled intelligent transportation systems for the smart city: Applications and challenges. *IEEE Communications Magazine*, 55(3):22–28, 2017.

[4] Shuang Qi, Bin Lin, Yiqin Deng, Xianhao Chen, and Yuguang Fang. Minimizing maximum latency of task offloading for multi-uav-assisted maritime search and rescue. *IEEE Transactions on Vehicular Technology*, 73(9):13625–13638, 2024.

[5] Pablo R Palafox, Mario Garzón, João Valente, Juan Jesús Roldán, and Antonio Barrientos. Robust visual-aided autonomous takeoff, tracking, and landing of a small uav on a moving landing platform for life-long operation. *Applied Sciences*, 9(13):2661, 2019.

[6] Ali Ghasemi, Farhad Parivash, and Serajeddin Ebrahimian. Autonomous landing of a quadrotor on a moving platform using vision-based fofpid control. *Robotica*, 40(5):1431–1449, 2022.

[7] Justin Thomas, Jake Welde, Giuseppe Loianno, Kostas Daniilidis, and Vijay Kumar. Autonomous flight for detection, localization, and tracking of moving targets with a small quadrotor. *IEEE Robotics and Automation Letters*, 2(3):1762–1769, 2017.

[8] Bryan Penin, Paolo Robuffo Giordano, and François Chaumette. Vision-based reactive planning for aggressive target tracking while avoiding collisions and occlusions. *IEEE Robotics and Automation Letters*, 3(4):3725–3732, 2018.

[9] Youcef Mezouar and François Chaumette. Path planning for robust image-based control. *IEEE transactions on robotics and automation*, 18(4):534–549, 2002.

[10] Jingjing Gu, Tao Su, Qiuhong Wang, Xiaojiang Du, and Mohsen Guizani. Multiple moving targets surveillance based on a cooperative network for multi-uav. *IEEE Communications Magazine*, 56(4):82–89, 2018.

[11] Senqiang Zhu, Danwei Wang, and Chang Boon Low. Ground target tracking using uav with input constraints. *Journal of Intelligent & Robotic Systems*, 69:417–429, 2013.

[12] Xuancen Liu, Yueneng Yang, Chenxiang Ma, Jie Li, and Shifeng Zhang. Real-time visual tracking of moving targets using a low-cost unmanned aerial vehicle with a 3-axis stabilized gimbal system. *Applied Sciences*, 10(15):5064, 2020.

[13] Ziyuan Huang, Changhong Fu, Yiming Li, Fuling Lin, and Peng Lu. Learning aberrance repressed correlation filters for real-time uav tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2891–2900, October 2019.

[14] Shuaijun Wang, Fan Jiang, Bin Zhang, Rui Ma, and Qi Hao. Development of uav-based target tracking and recognition systems. *IEEE Transactions on Intelligent Transportation Systems*, 21(8):3409–3422, 2019.

[15] Yiming Li, Changhong Fu, Fangqiang Ding, Ziyuan Huang, and Geng Lu. Autotrack: Towards high-performance visual tracking for uav with automatic spatio-temporal regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11923–11932, June 2020.

[16] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[17] Shuiwang Li, Xiangyang Yang, Xucheng Wang, Dan Zeng, Hengzhou Ye, and Qijun Zhao. Learning target-aware vision transformers for real-time uav tracking. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–18, 2024.

[18] Ziyue Lin, Wenbo Xu, and Wei Wang. A moving target tracking system of quadrotors with visual-inertial localization. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3296–3302, 2023.

[19] Jiang Zhao, Han Liu, Jiaming Sun, Kun Wu, Zhihao Cai, Yan Ma, and Yingxun Wang. Deep reinforcement learning-based end-to-end control for uav dynamic target tracking. *Biomimetics*, 7(4):197, 2022.

[20] Sheng Zhang, Zhe Zhang, and Siva Theja Maguluri. Finite sample analysis of average-reward td learning and q-learning. *Advances in Neural Information Processing Systems*, 34:1230–1242, 2021.

[21] Shiyu Huang, Bin Wang, Hang Su, Dong Li, Jianye Hao, Jun Zhu, and Ting Chen. Off-policy training for truncated td (λ) boosted soft actor-critic. In *PRICAI 2021: Trends in Artificial Intelligence*, pages 46–59. Springer, 2021.

[22] Kewei Xia, Xinyi Li, Kaidan Li, Yao Zou, and Zongyu Zuo. Cooperative tracking of quadrotor uavs using parallel optimal learning control. *IEEE Transactions on Automation Science and Engineering*, 22:3308–3319, 2025.

[23] Seungyeon Yoo, Seungwoo Jung, Yunwoo Lee, Dongseok Shim, and H. Jin Kim. Mono-camera-only target chasing for a drone in a dense environment by cross-modal learning. *IEEE Robotics and Automation Letters*, 9(8):7254–7261, 2024.

[24] Parakh M Gupta, Ondřej Procházka, Tiago Nascimento, and Martin Saska. Curvitrack: Curvilinear trajectory tracking for high-speed chase of a usv. *IEEE Robotics and Automation Letters*, 0(99):1–8, 2025.

[25] Guohuai Lin, Hongyi Li, Choon Ki Ahn, and Deyin Yao. Event-based finite-time neural control for human-in-the-loop uav attitude systems. *IEEE Transactions on Neural Networks and Learning Systems*, 34(12):10387–10397, 2023.

[26] Sanmit Narvekar, Bei Peng, Matteo Leonetti, Jivko Sinapov, Matthew E Taylor, and Peter Stone. Curriculum learning for reinforcement learning domains: A framework and survey. *The Journal of Machine Learning Research*, 21(1):7382–7431, 2020.

[27] Tambet Matiisen, Avital Oliver, Taco Cohen, and John Schulman. Teacher–student curriculum learning. *IEEE transactions on neural networks and learning systems*, 31(9):3732–3740, 2019.

[28] Steven D Morad, Roberto Mecca, Rudra PK Poudel, Stephan Liwicki, and Roberto Cipolla. Embodied visual navigation with automatic curriculum learning in real environments. *IEEE Robotics and Automation Letters*, 6(2):683–690, 2021.

[29] Sébastien Forestier, Rémy Portelas, Yoan Mollard, and Pierre-Yves Oudeyer. Intrinsically motivated goal exploration processes with automatic curriculum learning. *The Journal of Machine Learning Research*, 23(1):6818–6858, 2022.

[30] Honghu Xue, Benedikt Hein, Mohamed Bakr, Georg Schildbach, Bengt Abel, and Elmar Rueckert. Using deep reinforcement learning with automatic curriculum learning for mapless navigation in intralogistics. *Applied Sciences*, 12(6):3153, 2022.

[31] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60:91–110, 2004.

[32] Ziang Cao, Ziyuan Huang, Liang Pan, Shiwei Zhang, Ziwei Liu, and Changhong Fu. Tctrack: Temporal contexts for aerial tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14798–14808, June 2022.

[33] Francois Chollet. Xception: Deep Learning with Depthwise Separable Convolutions . In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1800–1807, Los Alamitos, CA, USA, July 2017. IEEE Computer Society.

[34] Scott Fujimoto, Herke van Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1587–1596. PMLR, 10–15 Jul 2018.

[35] Jiawei Wang, Teng Wang, Zichen He, Wenzhe Cai, and Changyin Sun. Towards better generalization in quadrotor landing using deep reinforcement learning. *Applied Intelligence*, 53(6):6195–6213, 2023.

[36] Chao Yan, Xiaojia Xiang, and Chang Wang. Towards real-time path planning through deep reinforcement learning for a uav in dynamic environments. *Journal of Intelligent & Robotic Systems*, 98:297–309, 2020.

[37] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1861–1870. PMLR, 10–15 Jul 2018.

[38] Riccardo Polvara, Sanjay Sharma, Jian Wan, Andrew Manning, and Robert Sutton. Autonomous vehicular landings on the deck of an unmanned surface vehicle using deep reinforcement learning. *Robotica*, 37(11):1867–1882, 2019.

[39] Andrew L Maas, Awni Y Hannun, Andrew Y Ng, et al. Rectifier nonlinearities improve neural network acoustic models. In *Proc. icml*, volume 30, page 3. Atlanta, Georgia, USA, 2013.

[40] Harm Seijen and Rich Sutton. True online td(lambda). In Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 692–700, Bejing, China, 22–24 Jun 2014. PMLR.

[41] Jinhao Liang, Kaidi Yang, Chaopeng Tan, Jinxiang Wang, and Guodong Yin. Enhancing high-speed cruising performance of autonomous vehicles through integrated deep reinforcement learning framework. *IEEE Transactions on Intelligent Transportation Systems*, 26(1):835–848, 2025.

[42] Wang Zhao, Ye Zhang, and Zikang Xie. Eppe: An efficient progressive policy enhancement framework of deep reinforcement learning in path planning. *Neurocomputing*, 596:127958, 2024.

[43] Chang Wang, Jiaqing Wang, Zhaowei Ma, Mingjin Xu, Kailei Qi, Ze Ji, and Changyun Wei. Integrated learning-based framework for autonomous quadrotor uav landing on a collaborative moving ugv. *IEEE Transactions on Vehicular Technology*, 73(11):16092–16107, 2024.

[44] Ziang Cao, Changhong Fu, Junjie Ye, Bowen Li, and Yiming Li. Siamapn++: Siamese attentional aggregation network for real-time uav tracking. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3086–3092, 2021.

[45] Junjie Ye, Changhong Fu, Ziang Cao, Shan An, Guangze Zheng, and Bowen Li. Tracker meets night: A transformer enhancer for uav tracking. *IEEE Robotics and Automation Letters*, 7(2):3866–3873, 2022.