UNIFollow: Natural Language Conditioned Target Following Framework for UAVs

V. R. Vasudevan[1], Suhani Grover[2], and Indu Sreedevi[3]

***Abstract*— Autonomous UAVs operating in dynamic environments face significant challenges when tasked with tracking arbitrary targets described in natural language, particularly in unstructured scenarios where traditional closed-world tracking systems fail. This work presents a novel open-vocabulary UAV target tracking framework that integrates vision-language models with classical tracking algorithms to enable real-time reactive control in dynamic mountainous environments. Our approach combines OWL-ViT for zero-shot object detection, CSRT for efficient tracking, and a hybrid control architecture featuring gimbal-based localization for distant targets and reinforcement learning-assisted visual servoing for precise following. The RL-adapted PD controller demonstrates robust performance across varying target velocities where traditional PD controllers fail, addressing the critical need for real-time reactivity and smooth trajectory generation. We validate our framework in AirSim's mountainous terrain with configurable vehicle dynamics, demonstrating stable tracking performance despite challenging viewing angles and environmental disturbances. Our modular architecture enables natural language target specification without predefined object classes, contributing to more adaptable and trustworthy robotic systems for search-and-rescue and surveillance applications in dynamic environments.**

I. INTRODUCTION

The mobility and wide-area perceptive capabilities of aerial robots make them indispensable platforms for surveillance, search-and-rescue, and environmental monitoring. In particular, the ability of a UAV to locate and follow a target is fundamental to these operations. Traditional approaches focus on building task-specific detectors coupled with lightweight trackers [1], which work efficiently in constrained environments but remain limited by closed-world assumptions, i.e. the UAV can only recognize and track predefined object categories. However, operational scenarios are highly unstructured, requiring UAVs to follow arbitrary targets described by human operators in natural language.

Recent advances in Vision-Language Models (VLMs) and Large Language Models (LLMs) have revolutionized how machines interpret multimodal information [2], enabling open-vocabulary recognition and reduced dependence on task-specific training. Vision-Language-Action (VLA) models have shown remarkable capabilities in robotics applications [3], [4], [5], offering new paradigms for natural language-guided robot control. These developments present opportunities to overcome the limitations of traditional tracking systems by integrating natural language understanding with visual perception. While works such as [6] leverage VLMs to generate a goal based on natural language description, they are designed for static goals and may perform poorly on active tracking. In this work, we propose a robust UAV target tracking framework for unstructured environments, with applications in search and rescue. We integrate the open-vocabulary OWL-ViT [7] for target identification with Channel and Spatial Reliability Tracker (CSRT) [8] tracker for target tracking. The perception head is coupled with a hybrid control architecture combining gimbal-based positioning for distant targets and reinforcement learning assisted visual servoing for robust following. The key contributions our work are summarized below:

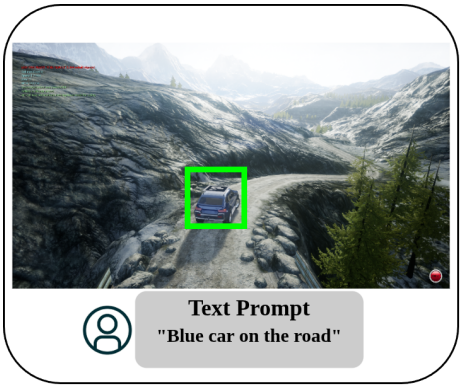

Fig. 1: Text-prompted detection in AirSim Mountain Landscape. Query "blue car on the road" yields successful target detection and tracking.

- A simulation environment based on AirSim and Gymnasium API, with user-configurable vehicle speeds and predefined traversal paths in the Landscape Mountain scenario, enabling systematic evaluation of tracking algorithms across diverse dynamic conditions.
- An efficient open-vocabulary target detection and tracking pipeline integrating OWL-ViT with CSRT tracker for robust and lightweight re-identification of target in real-time.
- A hybrid control framework combining gimbal-based localization for distant target with pure reinforcement learning-assisted PD visual servoing, enabling both long-range target acquisition and precise tracking for variable speed vehicle following.

[1]V. R. Vasudevan is with the Department of Mechanical Engineering, Delhi Technological University, Delhi, India. vasudevanvr2002@gmail.com

[2]Suhani Grover is with the Department of Applied Physics, Delhi Technological University, Delhi, India. suhani1077@gmail.com

[3]Indu Sreedevi is with the Department of Electronics and Communication Engineering, Delhi Technological University, Delhi, India. s.indu@dtu.ac.in

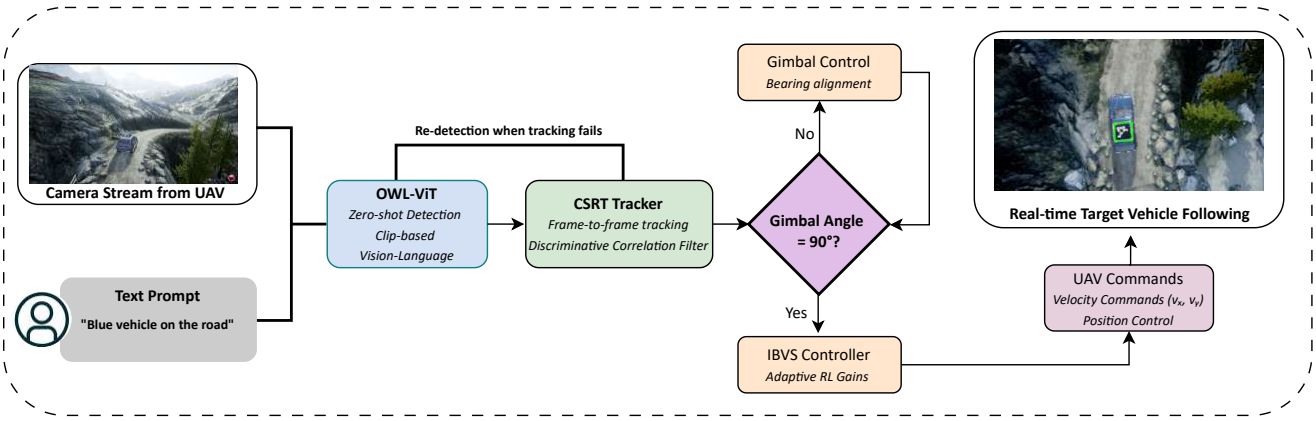

Fig. 2: System architecture of the proposed VLM-CSRT-RL pipeline for open-vocabulary UAV target tracking. The framework integrates OWL-ViT for zero-shot detection, CSRT for tracking, and hybrid control that switches between gimbal positioning for distant targets and RL-assisted IBVS for precise tracking based on camera angle.

II. Related Work

Advances in Vision-Language-Action (VLA) models have demonstrated remarkable capabilities in unifying language instructions with robot control. End-to-end VLA systems such as RT-2 [9], OpenVLA [10], $\pi 0$ [11], map natural language commands to robot actions, achieving impressive performance across diverse manipulation tasks. Recent work has extended VLA approaches to visual tracking, with TrackVLA [12] introducing an architecture combining trajectory planning and target recognition using shared LLM backbones. These approaches excel at generalization and semantic reasoning but often require extensive training, high computational resources, and lack modularity and transparency. While VLA models have shown promise in terrestrial robotics and aerial navigation tasks like AerialVLN [13], their application to dynamic UAV tracking scenarios remains limited, particularly for real-time target following with non-cooperative mobile objects.

Our work lies at the intersection of open-vocabulary perception and language-guided aerial robotics. Open-vocabulary detectors such as OWL-ViT [7], GroundingDINO [14] and YOLO-World [15] enable zero-shot object detection from text prompts beyond fixed dataset categories. We use OWL-ViT as our detection backbone due to its optimized inference speed for real-time applications. GroundingDINO's computational overhead limits real-time performance and YOLO-World showed insufficient accuracy for precise UAV tracking scenarios. To achieve real-time performance, we adopt a tracking-by-detection framework using a high-accuracy detector for initialization and a lightweight tracker for frame-to-frame localization [8]. Coupled with reinforcement learning policies, these modular systems provide adaptive decision-making with lower computational demands than end-to-end VLAs. Simulation platforms like AirSim [16] have further established strong baselines for RL-based UAV navigation, though their integration with open-vocabulary perception for dynamic tracking remains underexplored.

III. Methodology

A. System Architecture

Our proposed pipeline (as shown in Fig. 2) operates in real-time, processing RGB camera input $I_t \in \mathbb{R}^{H\times W\times 3}$ along with natural language target descriptions ℓ through three components: (1) OWL-ViT [7] for zero-shot text-prompted detection, (2) CSRT [8] for efficient frame-to-frame tracking, and (3) control for UAV navigation. Unlike end-to-end VLA approaches that require extensive training [12], [9], this tracking by detection approach enables real-time performance with reduced computational requirements while maintaining interpretability. Text queries initialize OWL-ViT to identify bounding boxes $\mathscr{B}_t = \{x, y, w, h, c\}$, which are then maintained by CSRT across frames, while control policies generate UAV commands.

B. Text-Prompted Object Detection

We employ OWL-ViT base-patch32 model [7] for zero-shot object detection. The system accepts natural language queries (e.g., "person," "red car") and processes them alongside RGB frames converted from BGR camera input. OWL-ViT leverages CLIP-based [17] vision-language alignment to compute similarity between visual patch embeddings and text query embeddings, enabling open-vocabulary detection without object-specific models.

Text queries are tokenized and encoded into embedding space $\mathbf{q} \in \mathbb{R}^d$, while visual features are extracted as patch-level embeddings $\mathbf{v}_i \in \mathbb{R}^d$. Object localization computes similarity scores $s_i = \cos(\mathbf{q}, \mathbf{v}_i)$ followed by post-processing to generate bounding box predictions. We implement a confidence-based detection threshold for tracker initialization and during detection mode, the system evaluates all detected objects and selects the highest-confidence bounding box exceeding this threshold. Upon successful detection, bounding box coordinates are converted from OWL-ViT format (x_1, y_1, x_2, y_2) to CSRT format (x, y, w, h) for tracker initialization.

C. Object Tracking

Channel and Spatial Reliability Tracker (CSRT) employs discriminative correlation filters with channel and spatial reliability measures, maintaining computational efficiency while handling scale variations and partial occlusions common in UAV applications. Tracker initialization occurs when OWL-ViT detection confidence exceeds τ_{detect}. During tracking mode, CSRT updates target position each frame, returning tracking status and updated bounding box coordinates. We implement a robust re-detection mechanism based on tracking success feedback. When potential target loss occurs due to occlusion, motion blur, or scene changes, the system transitions back to detection mode and reinvokes OWL-ViT for target reacquisition. This hybrid approach ensures long-term tracking robustness with CSRT providing efficient frame-to-frame tracking and OWL-ViT handling challenging re-detection scenarios when classical tracking fails.

D. UAV Position Control Policy

Our algorithm works by maintaining the target in the camera view by directly designing the control law based on the pixel coordinate of the target. The operator can move the gimbal in the direction of preference and can describe the target in natural language. Once detected, the algorithm works in two stages, described below:

1) Long distance trackng: When the target is detected within the camera's field of view at a non-vertical pitch angle (i.e., when the gimbal is not oriented at -90°), a target tracking stage is activated. In this stage, the gimbal is actuated to align the target with the center of the onboard camera image. The angles commanded are calculated based on the camera intrinsics [18]. Once alignment is achieved, the corresponding gimbal yaw command is mapped to the UAV's yaw control, ensuring that the vehicle is oriented directly toward the target. At this point, the gimbal yaw angle converges to 0°, as the target lies along the forward axis of the UAV.

Subsequently, a constant forward velocity command is applied to drive the UAV toward the target. As the vehicle approaches, the gimbal continuously adjusts its pitch angle to maintain the target at the center of the camera frame. When the gimbal pitch angle reaches a threshold of $-90^\circ \pm 5^\circ$, the gimbal is locked at -90°, under the assumption that the target remains visible in the downward-looking configuration. At this transition point, the system activates a secondary reinforcement learning-based proportional–derivative (PD) control module, which refines the UAV's position through fine-scale adjustments.

When targets are visible in the downward camera view, we switch to Image-Based Visual Servoing (IBVS) controller for precise target following and maintaining tracking stability. This dual mode approach combines geometric localization for navigating to distant visible targets with adaptive visual servoing, allowing the UAV to locate distant objects while maintaining agility for nearby target tracking scenarios.

2) Image-Based Visual Servoing (IBVS): We use an Proportional-Derivative (PD) Image-Based Visual Servoing (IBVS) controller with reinforcement learning-optimized gains. Traditional PD controllers use fixed gains and struggles with generalizing across varying speeds and environmental conditions. Our approach solves this issue by incorporating a learn gain-predicting model. We train the policy with the Soft Actor-Critic (SAC) method [19] for continuous control optimization due to its sample efficiency and stability in high-dimensional action spaces. The IBVS controller operates on normalized target center coordinates (x_t, y_t) from bounding box detection:

$$v_x = K_{p,x} \cdot e_x + K_{d,x} \cdot \dot{e}_x \tag{1}$$

$$v_y = K_{p,y} \cdot e_y + K_{d,y} \cdot \dot{e}_y \tag{2}$$

where $e_x = x_t - x_c$, $e_y = y_t - y_c$. x_c and y_c are the normalized center coordinate(0.5, 0.5). (v_x, v_y) are commanded velocities.

3) RL Formulation: We cast the gain adaptation problem as a continuous control problem:

a) State space: At each timestep, the agent observes

$$\mathbf{s}_t = \begin{bmatrix} e_x & e_y & \dot{x}_t & \dot{y}_t & \ddot{x}_t & \ddot{y}_t \end{bmatrix}^T, \tag{3}$$

which encodes the target's image-plane position, velocity, and acceleration. This representation provides the agent with sufficient information about target dynamics to modulate control gains.

b) Action space: The agent outputs

$$\mathbf{a}_t = \begin{bmatrix} K_{p,x} & K_{p,y} & K_{d,x} & K_{d,y} \end{bmatrix}^T, \tag{4}$$

where each element is bounded to ensure stability and safety.

c) Reward function: The reward is designed to penalize large tracking errors and excessive control effort, encouraging precise yet smooth tracking:

$$r_t = -\left(\lambda_e (e_x^2 + e_y^2)\right), \tag{5}$$

with $\lambda_e > 0$ as weighting factor.

4) Curriculum Learning Strategy: To enable robust generalization across varying target velocities, we employ a three-stage curriculum learning approach for training the SAC agent. The policy is first trained on a stationary target for 50 episodes of 300 steps to establish basic tracking capabilities. Training then continues for 50 episodes of 1000 steps with target velocity incrementally increasing from 1 m/s to 5 m/s in 1 m/s intervals every 10 episodes. Finally, the policy is trained for 100 episodes of 1000 steps with velocities randomly sampled for each episode, ensuring robust performance across the entire operational velocity range.

IV. Experimental Setup and Results

A. Simulation Environment

We test our approach on the Microsoft AirSim [16] simulator in Landscape Mountains environment (Fig. 3). Airsim's car mesh was modified to take commands from a python script to move at specified velocities along preset

Fig. 3: Microsoft AirSim Landscape Mountain environment

routes, enabling systematic evaluation across varied scenarios. The UAV operates with AirSim's multirotor API. The simulation environment is wrapped in a Gymnasium environment wrapper [20] for RL training integration. We used the SAC implementation from Stable Baselines3 [21]. The car location serves as the ground truth for tracking evaluation against UAV position.

B. Evaluation Metrics

We assess performance using four metrics: (1) Root Mean Square Error (RMSE) between UAV and target in image coordinates for tracking accuracy; (2) Control Effort, defined as the magnitude of commanded linear acceleration, for flight efficiency; (3) Tracking Duration, the time the UAV successfully follows the target before divergence; and (4) Frames Per Second (FPS) for computational feasibility. Lower RMSE and effort, longer durations, and stable FPS denote better performance across vehicle velocities.

C. Results

We evaluated our RL-assisted controller against a fine-tuned baseline PD controller at 1 m/s and 5 m/s target velocities. Table I shows that the RL controller achieves substantially lower RMSE (45-60% improvement) and reduced control effort (18-22% reduction), with tracking trajectories visualized in Fig. 4 and Fig. 5.

TABLE I: Comparison of PD and RL controllers across velocities with RMSE, control effort, and tracking duration.

Method	Velocity (m/s)	RMSE	Control Effort	Duration (s)
PD	1	6.38	0.043	20.6
PD	5	7.47	0.054	17.4
RL	1	3.50	0.035	19.4
RL	5	2.98	0.042	17.2

Despite OWL-ViT's transformer backbone not being designed for real-time operation, our system achieves 13.65 FPS, enabling near-real-time UAV deployment. The RL method introduces minimal computational overhead while validating the effectiveness of our adaptive control strategy for dynamic target following.

V. CONCLUSION

We present an open-vocabulary UAV target tracking framework integrating OWL-ViT detection, CSRT tracking,

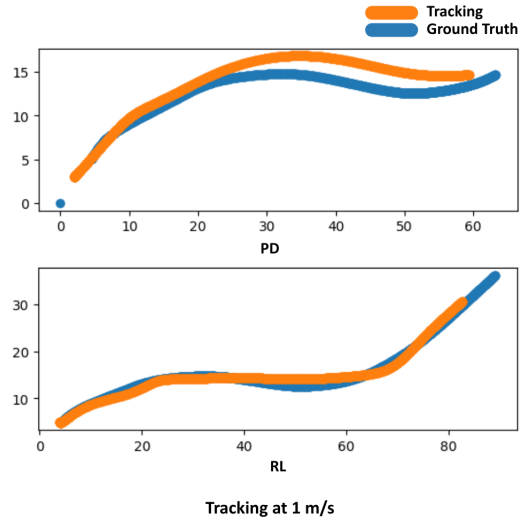

Fig. 4: PD vs RL Tracking coordinates for 1 m/s

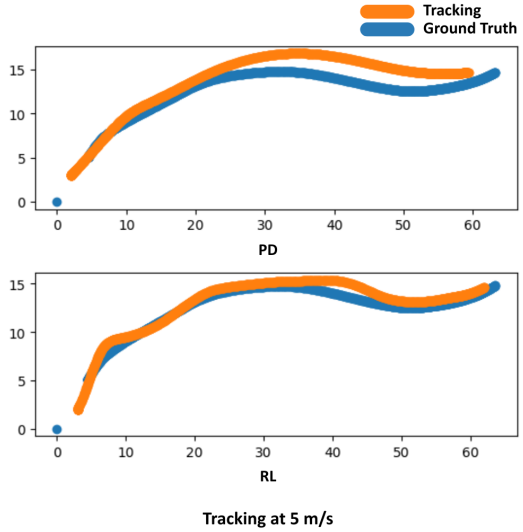

Fig. 5: PD vs RL Tracking coordinates for 5 m/s

and a hybrid control architecture with RL-assisted visual servoing. Curriculum learning-based SAC training enables the controller to robustly follow targets, outperforming fixed-gain PD controllers in tracking accuracy while requiring 20-40% less control effort. The gimbal-based approach handles distant target localization while adaptive IBVS ensures precise tracking at close range. Open-vocabulary detection allows natural language target specification without predefined classes, enhancing flexibility for search-and-rescue and surveillance missions. The modular architecture maintains real-time operation while providing interpretability across diverse scenarios. Future work will focus on fine-tuning detection models on aerial datasets to improve robustness and computational efficiency, developing advanced prompt engineering for complex target descriptions, investigating lightweight vision-language models for embedded platforms, and transitioning to real-world deployment with comprehensive field validation.

REFERENCES

[1] M. Rizk, F. Slim, and J. Charara, "Toward ai-assisted uav for human detection in search and rescue missions," in *2021 International Conference on Decision Aid Sciences and Application (DASA)*, 2021, pp. 781–786.

[2] X. Han, S. Chen, Z. Fu, Z. Feng, L. Fan, D. An, C. Wang, L. Guo, W. Meng, X. Zhang, R. Xu, and S. Xu, "Multimodal fusion and vision-language models: A survey for robot vision," 2025. [Online]. Available: https://arxiv.org/abs/2504.02477

[3] K. Kawaharazuka, J. Oh, J. Yamada, I. Posner, and Y. Zhu, "Vision-language-action models for robotics: A review towards real-world applications," *IEEE Access*, vol. 13, pp. 162 467–162 504, 2025.

[4] R. Sapkota, Y. Cao, K. I. Roumeliotis, and M. Karkee, "Vision-language-action models: Concepts, progress, applications and challenges," 2025. [Online]. Available: https://arxiv.org/abs/2505.04769

[5] M. U. Din, W. Akram, L. S. Saoud, J. Rosell, and I. Hussain, "Vision language action models in robotic manipulation: A systematic review," 2025. [Online]. Available: https://arxiv.org/abs/2507.10672

[6] Y. Zhang, H. Yu, J. Xiao, and M. Feroskhan, "Grounded vision-language navigation for uavs with open-vocabulary goal understanding," 2025. [Online]. Available: https://arxiv.org/abs/2506.10756

[7] M. Minderer, A. Gritsenko, A. Stone, M. Neumann, D. Weissenborn, A. Dosovitskiy, A. Mahendran, A. Arnab, M. Dehghani, Z. Shen, X. Wang, X. Zhai, T. Kipf, and N. Houlsby, "Simple open-vocabulary object detection with vision transformers," 2022. [Online]. Available: https://arxiv.org/abs/2205.06230

[8] A. Lukežič, T. Vojíř, L. Čehovin Zajc, J. Matas, and M. Kristan, "Discriminative correlation filter tracker with channel and spatial reliability," *International Journal of Computer Vision*, vol. 126, no. 7, p. 671–688, Jan. 2018. [Online]. Available: http://dx.doi.org/10.1007/s11263-017-1061-3

[9] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn, P. Florence, C. Fu, M. G. Arenas, K. Gopalakrishnan, K. Han, K. Hausman, A. Herzog, J. Hsu, B. Ichter, A. Irpan, N. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, L. Lee, T.-W. E. Lee, S. Levine, Y. Lu, H. Michalewski, I. Mordatch, K. Pertsch, K. Rao, K. Reymann, M. Ryoo, G. Salazar, P. Sanketi, P. Sermanet, J. Singh, A. Singh, R. Soricut, H. Tran, V. Vanhoucke, Q. Vuong, A. Wahid, S. Welker, P. Wohlhart, J. Wu, F. Xia, T. Xiao, P. Xu, S. Xu, T. Yu, and B. Zitkovich, "Rt-2: Vision-language-action models transfer web knowledge to robotic control," 2023. [Online]. Available: https://arxiv.org/abs/2307.15818

[10] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi, Q. Vuong, T. Kollar, B. Burchfiel, R. Tedrake, D. Sadigh, S. Levine, P. Liang, and C. Finn, "Openvla: An open-source vision-language-action model," 2024. [Online]. Available: https://arxiv.org/abs/2406.09246

[11] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter, S. Jakubczak, T. Jones, L. Ke, S. Levine, A. Li-Bell, M. Mothukuri, S. Nair, K. Pertsch, L. X. Shi, J. Tanner, Q. Vuong, A. Walling, H. Wang, and U. Zhilinsky, "π_0: A vision-language-action flow model for general robot control," 2024. [Online]. Available: https://arxiv.org/abs/2410.24164

[12] S. Wang, J. Zhang, M. Li, J. Liu, A. Li, K. Wu, F. Zhong, J. Yu, Z. Zhang, and H. Wang, "Trackvla: Embodied visual tracking in the wild," *arXiv preprint arXiv:2505.23189*, 2025. [Online]. Available: http://arxiv.org/abs/2505.23189

[13] Y. Gao, Z. Wang, P. Han, L. Jing, D. Wang, and B. Zhao, "Exploring spatial representation to enhance llm reasoning in aerial vision-language navigation," 2025. [Online]. Available: https://arxiv.org/abs/2410.08500

[14] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, Q. Jiang, C. Li, J. Yang, H. Su, J. Zhu, and L. Zhang, "Grounding dino: Marrying dino with grounded pre-training for open-set object detection," 2024. [Online]. Available: https://arxiv.org/abs/2303.05499

[15] T. Cheng, L. Song, Y. Ge, W. Liu, X. Wang, and Y. Shan, "Yolo-world: Real-time open-vocabulary object detection," 2024. [Online]. Available: https://arxiv.org/abs/2401.17270

[16] S. Shah, D. Dey, C. Lovett, and A. Kapoor, "Airsim: High-fidelity visual and physical simulation for autonomous vehicles," in *Field and Service Robotics*, 2017. [Online]. Available: https://arxiv.org/abs/1705.05065

[17] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh,

S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," 2021. [Online]. Available: https://arxiv.org/abs/2103.00020

[18] S. Sanyal, S. Bhushan, and K. Sivayazi, "Detection and location estimation of object in unmanned aerial vehicle using single camera and gps," in *2020 First International Conference on Power, Control and Computing Technologies (ICPC2T)*, 2020, pp. 73–78.

[19] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," 2018. [Online]. Available: https://arxiv.org/abs/1801.01290

[20] M. Towers, A. Kwiatkowski, J. Terry, J. U. Balis, G. De Cola, T. Deleu, M. Goulão, A. Kallinteris, M. Krimmel, A. KG *et al.*, "Gymnasium: A standard interface for reinforcement learning environments," *arXiv preprint arXiv:2407.17032*, 2024.

[21] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, "Stable-baselines3: Reliable reinforcement learning implementations," *Journal of Machine Learning Research*, vol. 22, no. 268, pp. 1–8, 2021. [Online]. Available: http://jmlr.org/papers/v22/20-1364.html