2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) | 978-1-6654-9346-8/23/$31.00 ©2023 IEEE | DOI: 10.1109/WACV56688.2023.00459

# Avoiding Lingering in Learning Active Recognition by Adversarial Disturbance

Lei Fan and Ying Wu
Northwestern University
2145 Sheridan Road, Evanston, IL 60208
leifan@u.northwestern.edu, yingwu@northwestern.edu

## Abstract

This paper considers the active recognition scenario, where the agent is empowered to intelligently acquire observations for better recognition. The agents usually compose two modules, i.e., the policy and the recognizer, to select actions and predict the category. While using ground-truth class labels to supervise the recognizer, the policy is typically updated with rewards determined by the current in-training recognizer, like whether achieving correct predictions. However, this joint learning process could lead to unintended solutions, like a collapsed policy that only visits views that the recognizer is already sufficiently trained to obtain rewards, which harms the generalization ability. We call this phenomenon lingering to depict the agent being reluctant to explore challenging views during training. Existing approaches to tackle the exploration-exploitation trade-off could be ineffective as they usually assume reliable feedback during exploration to update the estimate of rarely-visited states. This assumption is invalid here as the reward from the recognizer could be insufficiently trained.

To this end, our approach integrates another adversarial policy to constantly disturb the recognition agent during training, forming a competing game to promote active explorations and avoid lingering. The reinforced adversary, rewarded when the recognition fails, contests the recognition agent by turning the camera to challenging observations. Extensive experiments across two datasets validate the effectiveness of the proposed approach regarding its recognition performances, learning efficiencies, and especially robustness in managing environmental noises.

## 1. Introduction

Passive visual recognition, relying on human-taken images or videos, has achieved dramatic successes in recent decades. On the contrary, in robotic scenarios, active recognition systems are expected to involve intelligent control strategies in the recognition process. The primary motivation behind active recognition is to circumvent undesired viewing conditions while obtaining unambiguous and discriminative information.

Several learning-based active recognition methods [4, 23, 18, 17, 6, 25, 7, 33] have been proposed over the years. Commonly, these approaches recurrently deliver two outputs, i.e., the action to execute from the policy and the category probabilities from the recognizer. As two modules collaborate, multiple possible combinations exist to achieve the same final class prediction. An intuitive explanation is that various camera trajectories exist to classify the same object if different recognizers are proficient with different views. However, we observe that the policy of active recognition agents could collapse to a repetitious mode during training because of incorrect rewards from the recognizer. Reversely, the collapse of policy further exacerbates the overfitting of the recognizer. This phenomenon is named lingering in this paper (shown in Figure 1), consisting of the unwillingness to explore and the meaningless roll-out experience collecting. In Figure 2, we visualize the lingering by showing the view-specific visiting frequencies during training and their corresponding testing accuracy. We observe that the lingering jeopardizes the generalization ability of agents by overfitting to only limited views.

As escaping lingering is imperative to the realistic deployment of agents, the problem is still under-explored in the active recognition literature. Among several avail-able remedies, approaches to address the exploration-exploitation trade-off [33, 3, 13] are related while not suitable for active recognition scenarios. Considering classical methods ($\epsilon$-greedy, Thompson sampling, etc.) to tackle the trade-off, these methods assume reliable rewards during exploration to update the estimate of rarely-visited states. Unfortunately, the assumption does not hold for training the active recognition agent because the reward depends on the recognizer's current performance. In other words, the feedback could be negative not because this view is not informative but because the training of the recognizer is yet inadequate.

Staged training and using pre-trained recognizers [43, 13, 12, 46] serve as another strategy. Human interventions,

2642-9381/23/$31.00 ©2023 IEEE
DOI 10.1109/WACV56688.2023.00459

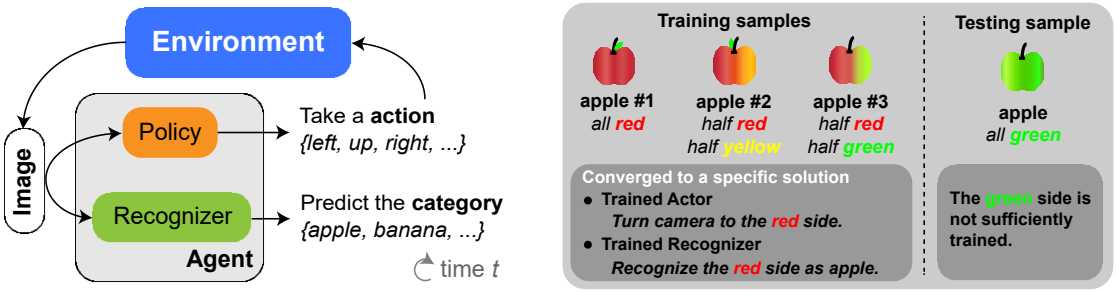
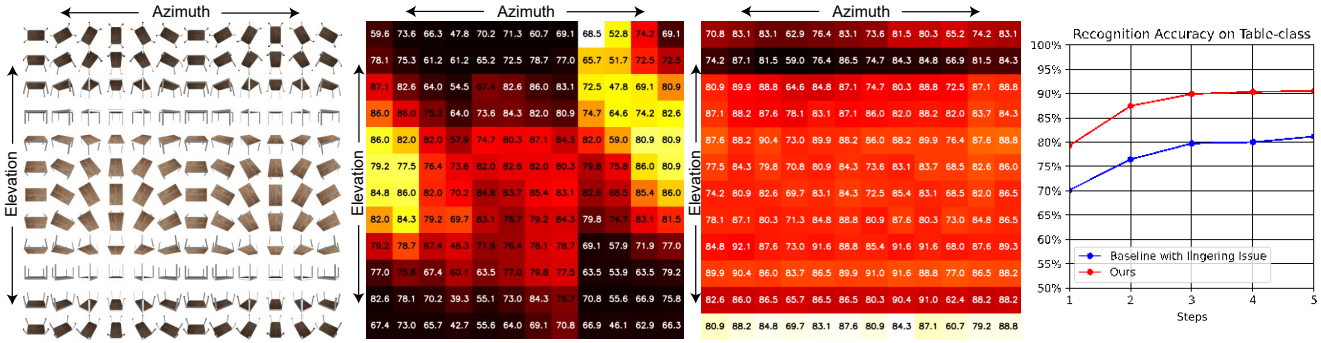



Figure 1. A conceptual overview of the lingering issue in active recognition systems. Embodied agents could interact with the environment by obtaining observations and making movements (in the left). The two modules, i.e., the policy and the recognizer, could converge to an undesired solution because the recognizer only provides rewards to views it could already correctly classify, leading to a collapsed policy. We call this phenomenon as lingering. The right gives a straightforward example of drawbacks if not resolving the lingering issue.




A sample from **Table**-class

View visiting heatmaps (*train*) and accuracies (*testing*) with ***lingering issue***.

Our view visiting heatmaps (*train*) and accuracies (*testing*) without ***lingering***.

The comparison on the final active recognition performance.

Figure 2. A comparison between ours and the baseline on their view visiting frequencies during training and their corresponding accuracy during testing. The ratios between the highest and the lowest visiting frequencies in each heatmap are 193.80 and 2.31, respectively.

like deficient offline data collecting, become inevitable. Overall, learning from interactions [19, 14] is considered the same noteworthy as actively performing in the environment, especially for active recognition agents.

In this paper, we first explain the inherent multiple-solution nature of active recognition by formulating it into a multiplication form. Then, by modeling its iterative train-ing procedure, we explain the reason behind lingering, i.e., converging to a specific solution lacking generalization ability. To address lingering, we propose to disturb the active recognition policy with an adversary during training. The adversarial policy is rewarded as providing action disturbances leading to recognition failures. Therefore, the recognition and adversarial policies establish a zero-sum competing game. The final optimum of the two policies is achieved by iteratively solving the min-max equilibrium.

To summarize, the contribution and insights are about examining the lingering issue of active recognition agents and addressing it by involving disturbances from the adversarial policy. We validate the proposed approach in both active object [10] and scene recognition scenarios [42]. The advantages of our method are demonstrated from three aspects. (1) With the presence of an adversary, the proposed method achieves better active recognition performance regarding avoiding the lingering issue. (2) The robustness of our method is shown by conducting experiments on introducing additional environmental noises, including view occlusions and movement failures. (3) Compared to uniform and Gaussian-distributed exploratory actions, the proposed adversarial policy could more effectively mine challenging views to improve overall performance.

## 2. Related Work

Active vision. Active vision, as a long-standing field pioneered by [2, 1, 8, 38], has been explored in several branches, like recognition [5, 6], exploration [33, 14, 24, 34, 12], localization [3], and navigation [15, 13, 16, 11, 41]. The common motivation is to allow the agent to observe from its own intentions, i.e., letting the agent actively select observations to accomplish different tasks.

Specifically, the motivations for active recognition are generally elaborated from three directions: reducing ambiguities [23, 34, 24], avoiding undesired viewing conditions, and maximizing information gains [36, 6, 4]. These motivations are inherently connected under the ultimate objective of better recognition performance. Based on their implementations, prior works could be mainly identified into two groups based on whether they represent the Markov Deci-

sion Process (MDP) with a reinforcement learning model. [5, 6] propose an active object hypothesis validation method balancing the movement cost and the chance to correct identification. In [36], a saliency module indicating potential information profits is inserted into the observation module of partially observable-MDP [26]. These methods focus on different ways of defining view-specific benefits and then planning trajectories.

In this paper, we compare the proposed method mostly with other reinforcement-learning approaches. In [23, 34], the author proposes an active recognition agent that is end-to-end trainable with reinforced policy descent. Three modules targeting view evidence aggregation, classification, and next-view prediction cooperate to guide action selection. [18] aims at placing active recognition in a more challenging but practical scenario, which considers continuously emerging novel categories. However, most existing active recognition works directly train their agents from scratch and deliver inadequate attention to the lingering issue.

Among other general approaches related to alleviating lingering, offline methods, like pre-training and staged training, are adopted in different active vision tasks [43, 35, 37]. To circumvent unstable joint training, in [43], they resort to an iteratively training strategy to train the perception and the policy modules, in which visual observations are required to be collected from the environment with predefined trajectories. [13, 12] includes pre-trained visual encoders into their active exploration agents to relieve the burden caused by joint training. For active recognition, collecting static image datasets and training the recognizer offline are laborious and might be infeasible, especially for embodied agents operating in the real world. Online methods to avoid policy exploitation, like random exploratory behaviors [23], are beneficial but inefficient, considering the reliability of rewards. Our approach, on the other hand, focuses on addressing lingering by adversarial disturbance, allowing more diversified online explorations.

Adversarial learning. Adversarial learning [29, 28, 39], which attempts more robust training by giving rise to malfunctions in machine learning models, has been widely applied to generative models [21, 32], transfer learning [9, 44] and active learning [45].

Recently, there have been a bunch of works [31, 30, 20, 27] showing interest in building robust reinforced agents by adversarial attacking. In [31], they treat the environment as an adversary, which imitates potential noises leading to failed generalization. [20] chooses to introduce perturbations to the agent's observations with an additional adversarial agent, which could uncover more unexpected failure cases than regular opponents. Our work shares a similar motivation with adversarial reinforcement learning, i.e., the adversarial disturbance generated by the antagonistic agent could prevent overfitting during policy learning.

# 3. Approach

Our goal is to identify and address the lingering issue in active recognition. We first introduce the setup and notations used in this paper. Then, we formulate the agent with two parts, i.e., the recognizer and the recognition policy, to convey its nature of multiple combinations under the same evaluation metric. From the perspective of the iterative training process, we explain how the combination could lead to lingering. The proposed method, adding adversarial disturbances to prevent lingering, together with model architecture, is described in the final.

## 3.1. Task settings and notations

The task setup is described by applying the agent to an object recognition scenario.

The agent for active recognition could be generally denoted as a single function $f \rightarrow R$, which is provided with an object instance $x$ and then predicts its category label as $\hat{y} = f(x)$. During each recognition episode, the agent is allowed to take a total of $T$ timesteps to achieve the final class prediction. An addition action $a \in A_{rec}$ for recognition, e.g., to rotate up the object, is taken at each timestep $t = 1, 2, \ldots T-1$. By taking movements, the agent is then able to obtain another observation of the target instance $x$. The total movement steps are fixed in this paper to compare the recognition performance better. Note that early terminations are allowed by adding the "stop" action to the action space during training.

To be more specific, we evenly discretize potential cam-era positions on the sphere around the target object into a view grid with the size of $M$ azimuths $\times$ $N$ elevations. The action is therefore defined as the difference of viewpoint coordinates around the target object by taking a movement, i.e., $a_t = \Delta c_{t-1,t}$, where $c_t$ is the corresponding camera viewpoint at time $t$. With the projection function $P(\cdot)$ from 3D to 2D, the observed visual input at time $t$ is $v_t = P_x(c_t)$.

Besides the category prediction, the active recognition agent is also required to select actions during exploration. The objective is, thus, three-fold, including evidence aggregation during exploration, efficient movements, and classification based on the collected information.

## 3.2. Active recognition formulation

We comprehend active recognition as a procedure in which the agent continuously reaches more informative views and performs classification. Recalling the motivation behind active recognition, the agent moves as the single static image does not contain enough information for an unambiguous classification. On the other hand, the desire to make movements is dramatically lessened if the recognizer is perfect, as it can recognize the object from every viewpoint, which is unlikely, especially in an unconstrained

environment. We also discuss the policy degeneration when a strong recognizer is presented in Sec. 4.6.

Our basic active recognition system is modeled with two groups of parameters, i.e., $\theta$ and $\varphi$, to denote the recognizer module and the recognition policy module. The recognizer module, defined as $q_\theta$, is a non-linear mapping function that takes in aggregated information $h_t$ and predicts the label as $\hat{y} = \arg\max q_\theta(y|h_t)$. In the proposed approach, we combine a visual encoder and a recurrent neural network to fuse observations to a hidden vector $h$ recurrently. During training, the recognizer is granted to predict an additional output, the next hidden vector $\hat{h}_{t+1}$, to encode view correlations and object structure knowledge into the recognizer. The recognizer during training is then formulated as $q_\theta(\hat{y}, \hat{h}_{t+1}|h_t)$.

The second module, i.e., the recognition policy, is treated as a partially observable-MDP, which attempts to maximize the cumulative discounted reward. The pdf of the stochastic policy is defined as $\pi_\varphi(a_{t+1}|h_t)$. In other words, the policy iteratively predicts action distributions with the previous aggregated information, i.e., the hidden vector.

Given the i-th object instance $x^i$, the category prediction of active recognition agent to timestep $t$ is formulated as:

$$\hat{y}^i = f_{\theta,\varphi}(x^i) = \arg\max_y q_\theta(y|v_0, \ldots, v_t), \qquad (1)$$

where $v_t = P_x(c_{t-1} + \arg\max_a \pi_\varphi(a_t|h_{t-1}))$. The over-all training objective is $L_{f_{\theta,\varphi}} = \sum_i |y^i - f_{\theta,\varphi}(x^i)|$, where $|\cdot|$ is the distance measurement.

With no loss of generality, we consider a two-step active recognition process on the object instance $x^i$. We have its specific loss as:

$$\begin{aligned} l^i &= |y^i - f_{\theta,\varphi}(x^i)| = |y^i - \arg\max_y q_\theta(y|v_0, v_1)| \\ &= |y^i - \arg\max_y \frac{q_\theta(y, \hat{v}_{a_1}|v_0)}{\pi_\varphi(a_1|v_0)}|, \end{aligned} \qquad (2)$$

where we use $v_0$ to represent the hidden vector $h_0$ as it is the only observation obtained. The detailed derivation process is included in the supplementary. Note that the loss term could not be directly optimized as the action selection process is non-differentiable. As the recognizer and the policy parts form a multiplication in Equation 2, multiple solution combinations exist to achieve the same loss. However, overfitting to a specific solution should always be avoided during training, which hurts the robustness of the agent to deal with unexpected environmental changes.

### 3.3. Lingering in learning active recognition

After introducing the multi-solution nature of active recognition, we formulate the training into an iterative updating process to explain the happening of lingering.

As policy learning contains non-differentiable maximum-selecting operations, it is purely updated by rewards from rolling-out experiences. On the contrary, the recognizer is directly back-propagated by training signals, like the measure between class predictions with ground truth labels. Specifically, at the training step $\tau$, we have:

- Recognizer updating:
  $\theta^\tau = \arg\max_\theta \log \mathbb{P}(\theta|D_{train}^{\tau-1})$, where $D_{train}^{\tau-1}$ is the collected observations by policy $\pi^{\tau-1}$ from the previous training step.
- Recognition policy updating:
  $\varphi^\tau = \arg\max_\varphi E_{\pi_\varphi}[R]$, where $R$ is the cumulative reward determined by the recognizer with parameters of $\theta^{\tau-1}$. That is to say, the reward $R$ reflects the recognizer's capabilities at the previous training step.

Accordingly, the recognizer inclines to correctly predict views trained in former steps and then offers rewards to drive the police to converge to the same viewpoints. Escaping from such specific solutions is unfortunately difficult as updating the reward function, i.e., the recognizer, is also data-demanding. We call this phenomenon during training the active recognition as lingering.

### 3.4. Disturbance with adversarial policy

Let us simplify the active recognition system into a symbolic multiplicative representation of two modules according to Equation 2, which is $f_{\theta,\varphi} = q_\theta \times \pi_\varphi$. Our intuition to avoid lingering during training is to introduce a perturba-tion term $\epsilon$ into the policy part, i.e., $f_{\theta,\varphi} = q_\theta \times (\pi_\varphi + \epsilon)$. As the disturbance varies during training, we prevent the active recognition agent from falling into a specific combination, i.e., improve the generalization ability by confronting more diverse situations during training. The disturbance could also be regarded as a momentum that progressively motivates the agent to explore other informative views.

Instead of modeling the disturbance with predefined noise distributions, like Gaussian noises, we express the disturbance $\epsilon$ with an adversarial policy $\pi_\psi$, which plays a competitive zero-sum game with the protagonist, i.e., the recognition policy $\pi_\varphi$. While the recognition policy finds familiar views to improve the recognition performance during training, the adversary tends to pilot the camera to more challenging or out-of-distribution views leading to failures. The competence between the two policies is demonstrated in Figure 3. By continuously digging for deficiencies in the active recognition agent during training, the agent would substantially improve its robustness over the same training object collections, which, in other words, avoids lingering.

Formally, the adversarial policy is defined as another partially observable-MDP with $\pi_\psi(\epsilon_{t+1}|g_t)$, where $g_t$ is another temporally aggregated hidden vector. The recognition

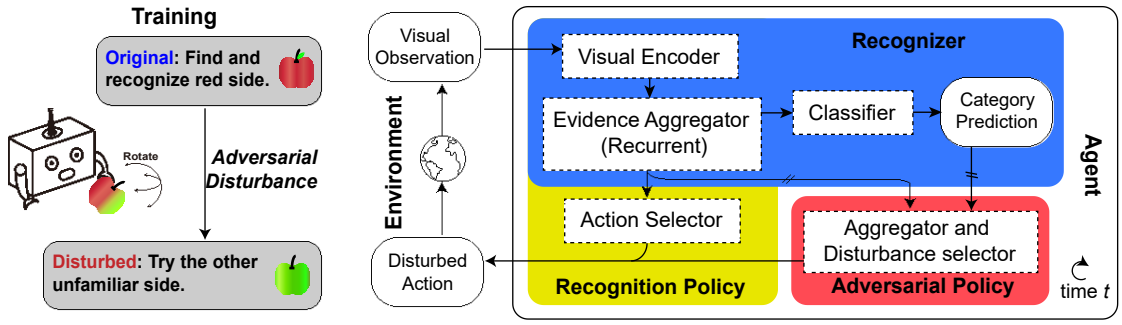



Figure 3. An overview of the proposed approach towards lingering. The left shows an example of the adversarial policy that disturbs the original policy to reach challenging while informative views. On the right, we demonstrate the architecture of the proposed approach, which is mainly composed of three modules, i.e., the recognizer, the recognition policy, and the adversarial policy.

policy's hidden vector $h_t$ and the recognizer prediction $\hat{y}_t$, together with other proprioceptions, including the timestep t and the relative position change $\Delta c_{t-1,t}$ are regarded as the observations for the adversary. In other words, the fea-ture $g_t$ contains the conditions of recognition policy and predictions over time, making the adversary able to track inconsistencies and uncertainties in the recognition agent.

The action space for the adversary is denoted as $A_{adv}$, which is set to a smaller or same-sized space with $A_{rec}$ in our approach. During training, we sample both the agent action and the disturbance at timestep t as $a_{t+1} \sim \pi_\varphi(a_{t+1}|h_t)$ and $\epsilon_{t+1} \sim \pi_\psi(\epsilon_{t+1}|g_t)$, respectively. The actual disturbed action to be taken by the agent is $\tilde{a}_{t+1} = a_{t+1} + \epsilon_{t+1}$. We assure the disturbed action still satisfies $\tilde{a} \in A_{rec}$ by masking unsatisfactory disturbances from the adversarial policy updating.

We design the rewards for two policies with exactly the opposite motivation. According to the motivation of active recognition, which is to select more discriminative views, we define the reward $r_{rec,t}(\hat{y}_t) = 1$ for the recognition policy when the category prediction is correct. On the contrary, the reward for the adversary is $r_{adv,t}(\hat{y}_t) = 1$ when the prediction is wrong. That is, the adversarial policy focuses on finding failure cases of the agent.

### 3.5. Architecture and training

Our active recognition system is modeled on the baseline architecture proposed in [23] with an additional adversarial policy to provide disturbances. An overview of our approach is demonstrated in Figure 3.

As demonstrated in Figure 3, we choose the combination of a visual encoder, i.e., multi-convolutional layers, and a recurrent neural network (LSTM), which performs a non-linear mapping from the visual observation sequence to the hidden vector h. Instead, the adversarial policy recurrently takes in the hidden vector h and the category prediction, which are not updated in the training of adversarial policy. We choose to input the category activation instead of one-hot labels to the adversarial policy. By incorporating the information, the adversary, supervised by the reward $r_{adv}$, is expected to understand the deficiencies of the current recognition agent. At each timestep t, the agent is supposed to select an action and a disturbance, both with the highest probabilities. The classifier in the recognizer, as a combination of linear layers, is then applied to produce class predictions.

**Algorithm 1:** Training the agent with the adversarial policy

Input: $E = \{(x^i, y^i)\}_{i=0}^{n}$ Environment containing n 3D objects
Initialize: Model parameters $\theta_0$, $\varphi_0$, $\psi_0$ and $\alpha_0 = 1$
while Training iterations $i = 1, \ldots$ reaches maximum do
  $\varphi_i \leftarrow \varphi_{i-1}$
  for $j = 1, \ldots, N_{rec}$ do
    Generate active recognition experiences $\{(a, \epsilon, r_{rec}, r_{adv}, \hat{y})\}$
    $\tilde{a} = a + \epsilon$ at the chance of $\alpha_{i-1}$ or $\tilde{a} = a$
    $\theta_i \leftarrow \arg\min_\theta |y^i - \hat{y}^i|$
    $\varphi_i \leftarrow$ REINFORCE with $\{(\tilde{a}, r_{rec})\}$
  end
  $\psi_i \leftarrow \psi_{i-1}$
  for $j = 1, \ldots, N_{adv}$ do
    Generate active recognition experiences $\{(a, \epsilon, r_{rec}, r_{adv}, \hat{y})\}$
    $\psi_i \leftarrow$ REINFORCE with $\{(\epsilon, r_{adv})\}$
    $\alpha_i \leftarrow$ The ratio between actual and maximum rewards of the adversary
  end
end

The recognition policy and the adversarial policy are optimized in an alternating procedure [31]. In each rotation, we alternatively hold one policy while updating the other one. The training procedure is terminated until the convergence of the active recognition agent. We outline the proposed method in Algorithm 1. Both rewards $r_{rec}$ and $r_{adv}$ are utilized in a batch policy updating algorithm, i.e., REINFORCE [40], which allows back-propagation to non-stochastic units. We define the loss for our recognition policy learning as:

$$L_{rec} = \sum_i \sum_{t=1}^{T-1} \log \pi_\varphi(\tilde{a}_t^i | h_{t-1}^i) r_{rec}(\hat{y}_t^i), \quad (3)$$

where the superscript i denotes the corresponding training sample. Similarly, the loss for the adversarial policy is:

$$L_{adv} = \sum_i \sum_{t=1}^{T-1} \log \pi_\psi(e_t \mid \dot{g}_{t-1}) r_{adv}(\hat{y}_t^i). \quad (4)$$

To stabilize the training of the active recognition policy, we reduce the influence of disturbances after the adversary cannot bring about failures. The disturbance chance is controlled with $\alpha$, the ratio between the actual obtained and maximum rewards while optimizing the adversary.

The category prediction loss is defined as $L_{category} = -\sum_i F_{softmax}(\hat{y}^i, y^i)$. Besides, there are two other losses included during the training. The $L_{entropy}$ is calculated both on the action and disturbance distribution, which also promotes producing more diversified outputs. Another term $L_{forecast}$ plays the role of encoding view correlations into our recognizer [23]. This term is formally defined as $L_{forecast} = \sum_i \sum_{t=2}^{T} D(\hbar_t^i, h_t^i)$, where the prediction of $\hbar_t^i$ is by a separate module in the recognizer with the input of $h_{t-1}^i$ and the previous action $a_{t-1}^i$. D denotes the cosine distance, which works as a similarity measure.

To sum up, the proposed method is trained with the loss:

$$L = L_{category} + L_{rec} + L_{adv} + L_{entropy} + L_{forecast}, \quad (5)$$

where each loss term is accompanied by a balance weight that is ignored here. The gradients back-propagated to each part could be tracked in Figure 3 where we use double slashes to indicate the detachment of variables. During the testing phase, the agent performs active recognition with only the recognition policy $\pi_\varphi$.

# 4. Experiments

We have three primary objectives in our experiments.

- Active Recognition Results. We compare the proposed approach with passive recognition, naive policy-based, and reinforced policy-based methods [23, 18]. We demonstrate, with the presence of the adversarial policy, our method could effectively avoid the lingering issue during training and achieves significant improvements, especially over other end-to-end trainable methods [23, 18].
- Robustness of agent. As the real-world environment is essentially noisy, robustness is critical for active recognition agents. We, therefore, introduce environmental noises to active recognition across two groups, i.e., the visual observation and the action executions.
- Adversarial policy. We examine the proposed adversarial policy by further ablation studies and comparisons with other predefined disturbance distributions.

## 4.1. Datasets and experimental setups

We evaluate the proposed method on two dataset datasets for active object [10] and scene[42] recognition.

ShapeNet Our experiments of active object recognition are conducted on the ShapeNet [10] dataset with 55 categories. The agent is given a 3D object instance for each episode and then manipulates the object with predicted movements. The class label is also predicted at every timestep until reaching the maximum steps. We discretize the viewing sphere around the target object by 30 degrees resulting in a viewing grid with $M = 12$ azimuths and $N = 12$ elevations. We set the action space of the agent to a $5 \times 5$ grid centered at the current camera location. We randomly sample 8340, 1075 and 1012 instances from the ShapeNetCore [10] for training, validation and testing, respectively.

SUN360 The SUN360 [42] is designated for our active scene recognition experiments, which has 26 diverse indoor and outdoor scene categories. The datasets contain 6174 training, 1013 validation, and 1805 testing spherical panoramas. Each panorama covering a $360 \times 180$ degrees field-of-view is then evenly separated into 32 grids with $M = 8$ azimuths and $N = 4$ elevations. For this dataset, we use the same pre-trained 1024-dim features to replace our visual encoder for fair comparisons with [23, 18]. Note that the agent could take up to $T = 5$ steps for both datasets.

## 4.2. Implementation details

The visual encoder is a simple 3-layer convolutional network. For the recognition policy, we use an LSTM to fuse temporal visual observations and other proprioceptions. We implement our adversarial policy as a single-layer Gated Recurrent Unit (GRU). During experience gathering in reinforcement learning, the starting camera viewpoint is given randomly. The training epochs $N_{rec}$ and $N_{adv}$ are set to 20 and 10 in our experiments. Moreover, we use the same-sized $A_{adv}$ with $A_{rec}$, i.e., $5 \times 5$. The balance weights in Equation 5 are set to 1, 1, 1, 0.01, and 1.5, respectively, over all datasets. We report the each-step performance considering all possible starting locations during testing.

## 4.3. Active recognition results

We extensively evaluate the proposed method against the other 5 baselines with two purposes, i.e., to show the improvements of including intelligent policies in visual recognition and the effectiveness of adversarial policy in avoiding lingering. We first introduce each baseline.

Single view: To this baseline, it consists of the same visual encoder and the classifier with our method, which only takes a random view as input. We choose this method to show the performance of single-view passive recognition.

Random views: This method shares the same visual encoder and the classifier while replacing the recognition policy with random action selections.

| Method | ShapeNet Dataset | | | | | | SUN360 Dataset | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | t=1 acc. | | t=3 acc. | | t=5 acc. | | t=1 acc. | | t=3 acc. | | t=5 acc. | |
| | w/ $c_t$ | w/o $c_t$ | w/ $c_t$ | w/o $c_t$ | w/ $c_t$ | w/o $c_t$ | w/ $c_t$ | w/o $c_t$ | w/ $c_t$ | w/o $c_t$ | w/ $c_t$ | w/o $c_t$ |
| Single view | - | 37.9 | - | - | - | - | - | 51.6 | - | - | - | - |
| Random views | - | 37.9 | - | 38.6 | - | 39.5 | - | 52.1 | - | 62.8 | - | 65.9 |
| Largest step | - | 37.9 | - | 38.2 | - | 39.0 | - | 51.1 | - | 57.0 | - | 58.3 |
| Look-Ahead[23] | 46.1±.2 | 44.9±.2 | 60.9±.3 | 58.0±.2 | 63.4±.3 | 60.3±.3 | 51.9±.2 | 51.8±.1 | 66.8±.1 | 66.4±.1 | 70.0±.2 | 69.5±.2 |
| FLAR[18] | 45.9±.2 | 45.6±.2 | 59.7±.3 | 56.8±.2 | 58.9±.2 | 59.3±.2 | 52.15±.1 | 51.7±.1 | 65.6±.1 | 64.6±.2 | 68.3±.2 | 67.6±.2 |
| Ours | 61.9±.1 | 62.0±.2 | 74.8±.1 | 74.0±.2 | 76.9±.3 | 76.4±.3 | 53.6±.1 | 54.6±.1 | 68.0±.2 | 67.4±.2 | 71.5±.2 | 69.6±.2 |

Table 1. Active recognition accuracy on both the ShapeNet dataset [10] and the SUN360 dataset [42]. The results are the average over 5 runs with different initializations. $c_t$ denotes the camera viewpoint.

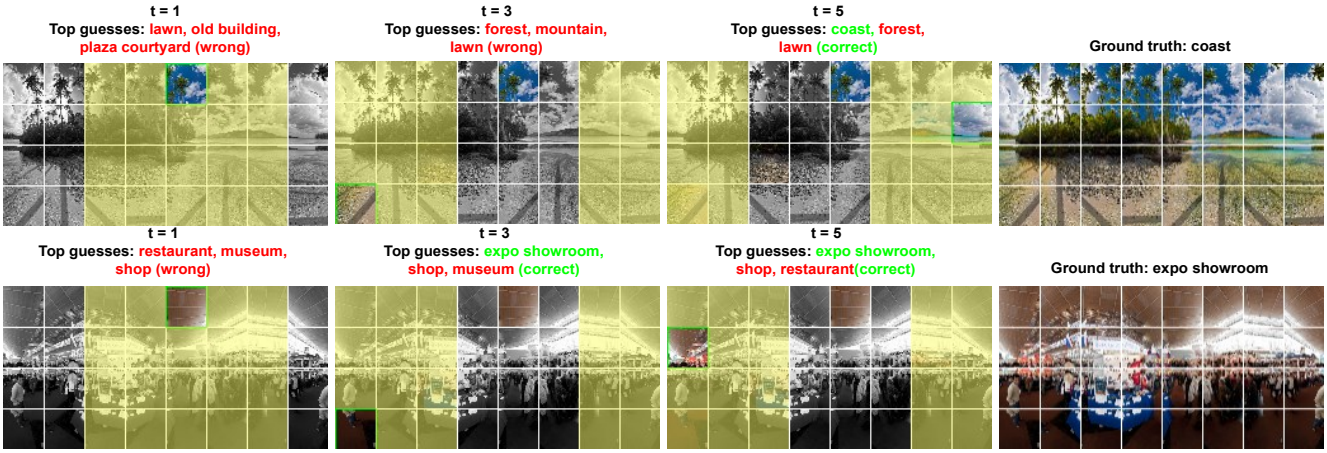



Figure 4. Our method performs active scene recognition. Each row contains results at 3 steps, i.e., t = 1, 3, 5. The current view is marked with a green box, while the next available movement is the light yellow area.

| Method | t=1 acc. | t=3 acc. | t=5 acc. |
|---|---|---|---|
| Ours+Uniform | 51.9±.1 | 67.3±.1 | 70.1±.2 |
| Ours+Gaussian | 51.9±.1 | 67.2±.1 | 70.0±.2 |
| Ours+Adversary | 53.6±.1 | 68.0±.2 | 71.5±.2 |

Table 2. Results on the SUN360 [42] with different disturbances.

| Method | t=1 acc. | t=3 acc. | t=5 acc. |
|---|---|---|---|
| Single view - during training | 99.8 | - | - |
| Random views | 67.6 | 78.6 | 80.9 |
| Ours | 67.6±.2 | 78.6±.2 | 80.9±.3 |

Table 3. Results on the ShapeNet [10] dataset by replacing the visual encoder with ResNet-18 [22].

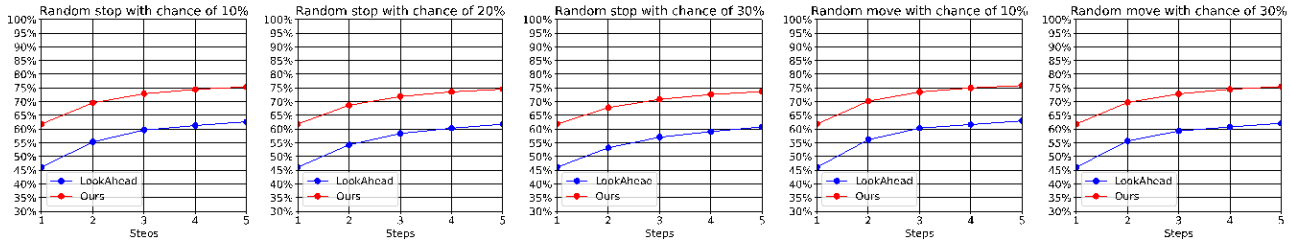
Largest step: It takes the farthest movement from the current viewpoint based on the assumption that neighboring views usually share similar information.

Look-Ahead: This baseline [23] shares the most structure with ours without the adversarial policy during training. Therefore, the improvement over this method could be considered as the benefit brought by the proposed adversary.

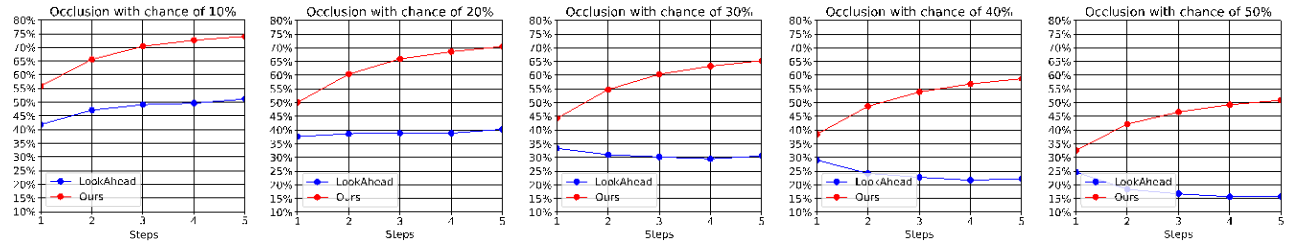
FLAR: The method [18] focuses on few-sample and lifelong learning challenges. We block its mechanism on incremental learning, leaving the agent for active recognition on fixed categories. Besides coming without the adversary, another significant difference with ours is utilizing a progressive reward function that measures the discrimination ability of each view in the embedding space.

The comparisons on both the ShapeNet [10] and the SUN360 [42] datasets are reported in Table 1. We demonstrate the recognition accuracy on timesteps t = 1, 3, 5. Since the camera viewpoint $c_t$ at each step could be unavailable in real-world scenarios, we both show the result with or without it by whether to include it as the proprioception input. All reinforcement learning-based policies, i.e., [23, 18] and ours, could already outperform both passive and naive policy-based approaches, including Random views and Largest step, denoting the advantages of involving intelligent control strategies during recognition.

Particularly, compared to other reinforced policy-based methods, including Look-Ahead [23] and FLAR [18], which share similar network architecture without the proposed adversary, the significant improvements in our recognition accuracy on both datasets are attributed to avoiding the lingering problem during learning. In other words, the compared methods [23, 18] repetitively visit limited views and then offer positive rewards to their policies which further converges to these limited views during training. Thinking of the heatmaps in Figure 2, without the proposed adversarial policy, the policy is reluctant to visit views that the recognizer fails, making the recognition accuracy drop due to insufficient training. On the contrary, the adversarial policy in the proposed method could constantly annoy the recognition policy by mining failures, which avoids lingering and achieves significant improvements on both the policy and the recognizer. On the SUN360 dataset, the overfitting issue is alleviated because of using pre-trained features instead of direct visual inputs, which also testifies to the jeopardy of lingering. The recognition process of our method is also demonstrated in Figure 4. As shown in




(a) The motor could not always successfully execute actions with random stopping and uncontrolled movements.




(b) The views of objects are randomly occluded.

Figure 5. We show the robustness comparison on the ShapeNet dataset [10] by modifying two groups of environmental setups.

the second row, our approach could disambiguate its predictions by moving to more informative observations.

From another perspective, the advantages of our method could be understood as it actively learns during training. Generally, the policy to learn and the policy to perform should be different. The agent should not only learn the skills to improve recognition (the recognition policy) but also make up for its deficiencies (the adversarial policy).

### 4.4. Robustness of agent

We evaluate the robustness of active recognition agents by introducing various environmental noises during testing. The comparisons between ours with Look-Ahead [23] that comes without mechanisms addressing lingering are shown in Figure 5 with two different groups of noises. As one of our insights is by constantly disturbing the policy such that the agent could confront different situations, the result confirmed our agent is more robust than other policy-based methods. Another interesting finding in Figure 5 (b) is that the performance of [23] even drops when views are heavily occluded with chances of 30% to 50% while ours remains increasing. The reason could be that the policy of [23] is fragile when the observation does not appear as expected, which leads to a worse temporal evidence fusion.

### 4.5. Adversarial policy

We study how different modelings of adversarial disturbances influence the performances. We choose two other disturbances with predefined distributions: the uniform and Gaussian distributions centered at no disturbance. For each movement of the training episode, the action is accordingly added with the disturbance sampled from these two distributions. We constrain the disturbance within the $5 \times 5$ view grid. The results are demonstrated in Table 2. By imitating the disturbance with an adversarial policy, it could more efficiently explore the deficiencies during training.

### 4.6. Discussion and future works

In our experiments, we find that the intelligent policy for active recognition vanishes by replacing our visual encoder (3 convolutional layers) with the ResNet-18 [22] of higher learning capacities. We show the results in Table 3. As we can observe, the ResNet-18 overfits all possible views during training which, in other words, consistently provides rewards to policy no matter what actions the agent takes. Namely, the agent has the incentive to observe other views only when the recognizer is imperfect, which leads to our future work on studying the necessity of active recognition.

## 5. Conclusions

In this paper, we study and then propose a novel approach with adversarial disturbances to address the lingering problem that happened in training active recognition. The conditions of lingering, including the multiple solution nature of joint training two modules of the agent, are explained by formulating the active recognition system and modeling the training process. To alleviate this issue, we incorporate perturbations from a reinforced agent by continuously mining undiscovered deficiencies. In other words, the adversary intelligently varies recognition experiences to prevent the agent from suffering overfitting and a monotonous policy. Experiments on two challenging datasets, along with robustness evaluation and ablation studies, confirm the effectiveness of the proposed method.

## References


[1] John Aloimonos. Purposive and qualitative active vision. In [1990] Proceedings. 10th International Conference on Pattern Recognition, volume 1, pages 346–360. IEEE, 1990.

[2] John Aloimonos, Isaac Weiss, and Amit Bandyopadhyay. Active vision. International journal of computer vision, 1(4):333–356, 1988.

[3] Alexander Andreopoulos and John K Tsotsos. A theory of active object localization. In IEEE International Conference on Computer Vision, 2009.

[4] Alexander Andreopoulos and John K Tsotsos. A computational learning theory of active object recognition under uncertainty. International Journal of Computer Vision, 2013.

[5] Nikolay Atanasov, Bharath Sankaran, Jerome Le Ny, Thomas Koletschka, George J Pappas, and Kostas Daniilidis. Hypothesis testing framework for active object detection. In 2013 IEEE International Conference on Robotics and Automation, pages 4216–4222. IEEE, 2013.

[6] Nikolay Atanasov, Bharath Sankaran, Jerome Le Ny, George J Pappas, and Kostas Daniilidis. Nonmyopic view planning for active object classification and pose estimation. IEEE Transactions on Robotics, 2014.

[7] Ruzena Bajcsy, Yiannis Aloimonos, and John K Tsotsos. Revisiting active perception. Autonomous Robots, 42(2):177–196, 2018.

[8] Dana H Ballard. Animate vision. Artificial intelligence, 1991.

[9] Zhangjie Cao, Mingsheng Long, Jianmin Wang, and Michael I Jordan. Partial transfer learning with selective adversarial networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2724–2732, 2018.

[10] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. arXiv preprint arXiv:1512.03012, 2015.

[11] Matthew Chang, Arjun Gupta, and Saurabh Gupta. Semantic visual navigation by watching youtube videos. Advances in Neural Information Processing Systems, 33:4283–4294, 2020.

[12] Devendra Singh Chaplot, Dhiraj Gandhi, Saurabh Gupta, Abhinav Gupta, and Ruslan Salakhutdinov. Learning to explore using active neural slam. arXiv preprint arXiv:2004.05155, 2020.

[13] Devendra Singh Chaplot, Dhiraj Prakashchand Gandhi, Abhinav Gupta, and Russ R Salakhutdinov. Object goal navigation using goal-oriented semantic exploration. Advances in Neural Information Processing Systems, 33:4247–4258, 2020.

[14] Devendra Singh Chaplot, Helen Jiang, Saurabh Gupta, and Abhinav Gupta. Semantic curiosity for active visual learning. In European Conference on Computer Vision, pages 309–326. Springer, 2020.

[15] Devendra Singh Chaplot, Ruslan Salakhutdinov, Abhinav Gupta, and Saurabh Gupta. Neural topological slam for visual navigation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 12875–12884, 2020.

[16] Changan Chen, Ziad Al-Halah, and Kristen Grauman. Semantic audio-visual navigation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 15516–15525, 2021.

[17] Ricson Cheng, Ziyan Wang, and Katerina Fragkiadaki. Geometry-aware recurrent neural networks for active visual recognition. arXiv preprint arXiv:1811.01292, 2018.

[18] Lei Fan, Peixi Xiong, Wei Wei, and Ying Wu. Flar: A unified prototype framework for few-sample lifelong active recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 15394–15403, 2021.

[19] Meng Fang, Yuan Li, and Trevor Cohn. Learning how to active learn: A deep reinforcement learning approach. arXiv preprint arXiv:1708.02383, 2017.

[20] Adam Gleave, Michael Dennis, Cody Wild, Neel Kant, Sergey Levine, and Stuart Russell. Adversarial policies: Attacking deep reinforcement learning. arXiv preprint arXiv:1905.10615, 2019.

[21] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. Advances in neural information processing systems, 27, 2014.

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016.

[23] Dinesh Jayaraman and Kristen Grauman. Look-ahead before you leap: end-to-end active recognition by forecasting the effect of motion. In European Conference on Computer Vision, 2016.

[24] Dinesh Jayaraman and Kristen Grauman. Learning to look around: Intelligently exploring unseen environments for unknown tasks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018.

[25] S Kasaei, Juil Sock, Luis Seabra Lopes, Ana Maria Tomé, and Tae-Kyun Kim. Perceiving, learning, and recognizing 3d objects: An approach to cognitive service robots. In Proceedings of the AAAI Conference on Artificial Intelligence, 2018.

[26] Hanna Kurniawati, David Hsu, and Wee Sun Lee. Sarsop: Efficient point-based pomdp planning by approximating optimally reachable belief spaces. In Robotics: Science and systems, volume 2008. Citeseer, 2008.

[27] Shihui Li, Yi Wu, Xinyue Cui, Honghua Dong, Fei Fang, and Stuart Russell. Robust multi-agent reinforcement learning via minimax deep deterministic policy gradient. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 33, pages 4213–4220, 2019.

[28] Shaohui Lin, Rongrong Ji, Chenqian Yan, Baochang Zhang, Liujuan Cao, Qixiang Ye, Feiyue Huang, and David Doermann. Towards optimal structured cnn pruning via generative adversarial learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 2790–2799, 2019.

[29] Xiaofeng Liu, Zhenhua Guo, Site Li, Fangxu Xing, Jane You, C-C Jay Kuo, Georges El Fakhri, and Jonghye Woo.

Adversarial unsupervised domain adaptation with conditional and label shift: Infer, align and iterate. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 10367–10376, 2021.

[30] Xue Bin Peng, Marcin Andrychowicz, Wojciech Zaremba, and Pieter Abbeel. Sim-to-real transfer of robotic control with dynamics randomization. In 2018 IEEE international conference on robotics and automation (ICRA), pages 3803–3810. IEEE, 2018.

[31] Lerrel Pinto, James Davidson, Rahul Sukthankar, and Abhinav Gupta. Robust adversarial reinforcement learning. In International Conference on Machine Learning, pages 2817–2826. PMLR, 2017.

[32] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434, 2015.

[33] Santhosh K Ramakrishnan and Kristen Grauman. Sidekick policy learning for active visual exploration. In Proceedings of the European conference on computer vision (ECCV), pages 413–430, 2018.

[34] Santhosh K Ramakrishnan, Dinesh Jayaraman, and Kristen Grauman. Emergence of exploratory look-around behaviors through active observation completion. Science Robotics, 2019.

[35] Santhosh K Ramakrishnan, Dinesh Jayaraman, and Kristen Grauman. An exploration of embodied visual exploration. International Journal of Computer Vision, 129(5):1616–1649, 2021.

[36] Andrea Roberti, Marco Carletti, Francesco Setti, Umberto Castellani, Paolo Fiorini, and Marco Cristani. Recognition self-awareness for active object recognition on depth images. In BMVC, page 15, 2018.

[37] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied ai research. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 9339–9347, 2019.

[38] Stefano Soatto. Actionable information in vision. In Machine Learning for Computer Vision. 2013.

[39] Yibing Song, Chao Ma, Xiaohe Wu, Lijun Gong, Linchao Bao, Wangmeng Zuo, Chunhua Shen, Rynson WH Lau, and Ming-Hsuan Yang. Vital: Visual tracking via adversarial learning. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 8990–8999, 2018.

[40] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. Machine learning, 8(3):229–256, 1992.

[41] Mitchell Wortsman, Kiana Ehsani, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Learning to learn how to learn: Self-adaptive visual navigation using meta-learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019.

[42] Jianxiong Xiao, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Recognizing scene viewpoint using panoramic place representation. In IEEE Conference on Computer Vision and Pattern Recognition, 2012.

[43] Jianwei Yang, Zhile Ren, Mingze Xu, Xinlei Chen, David J Crandall, Devi Parikh, and Dhruv Batra. Embodied amodal recognition: Learning to move to perceive objects. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 2040–2050, 2019.

[44] Chaohui Yu, Jindong Wang, Yiqiang Chen, and Meiyu Huang. Transfer learning with dynamic adversarial adaptation network. In 2019 IEEE International Conference on Data Mining (ICDM), pages 778–786. IEEE, 2019.

[45] Beichen Zhang, Liang Li, Shijie Yang, Shuhui Wang, Zheng-Jun Zha, and Qingming Huang. State-relabeling adversarial active learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 8756–8765, 2020.

[46] Yuke Zhu, Roozbeh Mottaghi, Eric Kolve, Joseph J Lim, Abhinav Gupta, Li Fei-Fei, and Ali Farhadi. Target-driven visual navigation in indoor scenes using deep reinforcement learning. In 2017 IEEE international conference on robotics and automation (ICRA), pages 3357–3364. IEEE, 2017.