Learn How to See: Collaborative Embodied Learning for Object Detection and Camera Adjusting

Lingdong Shen[1,2], Chunlei Huo[1,3,4*], Nuo Xu[5], Chaowei Han[1,2], Zichen Wang[1,2]

[1]School of Artificial Intelligence, University of Chinese Academy of Sciences
[2]MAIS, Institute of Automation, Chinese Academy of Sciences
[3]School of Information Engineering, Capital Normal University
[4]NLPR, Institute of Automation, Chinese Academy of Sciences
[5]Zhejiang Lab
{shenlingdong,hanchaowei,wangzichen}2022@ia.ac.cn,
clhuo@nlpr.ia.ac.cn, nuo.xu@zhejianglab.com

Abstract

Passive object detectors, trained on large-scale static datasets, often overlook the feedback from object detection to image acquisition. Embodied vision and active detection mitigate this issue by interacting with the environment. Nevertheless, the materialization of activeness hinges on resource-intensive data collection and annotation. To tackle these challenges, we propose a collaborative student-teacher framework. Technically, a replay buffer is built based on the trajectory data to encapsulate the relationship of state, action, and reward. In addition, the student network diverges from reinforcement learning by redefining sequential decision pathways using a GPT structure enriched with causal self-attention. Moreover, the teacher network establishes a subtle state-reward mapping based on adjacent benefit differences, providing reliable rewards for student adaptively self-tuning with the vast unlabeled replay buffer data. Additionally, an innovative yet straightforward benefit reference value is proposed within the teacher network, adding to its effectiveness and simplicity. Leveraging a flexible replay buffer and embodied collaboration between teacher and student, the framework learns to see before detection with shallower features and shorter inference steps. Experiments highlight significant advantages of our algorithm over state-of-the-art detectors. The code is released at https://github.com/lydonShen/STF.

Introduction

Object detection is a hot topic common to various domains. In recent years, due to the increased training data (Lin et al. 2014; Shao et al. 2019; Kuznetsova et al. 2020) and GPU capability, the performance of object detectors has significantly improved. From CNN-based two-stage detectors (Girshick et al. 2014; Ren et al. 2017; Cai and Vasconcelos 2018) and one-stage detectors (Redmon et al. 2016; Lin et al. 2017; Tian et al. 2019), to the latest transformer-based detectors (Carion et al. 2020; Zhu et al. 2021a; Zhang et al. 2023), and large visual models (Liu et al. 2023), the accuracy and speed have been continuously enhanced. However, above methods focus on training model on the given data and ignore the importance of camera in learning how to see before detection. Specifically, taking a car counting task as an example. Based on conventional object detectors, precise counting of cars from any view-point is challenging due to impacts caused by factors like occlusion. This situation is illustrated by the dialogue with mini-gpt4 (Zhu et al. 2023) depicted in Figure 1. A straightforward counting task is difficult for conventional object detectors as they can not dynamically adjust the camera for the detector. This issue stimulated the exploration of active vision (Blake and Yuille 1993) and embodied intelligence (Gupta et al. 2021), which urges an agent to learn to see objects through interactions with the environment. Unfortunately, expensive interactive data annotation is required by active vision and embodied intelligence, and they exhibit limited adaptability when confronted with new scenarios.

To address the above limitations, a Student-Teacher object detection Framework (STF) is proposed in this paper. Inspired by human learning and offline learning (Agarwal, Schuurmans, and Norouzi 2020), STF treats object detection as a collaborative embodied learning process where an agent (the student) learns from an offline dataset of object detection transformations (exercises) using an object detector (the tool) and aims to perform well in test scenarios (exams). In Figure 1, the STF framework demonstrates the above learning process. To leverage large amounts of unlabeled data and improve the adaptability to new scenarios, a teacher network is introduced to evaluate the student's performance during inference. This enables student network self-tuning on a large amount of unlabeled trajectory data. In short, the contributions of this paper are as follows:

- An embodied object detection replay buffer is constructed, which contains 220,000 trajectory transformations from 10 different agent configurations in 76 diverse scenarios, aiming to facilitate research in embodied vision for outdoor scenarios.
- STF addresses the issue of insufficient detection performance in specific scenarios of traditional object detection, achieving better results with shallower image features and shorter inference steps.
- STF models embodied object detection as a sequence

*Corresponding author.

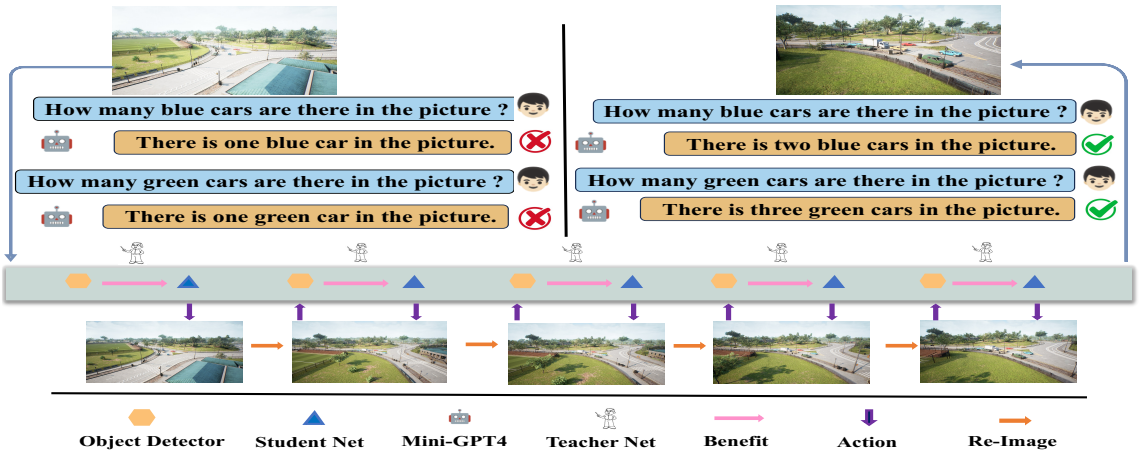

Figure 1: Idea illustration of STF. Similar to a Mini-GPT 4 dialogue, an agent interacts with an operator to count cars in an image. Initially, poor image quality leads to incorrect car counts. With the help of feedback provided by the answer, the camera adjusts its configuration driven by the detection performance and learns how to see. With collaborative learning, image quality improves gradually, and the answers become correct.

prediction task, introducing a method of self-tuning using unlabeled inference data.

- Different from previous methods, a novel benefit is used as the reward evaluation metric directly, which simplifies data acquisition and eliminates the need for complex reward function design.

Related Works

Embodied Vision

Embodied vision focuses on learning to perceive and act through interaction with the environment. Recently, a significant amount of research has been devoted to various embodied visual tasks, including embodied navigation (Wortsman et al. 2019; Li et al. 2020; Zhu et al. 2021b; Liang et al. 2022; Paul, Roy-Chowdhury, and Cherian 2022), vision question answering (Luo et al. 2023), embodied semantic segmentation (Nilsson et al. 2021), and embodied object detection(Yang et al. 2019; Xu et al. 2021).

To address the modal perception problem, (Yang et al. 2019) proposed the use of agents to strategically move and enhance their visual recognition capabilities. The agents relied on left-right movements to identify occluded objects. In contrast, our approach not only tackles occlusion but also considers factors such as illumination, viewpoint, and scale, making the action space more complex. (Kotar and Mottaghi 2022) proposed adaptive object detection in interactive environments, with agents navigating varied scenes and merging predictions from multiple frames. Our method aims to adjust the camera to obtain the most suitable configuration for detection. (Fang et al. 2019) proposed the Scene Memory Transformer and verified its effectiveness on long-horizon embodied visual navigation tasks.

Motivated by these methods, our work utilizes a GPT structure for sequence modeling and addresses complex challenges involving occlusion, illumination, view-point, and scale variations in outdoor scenes.

Actor-Critic Methods

Actor-critic methods (Mnih et al. 2016; Schulman et al. 2017, 2015) combine the advantages of both policy-based (actor) and value-based (critic) approaches. (Mnih et al. 2016) introduces an actor network for action proposals and a critic network to estimate the value function. Under the guidance of the advantage function, the updates to the policy for the actor are dynamically influenced by the action advantages in comparison to the average rewards. (Schulman et al. 2015) focuses on maintaining policy stability by imposing a trust region constraint during updates to prevent significant policy deviations from the original. (Schulman et al. 2017) simplifies the trust region constraint while ensuring policy improvement, employs multiple steps per iteration and employs a surrogate objective function to avert policy divergence.

Our STF is distinct from these methods. Unlike value-based approaches, STF employs the teacher network to assess the quality of sequential states. Moreover, the key contribution of the teacher network is its ability to facilitate student network self-tuning through the utilization of abundant unlabeled data.

Self-Adaptive and Semi-Supervised

Recently, great attention has been paid to enable agents to learn in an adaptive or weakly supervised manner.

(Wortsman et al. 2019) proposed a meta-reinforcement learning approach where an agent learns a self-supervised interaction loss to encourage effective navigation. (Kotar and Mottaghi 2022) proposed a similar idea of learning a loss function to integrate information from multiple frames for object detection. (Srinivasan et al. 2022; Wang et al. 2022) aimed to tackle catastrophic forgetting in neural networks. (Chu et al. 2022; Liu, Qi, and Fu 2021) proposed a self-training approach utilizing semi-supervised learning to enhance model performance through iterative training with

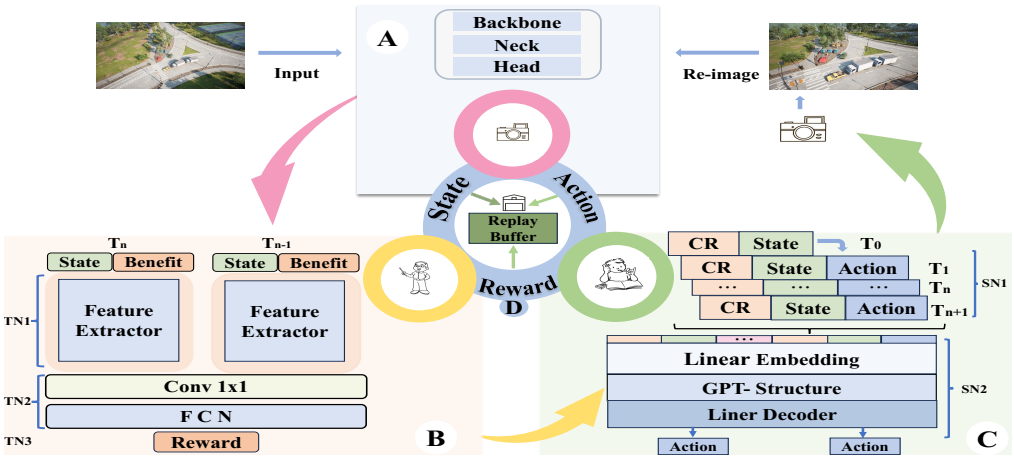

Figure 2: The teacher-student framework comprises an object detector(A), teacher network(B), student network(C), and replay buffer(D). The Teacher network is to produce a reliable reward for the student network. Student network aims to predict camera action, whose inputs are triplets from the replay buffer.

pseudo-labeling.

Our work differs significantly from these methods. Instead of training a loss function for self-adaptive learning, we directly train a teacher network to evaluate rewards for the agent's states. While relate to the above tasks, our motivation is distinct. We focus on the continuous fine-tuning of the student network using unlabeled inference data to enhance detection performance and adapt it to new scenarios.

Methodology

In this paper, collaborative embodied learning means embodied learning of the camera for configuration adjustment and collaborative learning between the camera and detector, which is accomplished by the following STF framework.

Student-Teacher Framework

As illustrated by Fig. 2, STF consists of the following four components, object detector(A), student network(B), teacher network(C) and replay buffer(D). The object detector is responsible for extracting features and detecting objects from input images. The teacher network is to evaluate the action performance and to help the student network predict the action at the next time step. The role of replay buffer is to store inference trajectory information to fine-tune the student network.

Formally, the proposed framework is formulated as follows:

$$STF(I_t) = f(\Phi(I_t), \Psi(S_t, S_{t-1}), \varphi(S_t, \hat{R}_t, A_{t-1})) \ , \ (1)$$

Where the object detector is denoted by $\Phi(x)$, the student network by $\varphi(x)$, the teacher network by $\Psi(x)$, the image under the imaging configuration at time T by the I_t, the current state S_t, the reward of the state by R_t, the distance from the current reward to the target reward is defined by $\hat{R}_t$, the benefit of the state by B_t, and the current action by A_t.

As illustrated in Fig. 2, STF works as follows. The image obtained at the time step t, I_t, is fed into the object detector $\Phi(x)$, which provides reference benefit B_t and state (backbone feature) S_t for training the teacher network $\Psi(x)$. Based on the reward provided by the teacher network R_t and the current state S_t and the previous action A_{t-1}, the student network $\Psi(x)$ predicts the action A_t for adjusting the camera configuration. The new image can be acquired by the new configuration, and the above procedure repeated until all objects are detected correctly.

Through the implementation of STF, object detection and camera adjustment are interactive adaptively driven by the object performance, and the advantages of collaborative embodied learning are being further elaborated. Among the four components mentioned above, the object detector is not the primary focus of this paper, and traditional detectors can be seamlessly integrated into the SFS without any issues. The primary attention of this paper is directed towards the other three components, which will be elaborated upon in a sequential manner below.

Student Network

The student network is aimed to predict action for camera adjustment, which is usually accomplished by conventional reinforcement learning methods (van Hasselt, Guez, and Silver 2016; Schulman et al. 2017, 2015; Xu et al. 2021) based on value functions or policy gradients.

Considering the inefficiency of reinforcement learning methods in utilizing information from adjacent time steps, inspired by the successful Transformer architecture in game tasks (Chen et al. 2021), a student network is proposed, which reformulates the decision-making problem as a sequential prediction task. As illustrated by the part B of Fig.2, the student network is essentially a causally masked transformer decoder, which maps past states, rewards and actions to the action that is closest to the defined final cumulative reward.

$$Action = \varphi(S, \hat{R}, A) \ , \ (2)$$

The input of the student network is the trajectory, and the output is the action A_t. The student network contains two

modules, trajectory information embedding(SN1) and action prediction(SN2).

In SN1, trajectory information is fed into different embedding layers, i.e., linear embeddings and time-step embedding. Note that each trajectory item is a triplet obtained at the time step t about the state S_t, the action A_t and the cumulative reward $\hat{R}_t$.

$$\hat{R}_t = \sum_{t'=t}^{T} R_{t'} \ , \ (3)$$

In SN2, various types of encoded information are sent to GPT (Radford et al. 2018), which utilizes an autoregressive model to predict future action tokens. By taking advantage of causally masked transformer, self-attention contained in the trajectory feature itself and cross-attention between trajectory features of different time steps are captured. Finally, the state enriched with decision attention is transmitted to a linear decoder layer for classification, and the optimal action is thus yielded.

In training, the input trajectory is from the replay buffer, and the reference action is from the inference procedure. Cross-entropy loss is used for the loss function, and supervised learning is chosen to train the student network.

Compared to traditional reinforcement learning methods, the transformer architecture is more promising for our task. Firstly, its sequence-to-sequence prediction and attention mechanism are important for the student network to capture the temporal context, which is the key factor in adaptively adjust the camera. Secondly, causally transformer helps the student network produces the action in a more stable fashion, avoiding the problem of unstable prediction in reinforcement learning methods. Thirdly, transformer established long-term decision-making attention makes the decision-making more robust.

Teacher Network

For the student network, reliable reward is the key factor of action selection. In this paper, the teacher network is aimed to offer a reliable reward to guide the student network. As illustrated in part C of Fig. 2, the teacher network consists of the following three blocks, state feature extraction(TN1), state difference representation(TN2) and reward prediction(TN3).

The reward is evaluated based on states at two consecutive time steps, and Siamese architecture (Bromley et al. 1993) is the best choice for this task since it is promising in capturing the paired state information. Taking into account the significance of features derived from the object detector backbone as essential state indicators, these features are transmitted to a Siamese network with shared parameters between state T_n and state T_{n-1}. Considering the different focuses of the detection task and classification task on feature learning, the following two strategies are explored:

The first one is to directly use the compressed features extracted by the object detector for the reward learning. The second one is to add a classification head after the object detector's backbone, which is initialized with the object detector's backbone and fine-tuned on the training dataset. By comparison, we observed that the second method (the TN2 block in Fig. 2) is more effective. Therefore, the second method is chosen in the following experiments. The TN3 block is to predict the reward.

To train the teacher network, reference reward is required. In this paper, a novel detection-based benefit is proposed for the reference reward. Prior works (Ren et al. 2018; Pirinen and Sminchisescu 2018; Xu et al. 2021) utilized IOU or F1 score to calculate rewards between two consecutive time steps. These metrics capture the confidence about label similarity and position regression, however, the valuable original information contained in the object detection loss contains indirect information of IOU or F1 score. For this reason, the detection loss difference at the time step T_n and T_{n-1} is directly used as the reference reward. Considering the fact that the reward to be predicted is a continuous real number instead of simple binary label, this modification is a simple yet effective. This topic will be discussed in detail in the ablation experiments.

Replay Buffer

During training, a replay buffer is established to train the student network, which is collected from diverse configurations of reinforcement learning agents' experiences. For the student network, it constitutes a collection of exercises with varying levels of quality. The student network learns from an extensive pool of trajectory information, with the aspiration to surpass even its most adept predecessors.

After the end of the training, the replay buffer is used for fine-tuning the student network, as illustrated in part D of Fig. 2. The student-teacher framework, after the integration of the replay buffer, forms a closed loop of training-inference-training. The teacher network generates $\hat{R}_t$, the object detector produces S_t, and the student network generates A_t, forming a triplet that is stored in the replay buffer (Eq. 4).

$$\iota = (\hat{R}_1, S_1, A_1, ..., \hat{R}_T, S_T, A_T) \ , \ (4)$$

The replay buffer is organized by a fixed-size queue, where old inference data are replaced by new ones.

Implemention Details

During the training phase, the object detector is trained using various imaging configurations on the training set. Based on the backbone features and detection loss difference at adjacent time steps, the teacher network is trained. The student network is pre-trained by selecting training samples from the replay buffer.

During the inference phase, an image is processed by the object detector, whose backbone features and detection results are saved as states in the replay buffer. The teacher network evaluates the student network about the current state and the previous actions, based on the reward provided by the teacher network, the student network predicts the optimal action at the next time step. Upon completing the population of the inference dataset in the replay buffer, the student network proceeds with self-tuning.

Experiments

Datasets Description

Two following datasets (Xu et al. 2021) are used for experiments. The Small Airport (SA) scenario involves aircraft detection with 12 categories, including civil aircraft (Boeing787, Boeing777, Boeing747, L-1011) and fighters (F-16, J-10, Su-33UB, E-2c, YF-22, F-14, J-5, F/A-18E/F). The dataset comprises 4500 images, divided into a training set of 64 scenes with 3200 images and a test set of 26 scenes with 1300 images.The Virtual Park (VP) scenario has 5 car types (Box Truck, SUV, Pickup, Hatchback, Sports Car) in 20 scenes. The dataset holds 5000 images: 3000 training and 2000 testing from 12 and 8 scenes, respectively. By adjusting longitude, latitude, distance and exposure compensation, the camera learns to determine the best imaging setup. Nine actions are used for camera configuration: up, down, left, right, forward, backward, brighten, darken, and termination.

A new replay buffer dataset is constructed for outdoor scenarios using reinforcement learning agent based on the above datasets. It contains 220,000 trajectory transformations from ten different agent configurations in 76 diverse scenarios. As illustrated in the ablation study, the replay buffer dataset is very important for the student network and the final performance.

Experiment Setting

AP is a widely used metric for evaluating detection performance. In this paper, we use average precisions at different IoU thresholds (AP at IoU=0.50:0.05:0.95) to assess the performance variations during camera configuration adjustments in two datasets. For a fair comparison, the same inference steps as in (Xu et al. 2021) is adopted. The STF framework used 3200 and 3000 initial scenarios in SA and VP, respectively, to train the student network. Additionally, 1000 and 600 inference scenarios data are used for fine-tuning.

	PM	Performance		PM	Performance
SA	DINO	80.9	VP	DINO	71.1
	FCOS	72.4		FCOS	57.9
	FSAF	74.3		FSAF	64.8
	ours	88.2		ours	82.5

Table 1: Performance comparison of STF and passive object detectors.

Comparisons with State-of-the-art Methods

To validate the novelty of the proposed framework, three state-of-the-art passive detectors(DINO (Zhang et al. 2023), FCOS (Tian et al. 2019), and FSAF (Zhu, He, and Savvides 2019)) and six state-of-the-art active detectors(FCOS+PPO, FCOS+DQN, FCOS+DDQN, FSAF+PPO, FSAF+DQN, FSAF+DDQN) are used for comparison. Performances are listed in Table 1 and Table 2.

Performance Comparison. In Table 1, we compare STF with passive object detectors using DINO, FCOS, and FSAF on two datasets. STF outperforms FCOS by 15.8% (SA) and 24.6% (VP), and it surpasses FSAF by 13.9% and 17.7%, and DINO by 7.3% and 11.4%. This demonstrates the significant advantage of embodied learning in handling challenging scenarios like occlusion and uneven lighting. On the one side, FCOS and FSAF are traditional deep-learning-based passive detectors, they aim to capture the relationship between complex images and object signatures by the direct mapping. In fact, the direct mapping is very difficult even based on massive training data. In other words, they fail to learning how to learn. In contrast, DINO achieved significant improvements due to the transformer structure. As we know,the transformer-based detector is superior to traditional CNN-based detector due to its self-attention mechanism. Moreover, with the help of powerful semantic information taken and vast amount of training data, DINO is very promising even for open-vocabulary object detection. In this context, DINO is capable of learning how to learn. For instance, in VP, it improves 57.9% of FCOS to 71.1%. However, passive detectors ignore the importance of embodied learning, this is inadequate for challenging scenes, i.e., they are limited in learning how to see.

In Table 2, we evaluate reinforcement learning methods (Xu et al. 2021) across different step lengths and detectors. Our approach, with FCOS, surpasses DDQN(value-based), PPO(policy-based) and TRPO(policy-based) by 4.5%, 7.9%, 5.2% (SA) and 4.6%, 8%, 8.7% (VP) respectively. Even when using stronger detectors, our method outperforms the best performances by 3.5% (SA) and 4.3% (VP). The performance differences demonstrate the novelty and effectiveness of the STF framework.

Specifically, traditional active detectors outperform passive detectors since they aim to predict action for camera adjustment dynamically. However, for reinforcement learning methods, due to the implicit relationship between camera action and visual features, the network as well as reward function are difficult to design, whether for value-based methods or for policy-based methods. For instance, in SA, at the 5th step, FCOS+TRPO achieved the performance 82.6%, however, it degraded to 82.4% at the 6th step. More generally, for most reinforcement-learning-based active detectors, the camera is adjusted slowly or unsteadily. For another example, in VP, for FSAF+DDQN, FSAF+PPO and FSAF+TRPO, the performances are not improved from the 7th step to the 8th step.

The above instances illustrate the limitation of reinforcement learning in capturing decision attention. In contrast, STF improves the detection performance more stablely, and there is no performance halt or degradation between adjacent steps. In detail, in Table 2, by the 5th step, STF reached 85.4% and 81.3% respectively on the two datasets, and it outperformed the final performance of reinforcement learning-based methods(84.7%,78.2%). In SA, STF achieved 84.1%performance in just 3 steps, while it is higher than the final performance of FCOS+DDQN, 83.7%.

Similar conclusions could be drawn from Fig. 3, where performances variations with increasing inference steps are shown. STF often stops early when initial settings are favorable(C1 of Fig. 3), preventing overshooting. STF maintains stable inference and identifies good imaging configurations quickly(C2,C3 of Fig. 3). In contrast, performances by

Active Methods		Step Performance									
		0	1	2	3	4	5	6	7	8	9
SA	FCOS+DDQN	72.4	76.9	79.7	81.2	82.7	83.3	83.5	83.5	83.7	83.7
	FCOS+PPO	72.4	76.7	79.5	80.9	82.7	83.1	83.2	83.6	83.6	83.6
	FCOS+TRPO	72.4	76.0	78.8	80.3	81.7	82.6	82.4	82.6	82.9	83.0
	FSAF+DDQN	74.3	78.2	80.8	82.5	83.4	84.5	84.5	84.6	84.6	84.7
	FSAF+PPO	74.3	77.9	80.9	82.2	83.3	84.4	84.4	84.6	84.7	84.7
	FSAF+TRPO	74.3	77.3	80.1	81.5	82.8	83.5	83.7	83.8	83.9	83.8
	(**ours**)	**72.4**	**79.1**	**83.3**	**84.1**	**84.9**	**85.4**	**86.9**	**87.1**	**87.7**	**88.2**
VP	FCOS+DDQN	57.9	62.9	66.6	70.2	72.0	73.1	73.8	74.4	74.5	74.6
	FCOS+PPO	57.9	62.6	66.4	70.0	71.9	73.0	73.9	74.1	74.4	74.5
	FCOS+TRPO	57.9	62.0	65.6	69.1	71.3	72.2	73.1	73.3	73.7	73.8
	FSAF+DDQN	64.8	69.0	72.4	75.1	76.6	77.2	77.4	78.0	78.0	78.2
	FSAF+PPO	64.8	69.0	72.1	74.9	76.3	76.9	77.3	77.7	77.7	78.1
	FSAF+TRPO	64.8	68.2	71.1	74.2	75.4	76.0	76.6	77.2	77.2	77.4
	(**ours**)	**57.9**	**69.7**	**74.2**	**77.6**	**80.8**	**81.3**	**81.9**	**82.2**	**82.4**	**82.5**

Table 2: Performance comparison of STF and active object detectors.

reinforcement learning-based methods are degraded unanimously.

The above comparisons demonstrate the advantages of the proposed framework. Without the reliable reward provided by the teacher network and the reliable action predicted by the student network, it is impossible for STF to improve the performance quickly and stably. In consequence, collaborative embodied learning between the student network and the teacher network is effective. Of course, the role of replay buffer could not be ignored, since it is the bridge that links the student network and the teacher network.

Ablation Study

To understand how the proposed framework works, the following ablation experiments are conducted. The first ablation experiment is to investigate the student network with respect to action selection. In Table 3, the "Greedy" approach represents the use of a greedy algorithm for action selection, where each step chooses the action that maximizes the immediate benefit. IOU and Loss denote IOU-based benefit and detection-loss-based benefit, respectively. Fine-tuning means fine-tuning the student network using the inference dataset.

No.	Greedy	Loss	IOU	Fine-tuning	SA	VP
1	✓				78.6	64.4
2			✓		86.9	79.3
3		✓			87.1	80.1
4			✓	✓	87.8	81.9
5		✓		✓	88.2	82.5

Table 3: Performance comparison on the first ablation experiment.

The first experiment is to validate our innovation. From Table 3, it is observed that the pure usage of the greedy search method leads to poor performances on the two datasets, 78.6% and 64.4%. However, by substituting the greedy search with STF, the object detection performances are improved to 86.9% and 79.3%, irrespective of the reference benefit used. However, by the loss-based approach, performances are improved to 87.1% on SA and 80.1% on VP. In comparison, by the new benefit-based method, performance gains are 0.2% and 0.8%, respectively. This demonstrates the effectiveness of the proposed new benefit.

By freezing the weights of the action coding and cumulative reward coding layers, we fine-tune the student network using the inference dataset consisting of about 12,000 trace transitions. From Table 3, it is evident that two different benefit evaluation methods achieved performance improvements of 1.1% and 0.9% respectively on SA. Similarly, on VP, performance improvements are 2.4% and 2.6%. Fine-tuning enables the student network to leverage the substantial amount of unlabeled inference data more efficiently. Note that STF outperforms the greedy search algorithm by 9.6% and 18.1%, respectively. This ablation experiment demonstrates the effectiveness of the detection-loss-based benefit and the importance of fine-tuning the student network.

The second experiment is to validate STF on downstream task. For this purpose, 500 images are chosen from VP and SA, and STF is used to accurately count objects of each category. As illustrated in Table 4, detection accuracies of passive detector FCOS are 71.6% and 85.2% on VP and SA, respectively. Reinforcement learning-based active detection methods achieved 90.4% and 93.6% . Remarkably, accuracies by the STF framework are improved to 95.6% and 97.2%. In Fig. 4, results by reinforcement learning-based methods are shown. There exist missed and over-counted objects(left of Fig. 4). In contrast, the STF framework consistently performs best.

	Passive Detector	DDQN	STF
VP	71.6	90.4	95.6
SA	85.2	93.6	97.2

Table 4: Performance comparison on counting tasks

The third experiment is to validate whether the longer contextual information performs better. Table 5 lists per-

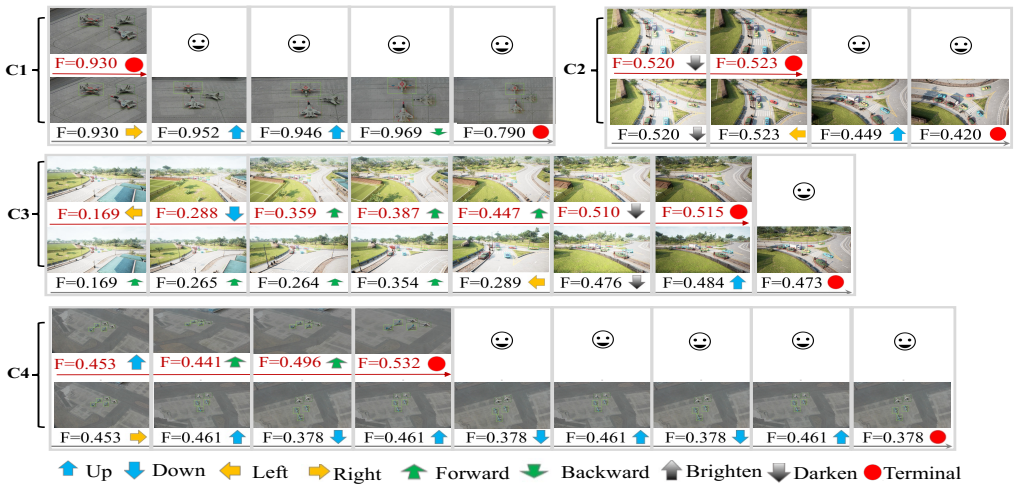

Figure 3: Result comparison on camera adjustment: our method in first, third and fifth rows, RL method in second, fourth and sixth rows.

Figure 4: Result comparison on counting task. (a): Result by DDQN in case 1, one car is missed. (b): Result by our method in case 1, cars are counted correctly. (c): Result by passive detector in case 2, one object is detected as car wrongly. (b): Result by our method in case 2, cars are counted correctly.

formances varied with the context length, where AB indicates the size of the attention blocksize. From Table 5, it can be inferred that an attention block size of 6 achieved the best performance, striking a balance between local and long-range context information.

Table 5 suggests that too small attention block prevents the model from capturing long-range dependencies. Too small attention block resulted in 1.1% and 2.8% performance degradations on both datasets. On the other hand, employing an excessively large attention block size enables the model to incorporate longer-range contextual information; however, this might lead to the model overfitting to intricate details and noise present in the training dataset. And similar performance declines are obtained on the two datasets, 2.8% and 3.4%, respectively.

The fourth ablation experiment is to validate whether dense feature representation is necessary. In(Xu et al. 2021) method, all features after FPN are used as the input of the policy network.In Table 5, P represents the feature extracted by FPN.The size of P3 is(100, 180), P4 is (50,90), P5 is (25,45), P6 is (13,23). From Table 5, it can be observed that dense features of P3 result in performance degradation by 3.3% and 5.1% respectively than the best performance. Conversely, the smaller-size P6 yields performance improvements of 2.5% and 2.3% over P3.

AB Size	SA	VP	Feature Map	SA	VP
3	87.1	79.7	P3	84.9	77.1
6	88.2	82.5	P4	87.8	81.3
9	87.4	80.9	P5	88.2	82.5
12	85.4	79.1	P6	87.4	79.4

Table 5: Performance comparison on context length and feature map size.

The above comparison revealed that for the student network, dense features don't necessarily lead to better performances. In other words, for decision-making, the key lies in selecting appropriate feature representations.

Conclusion

In this paper, a novel student-teacher framework is proposed for collaborative embodied learning between object detection and camera adjustment. By leveraging shallower image features and shorter inference steps, STF outperforms previous methods. Specifically, the novelties lie in the reliable reward provided by the teacher network, the credible action predicted by the casually masked transformer-based student network, and the effective fine-tuning on the replay buffer. The detection-loss-based reward evaluation simplifies the teacher network and enhances the confidence degree. The casually masked transformer is beneficial for the student network to capture long-term decision attention, and fine-tuning on unlabeled inference replay buffer enables the student network to learn how to see. In consequence, STF offers a new solution to embodied vision.

Acknowledgments

This research is supported by National Natural Science Foundation of China (Grant No. 62071466).

References

Agarwal, R.; Schuurmans, D.; and Norouzi, M. 2020. An Optimistic Perspective on Offline Reinforcement Learning. In *ICML*, 104–114.

Blake, A.; and Yuille, A. 1993. *Active Vision*. *Robotica*, 11(5): 487.

Bromley, J.; Guyon, I.; LeCun, Y.; Säckinger, E.; and Shah, R. 1993. Signature Verification Using a Siamese Time Delay Neural Network. In *NeurIPS*, 737–744.

Cai, Z.; and Vasconcelos, N. 2018. Cascade R-CNN: Delving Into High Quality Object Detection. In *CVPR*, 6154–6162.

Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; and Zagoruyko, S. 2020. End-to-End Object Detection with Transformers. In *ECCV*, 213–229.

Chen, L.; Lu, K.; Rajeswaran, A.; Lee, K.; Grover, A.; Laskin, M.; Abbeel, P.; Srinivas, A.; and Mordatch, I. 2021. Decision Transformer: Reinforcement Learning via Sequence Modeling. In *NeurIPS*, 15084–15097.

Chu, R.; Ye, X.; Liu, Z.; Tan, X.; Qi, X.; Fu, C.; and Jia, J. 2022. TWIST: Two-Way Inter-label Self-Training for Semi-supervised 3D Instance Segmentation. In *CVPR*, 1090–1099.

Fang, K.; Toshev, A.; Fei-Fei, L.; and Savarese, S. 2019. Scene Memory Transformer for Embodied Agents in Long-Horizon Tasks. In *CVPR*, 538–547.

Girshick, R. B.; Donahue, J.; Darrell, T.; and Malik, J. 2014. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In *CVPR*, 580–587.

Gupta, A.; Savarese, S.; Ganguli, S.; and Fei-Fei, L. 2021. Embodied intelligence via learning and evolution. *Nature communications*, 12(1): 5721.

Kotar, K.; and Mottaghi, R. 2022. Interactron: Embodied Adaptive Object Detection. In *CVPR*, 14840–14849.

Kuznetsova, A.; Rom, H.; Alldrin, N.; Uijlings, J. R. R.; Krasin, I.; Pont-Tuset, J.; Kamali, S.; Popov, S.; Malloci, M.; Kolesnikov, A.; Duerig, T.; and Ferrari, V. 2020. The Open Images Dataset V4. *Int. J. Comput. Vis.*, 128(7): 1956–1981.

Li, J.; Wang, X.; Tang, S.; Shi, H.; Wu, F.; Zhuang, Y.; and Wang, W. Y. 2020. Unsupervised Reinforcement Learning of Transferable Meta-Skills for Embodied Navigation. In *CVPR*, 12120–12129.

Liang, X.; Zhu, F.; Zhu, Y.; Lin, B.; Wang, B.; and Liang, X. 2022. Contrastive Instruction-Trajectory Learning for Vision-Language Navigation. In *AAAI*, 1592–1600.

Lin, T.; Goyal, P.; Girshick, R. B.; He, K.; and Dollár, P. 2017. Focal Loss for Dense Object Detection. In *ICCV*, 2999–3007.

Lin, T.; Maire, M.; Belongie, S. J.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft COCO: Common Objects in Context. In *ECCV*, 740–755.

Liu, S.; Zeng, Z.; Ren, T.; Li, F.; Zhang, H.; Yang, J.; Li, C.; Yang, J.; Su, H.; Zhu, J.; and Zhang, L. 2023. Grounding DINO: Marrying DINO with Grounded Pre-Training for Open-Set Object Detection. *In ArXiv*, abs/2303.05499.

Liu, Z.; Qi, X.; and Fu, C. 2021. One Thing One Click: A Self-Training Approach for Weakly Supervised 3D Semantic Segmentation. In *CVPR*, 1726–1736.

Luo, H.; Lin, G.; Yao, Y.; Liu, F.; Liu, Z.; and Tang, Z. 2023. Depth and Video Segmentation Based Visual Attention for Embodied Question Answering. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(6): 6807–6819.

Mnih, V.; Badia, A. P.; Mirza, M.; Graves, A.; Lillicrap, T. P.; Harley, T.; Silver, D.; and Kavukcuoglu, K. 2016. Asynchronous Methods for Deep Reinforcement Learning. In *ICML*, 1928–1937.

Nilsson, D.; Pirinen, A.; Gärtner, E.; and Sminchisescu, C. 2021. Embodied Visual Active Learning for Semantic Segmentation. In *AAAI*, 2373–2383.

Paul, S.; Roy-Chowdhury, A.; and Cherian, A. 2022. AVLEN: Audio-Visual-Language Embodied Navigation in 3D Environments. In *NeurIPS*.

Pirinen, A.; and Sminchisescu, C. 2018. Deep Reinforcement Learning of Region Proposal Networks for Object Detection. In *CVPR*, 6945–6954.

Radford, A.; Narasimhan, K.; Salimans, T.; Sutskever, I.; et al. 2018. Improving language understanding by generative pre-training.

Redmon, J.; Divvala, S. K.; Girshick, R. B.; and Farhadi, A. 2016. You Only Look Once: Unified, Real-Time Object Detection. In *CVPR*, 779–788.

Ren, L.; Lu, J.; Wang, Z.; Tian, Q.; and Zhou, J. 2018. Collaborative Deep Reinforcement Learning for Multi-object Tracking. In *ECCV*, 605–621.

Ren, S.; He, K.; Girshick, R. B.; and Sun, J. 2017. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(6): 1137–1149.

Schulman, J.; Levine, S.; Abbeel, P.; Jordan, M. I.; and Moritz, P. 2015. Trust Region Policy Optimization. In *ICML*, 1889–1897.

Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; and Klimov, O. 2017. Proximal Policy Optimization Algorithms. *ArXiv*, abs/1707.06347.

Shao, S.; Li, Z.; Zhang, T.; Peng, C.; Yu, G.; Zhang, X.; Li, J.; and Sun, J. 2019. Objects365: A Large-Scale, High-Quality Dataset for Object Detection. In *ICCV*, 8429–8438.

Srinivasan, T.; Chang, T.; Alva, L. L. P.; Chochlakis, G.; Rostami, M.; and Thomason, J. 2022. CLiMB: A Continual Learning Benchmark for Vision-and-Language Tasks. In *NeurIPS*.

Tian, Z.; Shen, C.; Chen, H.; and He, T. 2019. FCOS: Fully Convolutional One-Stage Object Detection. In *ICCV*, 9626–9635.

van Hasselt, H.; Guez, A.; and Silver, D. 2016. Deep Reinforcement Learning with Double Q-Learning. In *AAAI*, 2094–2100.

Wang, Z.; Liu, L.; Duan, Y.; Kong, Y.; and Tao, D. 2022. Continual Learning with Lifelong Vision Transformer. In *CVPR*, 171–181.

Wortsman, M.; Ehsani, K.; Rastegari, M.; Farhadi, A.; and Mottaghi, R. 2019. Learning to Learn How to Learn: Self-Adaptive Visual Navigation Using Meta-Learning. In *CVPR*, 6750–6759.

Xu, N.; Huo, C.; Zhang, X.; Cao, Y.; Meng, G.; and Pan, C. 2021. Dynamic camera configuration learning for high-confidence active object detection. *Neurocomputing*, 466: 113–127.

Yang, J.; Ren, Z.; Xu, M.; Chen, X.; Crandall, D. J.; Parikh, D.; and Batra, D. 2019. Embodied Amodal Recognition: Learning to Move to Perceive Objects. In *ICCV*, 2040–2050.

Zhang, H.; Li, F.; Liu, S.; Zhang, L.; Su, H.; Zhu, J.; Ni, L. M.; and Shum, H. 2023. DINO: DETR with Improved DeNoising Anchor Boxes for End-to-End Object Detection. In *ICLR*.

Zhu, C.; He, Y.; and Savvides, M. 2019. Feature Selective Anchor-Free Module for Single-Shot Object Detection. In *CVPR*, 840–849.

Zhu, D.; Chen, J.; Shen, X.; Li, X.; and Elhoseiny, M. 2023. MiniGPT-4: Enhancing Vision-Language Understanding with Advanced Large Language Models. *In ArXiv*, abs/2304.10592.

Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; and Dai, J. 2021a. Deformable DETR: Deformable Transformers for End-to-End Object Detection. In *ICLR*.

Zhu, Y.; Weng, Y.; Zhu, F.; Liang, X.; Ye, Q.; Lu, Y.; and Jiao, J. 2021b. Self-Motivated Communication Agent for Real-World Vision-Dialog Navigation. In *ICCV*, 1574–1583.