An Azimuth Aware Deep Reinforcement Learning Framework for Active SAR Target Recognition

Xinhua Jiang, Tianpeng Liu, Yongxiang Liu, *Member, IEEE*, Shuanghui Zhang, Huangxing Lin, and Li Liu, *Senior Member, IEEE*

***Abstract*—Achieving automatic target recognition in synthetic aperture radar (SAR) imagery is a long-standing difficulty because of the limited training samples and its sensitivity to imaging condition. Active target recognition methods can offer an innovative perspective to improve recognition accuracy compared to their passive counterparts. Although prevailing in the optical imagery area, the active target recognition in SAR image processing remains underexplored. This article proposes an active SAR target recognition framework based on deep reinforcement learning for the first time, where we design a simple view-matching task and model it as a Markov decision process. The proximal policy optimization algorithm is used to help the agent learn how to alter the observing azimuth to seek more discriminative target images for the classifier. Furthermore, the single-view feature extractor is trained with the contrastive learning method to help distinguish the target images under different azimuths, allowing the agent to successfully learn the active data collection policy in the training environment and transfer it to the test environment. Lastly, the effectiveness and advancement of the proposed framework are verified on the SAMPLE dataset. When the training samples for the classifier are very scarce, it could bring around 10% more gain in target recognition rate compared to existing active target recognition frameworks.**

***Index Terms*—Active target recognition (AcTR), contrastive learning, deep reinforcement learning (DRL), synthetic aperture radar (SAR).**

I. Introduction

WITH the advantages of high-resolution, day-and-night, and weather-independent imaging, synthetic aperture radar (SAR) has been widely used in both military and civilian fields [1]. Achieving the SAR automatic target recognition (ATR) is a long-standing goal for the researcher in the remote sensing area. The past several years have witnessed the blossoming of the deep learning-based SAR ATR methods [2], [3]. For its single-view recognition branch, a key challenge is that the feature for classification should be highly robust to different observing azimuths, under which target exhibits large intraclass variation. Especially when the training samples are very scarce, it is extremely hard to overcome this obstacle for the model trained. Many previous works prove that, compared to the single-view SAR ATR, using multiple SAR images from different viewing azimuths enables a better and more robust recognition performance [4], [5]. In essence, not only can the images from different views give complementary descriptions concerning the target, but they also bring the discriminative inner correlation among different views to target recognition [6]. Accordingly, leveraging multiple images from different aspects can greatly improve recognition accuracy.

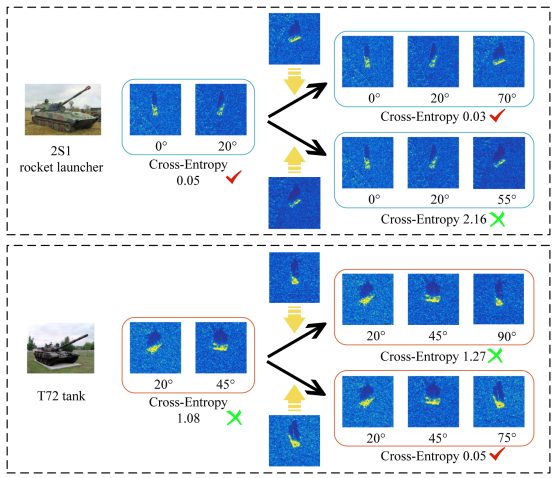

Fig. 1. Impact of observations under different azimuth angles on multiview SAR target recognition. For a classifier trained on a small dataset, a few target images under certain azimuths are conducive to recognizing the target, while the others would confuse its judgment based on past observations.

However, traditional multiview recognition methods assume that the target is observed in a fixed pattern, that is, the viewing azimuth angles are randomly distributed or in a uniformly increasing manner, ignoring the way of active data collection. In contrast, active SAR ATR methods can autonomously seek more discriminative target images to achieve high-performance recognition. As shown in Fig. 1, when performing a multiview SAR target recognition task, the impact of new observations on the recognition results based on previous observations can be positive or negative. Therefore, teaching the agent to actively observe the target based on past observations is very promising in raising recognition accuracy. In addition, SAR imaging requires a lot of time costs, storage resources, and energy from the SAR-equipped platform. Active data collection can focus these

Manuscript received 3 November 2023; revised 14 January 2024; accepted 2 February 2024. Date of publication 8 February 2024; date of current version 22 February 2024. This work was supported in part by the National Key Research and Development Program of China under Grant 2021YFB3100800, in part by the National Outstanding Youth Science Fund Project under Grant 62022091 and Grant 62322121, in part by the National Natural Science Foundation of China under Grant 61921001, Grant 62376283, and Grant 62201588, and in part by the Key Stone of the National University of Defense Technology (NUDT) under Grant JS2023-03. *(Corresponding authors: Tianpeng Liu; Yongxiang Liu.)*

The authors are with the College of Electronic Science and Technology, National University of Defense Technology, Changsha 410073, China (e-mail: everliutianpeng@sina.cn; lyx_bible@sina.com).

Digital Object Identifier 10.1109/JSTARS.2024.3363915

resources on obtaining high-quality samples, thus improving the efficiency in performing the task.

In this article, active target recognition (AcTR) in SAR imagery is defined as follows. The SAR-equipped platform serves as an agent, and it is able to autonomously determine the azimuth angle of the subsequent observation based on the observed images in the past (the observation at the initial moment is given), and all previous observations are combined to recognize the target at last, thereby improving classification accuracy using the same imaging time and storage resources. Furthermore, it should be noted that all the target images mentioned in this article, unless stated otherwise, refer to the SAR image slices containing the target.

The pioneering work for the AcTR problem in the remote sensing area was first seen in the literature [7], which proposed to improve the quality of data input to the ATR algorithm by optimizing sensor movement, settings, or collaboration between sensing platforms, thus improving the recognition rate. Unfortunately, few relevant fruits were published after this work. Pei et al. [8] proposed a multiview SAR ATR method based on unmanned aerial vehicle (UAV) path planning, which uses sufficient measured data to approximate the optimization function and transforms the SAR AcTR problem into a constrained optimization problem. However, the premise of sufficient measured data is somewhat unrealistic, and the necessity of active data collection would be greatly reduced because the performance of the multiview recognition algorithms with passively observed data can already reach satisfactory level with this condition.

According to the definition above, the AcTR task in SAR imagery asks for the capabilities of scene understanding and decision-making from the sensing platform. Deep reinforcement learning (DRL), combining the powerful approximation ability of deep neural network (DNN) and the excellent decision-making ability of reinforcement learning (RL), has made great progress in fields like robot control [9], adversarial games [10], and the foundation model training [11] with the in-depth research in recent years. In the context of the AcTR task, this learning paradigm also suits very well, which can be validated by the extensive studies and applications [12], [13]. The features of the observed images can be effectively extracted through DNN to construct the state in the Markov decision process (MDP). And the agent can learn an active data acquisition strategy from interacting with the environment through the RL algorithm. However, although prevailing in the optical imagery area, the DRL-based AcTR in SAR imagery remains underexplored. Hence, we intend to tackle the problem of SAR AcTR from the DRL perspective for the first time.

To cope with the three main difficulties in the AcTR in SAR imagery, this article proposes an azimuth aware DRL framework, referred to as AaDRL. First, a complete training environment is needed to provide plenty of interactions between the agent and the environment, so as to facilitate the policy learning. However, unlike the convenience of collecting optical images, obtaining measured SAR target images is costly and time-consuming, so its amount is usually insufficient to construct a complete training environment. To address this, we use the synthetic SAR images and a small number of measured samples to build a relatively complete training environment for the agent's policy learning, alleviating the difficulty of lacking measured samples.

Second, since SAR images are very sensitive to imaging settings, the recognition performance at various azimuth angles of the classifier would fluctuate in different environments [14]. If the reward function is designed only based on the recognition result, its mapping can be very vulnerable to the environment's variation, and the agent's policy would easily fail when transferred from the training environment to the test one. On this point, given that the classifier is inclined to be overfitted since the scarcity of the training samples, and it can only recognize a small part of target images that share the same or very close azimuth with the training samples. Hence, we design a simple view-matching task, where the RL algorithm of proximal policy optimization (PPO) [15] is utilized to help the agent learn how to search the target image as similar as possible to the training samples. With this design, the mapping of reward function could be sufficiently robust to the environment's variation.

In addition, in the scenario of the sim-to-real active SAR target recognition problem, there are two premises for policy learning and transfer. First, the image features of different targets under various azimuths can be distinguished by the policy network; second, for the features of training and test images, it is supposed that those with the same target class and azimuth should be matched, i.e., the most alike. In the existing AcTR frameworks [12], [13], [14], [15], [16], the single-view image representations are directly borrowed from that used for class identification, blurring the differences among the individual target images holding different azimuths within a single category. Besides, there is usually a distinct distribution gap between the measured and the simulated data [17]. Hence, there would be massive state representation mismatches when transferring the agent's policy to the test environment, which can also make its policy failed. In our AaDRL framework, the contrastive learning method is leveraged to train the single-view feature extraction module. In this way, an effective state representation is generated for the policy network, which could help distinguish the characteristics of different targets at various azimuths. This practice can not only raising the training sample efficiency, but also enhance the policy generalization capability in the test phase. Lastly, we conduct extensive experiments on the SAMPLE dataset [18] to demonstrate the effectiveness of the proposed framework.

The main contributions of this article are summarized as follows.

1) The DRL framework is employed to solve the AcTR problem in SAR imagery for the first time. Experimental results show a significant improvement in recognition rate by using the policy derived under the AaDRL framework compared to state-of-the-art policies.
2) A simple view-matching task is designed and modeled as an MDP, where the agent is guided with the PPO algorithm to learn how to find images that are easy to be recognized. With this design, the reward function in the MDP could remain robust to the environment's variation.
3) In our framework, the contrastive learning method is utilized to learn an effective representation that helps the

agent distinguish the target images at various azimuths. This practice can not only raise the training sample efficiency but also enhance the policy generalization capability in the test phase.

II. Related Work

In this section, we will introduce the relevant research works of multiview SAR ATR and AcTR, respectively. Next, since the conventional target's azimuth angle estimation task is close to the view-matching task in this article, we also illustrate the similarity and difference between these two works.

A. Multiview SAR ATR

A large number of existing works [4], [5] suggest that, compared to single-view SAR ATR, using multiple SAR images from different viewing angles enables a better and more robust recognition performance. Based on the means of information fusion, previous multiview SAR ATR methods could be generally divided into two categories: feature-level fusion and decision-level fusion. The former merges different image features after the feature extraction step, generating a new output, including not only targets' identity information under different aspects but also the correlation among them. Pei et al. [4] proposed a deep learning-based multiview SAR ATR framework, whose main idea is to extract features from the input images and concatenate all intermediate features layer by layer with a convolutional neural network (CNN), and the final classification result is derived based on the feature absorbing the information from all images. Bai et al. [19] utilized the bidirectional long short-term memory (Bi-LSTM) network to merge feature vectors extracted by CNN and train the whole CNN-LSTM model in an end-to-end manner, allowing extracting features from single-view images while mining the correlation in the image sequence. Similar to [19], Li et al. [20] proposed a convolutional-transformer network. In the beginning, a convolutional auto-encoder is pretrained and serves as the feature extractor, and the encoder part of the Transformer is used to explore the intrinsic correlation among feature vectors.

In contrast, the decision-level fusion method focuses on merging the classification results of the individual target images, which can be further grouped into two kinds [6]. One is the parallel decision fusion method, which assumes that the target images from different viewpoints are independent and all classification results are directly fused. In this regard, Huan et al. [21] applied principal component analysis (PCA) based and ranking-based parallel decision fusion methods. The other is the joint decision fusion method, which utilizes the intrinsic correlation between different images when calculating the classification results of the images under each azimuth angle and then fuses all the classification results. A representative work is the literature [5], where Zhang et al. applied joint sparse representation (JSR) for multiview SAR ATR. For simplicity, the multiview classification in this article adopts the parallel decision-level fusion method, where the classification results of all individual target images are fused by summing.

B. Active Target Recognition

AcTR or active object recognition (AOR), an important branch in active vision, is a continuous decision-making process during which the observation platform reduces the uncertainty of target recognition by adjusting its position or observation angle to obtain more favorable information for target recognition. On the other hand, actively observing targets can focus imaging resources on the images with more discriminative features, thus improving the efficiency of the observation platform in performing the recognition task.

Since the introduction of the pioneering work [22], AcTR has received extensive attention, and researchers have leveraged tools such as attention mechanism [23], information theory [24], [25], and RL [12] to preferentially select target observation perspectives to reduce the uncertainty of the target identity. Among them, the basic idea of the information theory method is to select the target image that can bring the maximum information gain. Methods such as Monte Carlo sampling [25], Gaussian process regression [26] can be used to estimate the information gain of different viewpoints. Paletta and Pinz [12] modeled the observation viewpoint selection problem as an MDP, where the reward function is designed based on the reduction of Shannon's entropy, and the state is defined as the fusion of the previously observed images' features. The Q-learning algorithm is used to help learn the viewpoint selection strategy, which guides the agent to search for the observation viewpoints that can bring out the most discriminative information.

The methods mentioned above are based on handcrafted features. With the revival of deep learning, DRL starts to shine in the active vision field. Malmir et al. [27] first used DRL to solve the AOR problem by enabling the agent to learn viewpoint selection strategies directly from raw image sequences. However, their feature extraction module was obtained by pretraining on the ImageNet dataset instead of training in an end-to-end manner along with the overall model. In contrast, Jayaraman and Grauman [16] proposed an end-to-end AcTR framework, where single-view processing, information fusion, and decision-making modules are trained simultaneously; however, to improve the sample efficiency of training, the authors also used a pretrained feature extraction module during the actual experiments. By reviewing literature [12], [13], [14], [15], and [16], we can find that all their single-view image representations are directly borrowed from that used for class identification, blurring the differences among the individual target images within a single category. For a well-trained agent, it selects the action based on the state observed, and the state is constructed from multiple single-view features. The representation method may succeed in the AOR task facing optical images, since the training and testing data usually share the same distribution, while in the sim-to-real context, there is always a distinct distribution gap between these two kinds of data (see [17] for graphic proof) so it is much harder to match the feature representations, in this way, the agent that has been trained in the training environment would easily mistake the state representation and take the wrong action in the test environment.

The key to this feature mismatch problem is to distinguish each individual target image, which is similar to the instance of discrimination in computer vision. To achieve this, the contrastive unsupervised learning method is utilized in this article.

C. Azimuth Estimation for Targets in SAR Imagery

Target azimuth estimation can provide important information for SAR image target recognition or other interpretation tasks [28]. For example, template-based recognition algorithms need to match the samples with all possible templates, which imposes a huge computational burden. However, if the azimuths of the test samples are known, the amount of algorithmic computation required can be greatly reduced [29]. Wen et al. [30] used target azimuth information as a self-supervised signal to help CNN learn the representation with better generalization ability for target recognition. Common target azimuth estimation methods for SAR images include sparse representation [29], corner point estimation [31], and so on.

The view-matching task designed in this article is similar to but also different from the target azimuth estimation task. Both must be trained to distinguish the target images under different azimuth angles to make viewpoint selections or label them. If the total number of decision steps in the task equals 1, the two tasks are essentially the same. However, when this number is greater than 1, the former becomes a sequential decision-making problem, which cannot be well solved by azimuth estimation alone. Instead, the agent needs to learn an effective policy to maximize the expectation of reward summation after multiple decisions, i.e., return.

III. Methodology

This section focuses on the proposed AaDRL framework. First, in Section III-A, the scenario of active SAR target recognition and the details concerning MDP modeling are described. Section III-B provides a general introduction to the proposed AaDRL framework. Sections III-C and III-D introduce the feature extraction module for single-view processing and the training process of the agent using the RL algorithm, respectively.

A. Problem Formulation

The assumed practical scenario is shown in Fig. 2. Considering the UAV's advantages of mobility and flexibility, the UAV airborne SAR imaging platform is adopted as the autonomous decision-making agent in the RL framework, and the action selection corresponds to the change of the UAV's azimuth angle when observing the target. The agent mainly consists of two functional modules: the classifier and the decision-maker. The former is fine-tuned on a small number of labeled measured samples, and the decision maker is trained in an environment built from a mixup of both measured and synthetic SAR images.

Suppose that the number of target classes is N, and we denote the measured target image slice of the nth class at azimuth angle θ as $\boldsymbol{x}_\theta^n$, and its synthetic counterpart is denoted by $\hat{\boldsymbol{x}}_\theta^n$. Assuming that the depression angles of all target images are kept constant. The dataset used to train the classifier consists of

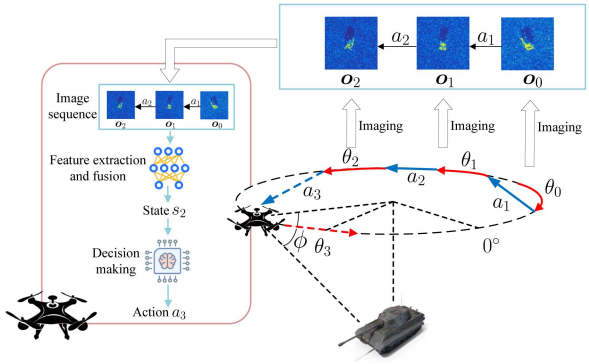

Fig. 2. Scene of AcTR using the SAR-equipped UAV platform. By sequentially altering the observing azimuths, the well-trained agent can obtain multiple target images, based on which it continues to make the next decision. The final recognition result is derived from all the newly collected images.

a small number of measured samples, which is defined as

$$\mathbf{D}_{\text{train}} = \{\boldsymbol{x}_\theta^n | n \in \{0, 1, \ldots, N-1\}, \theta \in \{\tilde{\theta}_1^n, \ldots, \tilde{\theta}_M^n\}\} \quad (1)$$

where M is the sample number for each class of target, $\tilde{\theta}_m^n$ denotes the azimuth angle corresponding to the mth sample of the nth class. For simplicity, we uniformly discretize the interval $[0, 360^\circ)$ into an azimuth angle set $\mathbf{\Pi}$, and the minimum interval between adjacent azimuth angles is $\Delta\theta$. The classifier can be trained by fine-tuning or other few-shot learning methods. For simplicity, the article adopts fine-tuning the pretrained ResNet18 [32] on $\mathbf{D}_{\text{train}}$ to obtain the classifier. When used for multiview recognition, the unnormalized log probability vectors corresponding to single-view images are summed and fed into the softmax layer, deriving the result of multiview target recognition.

The dataset used for the agent's training includes both the measured and simulated data, which is defined by

$$\mathbf{D}_{\text{train}}^{\text{RL}} = \{\hat{\boldsymbol{x}}_\theta^n | n \in \{0, 1, \ldots, N-1\}, \theta \notin \{\tilde{\theta}_1^n, \ldots, \tilde{\theta}_M^n\}, \theta \in \mathbf{\Pi}\} \cup \mathbf{D}_{\text{train}}. \quad (2)$$

And the test set for the agent is written as

$$\mathbf{D}_{\text{test}}^{\text{RL}} = \{\boldsymbol{x}_\theta^n | n \in \{0, 1, \ldots, N-1\}, \theta \in \mathbf{\Pi}\}. \quad (3)$$

The training and test environments of the agent are constructed based on $\mathbf{D}_{\text{train}}^{\text{RL}}$ and $\mathbf{D}_{\text{test}}^{\text{RL}}$, respectively, and the azimuthal distributions of the target images in the two environments are identical. Before training, the AcTR task needs to be modeled as a MDP, which is commonly represented by a tuple $\langle \mathcal{S}, \mathcal{A}, P, r, \gamma \rangle$, where $\mathcal{S}$ and $\mathcal{A}$ denote state space and action space, respectively. $P(s'|s, a)$ represents the state transition function, which indicates the probability that the state shifts to s' after taking an action a at the state s, and $r(s, a)$ denotes the reward function, used to calculate the reward value fed back from the environment after the agent takes an action a at the state s. γ means the discount factor and its value could reflect the preference for the reward at present over the future reward. To help understand the MDP modeling, a simple flowchart concerning the interaction between the agent and the environment is given in Fig. 3. The subscripts of each letter in the figure indicate the time step.

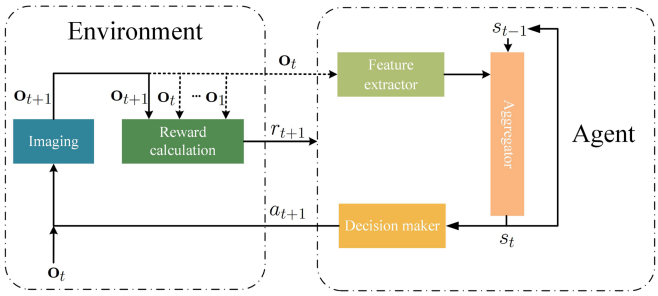

Fig. 3. Flowchart concerning the interaction between the agent and the environment. At the time step t, the agent forms the state s_t by extracting and aggregating the features from the historical image sequence, based on which the decision maker selects the action at $t+1$ time step. Next, the environment feeds the new observation $\boldsymbol{o}_{t+1}$ and the reward r_{t+1} back into the agent. This loop is ended when the time allowed is used up.

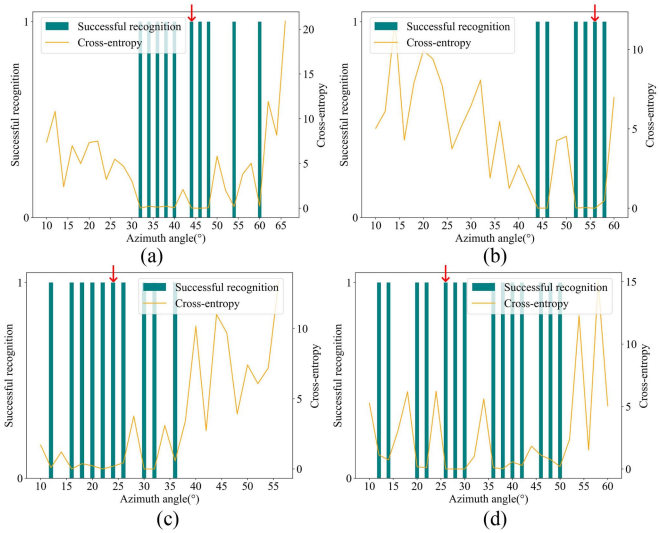

Fig. 4. Classification performances versus azimuth angle concerning four types of targets in the test environment. The green bar denotes whether the image is correctly recognized, with 1 for successful recognition and 0 referring to the opposite. The yellow solid line indicates how far the classification results deviate from the correct labels. Let the number of training samples be $M=1$ per class, and the corresponding azimuths of these training samples are labeled with red arrows in the figure. The classifier can easily recognize those images sharing the same or a similar azimuth angle with the training samples in $\mathbf{D}_{\text{train}}^{\text{RL}}$, while it is very uncertain to classify the images at other azimuths correctly. (a) 2S1. (b) BMP2. (c) BTR70. (d) T72.

For instance, $\boldsymbol{o}_t$ refers to the target image observed at the tth time step.

The purpose of changing the observing angle of the UAV is to provide the classifier with more recognizable target images, based on which we can design the reward function. To this end, we first analyze the classification performance of the classifier with the variation of the azimuth angle. Fig. 4 gives several instances for the qualitative analysis, presenting the classification results of the four types of targets (2S1 rocket launcher, BMP2 infantry fighting vehicles, BTR70 armored personnel carriers, and T72 tanks) in the test set $\mathbf{D}_{\text{test}}^{\text{RL}}$. The green bar denotes whether the image is correctly recognized, with 1 for successful recognition and 0 referring to the opposite. The yellow solid line indicates how far the classification results deviate from the correct labels. Let the number of training samples be $M=1$ per class, and the corresponding azimuths of these training samples are labeled with red arrows in the figure. It is obvious that during the inference process, when the classifier encounters the target image under the same azimuth as the training sample's, it can recognize the target easily, while for the rest of the target images, whether the target can be recognized is quite uncertain. In addition, it can be found that the target images "adjacent" to the training samples, i.e., images with the azimuths close to that of the training samples, have a much higher probability of being recognized successfully.

Based on the findings above, we design the view-matching task, through which the agent is guided to search for target images that are closer to or even equivalent to the training samples, and the reward function for the target of the nth class is given by

$$r(s_t, a_{t+1}) = \frac{1}{\left(\min\limits_m \left|\theta_{t+1} - \tilde{\theta}_m^n\right| + a\right)^2} - b + check\_redundance(\theta_{t+1}, (\theta_1, \ldots, \theta_t)) \quad (4)$$

where θ_t denotes the azimuth angle of the target image obtained at the tth time step, and the azimuth angle of the mth training sample is $\tilde{\theta}_m^n \in \{\tilde{\theta}_1^n, \ldots, \tilde{\theta}_M^n\}$. a and b are the hyperparameters that adjust the differences among various reward values and the preference for azimuthal interval, respectively. In addition, given that the performance of multiview recognition is better than that of single-view recognition from the perspective of average recognition rate, we use a $check\_redundance(\cdot)$ function to avoid redundancy in the observation sequence. If the newly obtained image shares the same azimuth angle with a certain one in the historical observations, then a penalty of -1 is given; otherwise, the function value is set to 0.

Next, we are going to define the state in the MDP. According to the Markovian property, the state s_{t+1} only correlates with the former state s_t, regardless of the states before. Besides, a proper state's definition is supposed to conclude all the relevant factors except action that would influence the reward $r(s_t, a_{t+1})$. Therefore, the state is defined by the aggregation of the features of all the previously observed SAR image slices. Specifically, it is written as

$$s_t = aggregator(f(\boldsymbol{o}_0), f(\boldsymbol{o}_1), \ldots, f(\boldsymbol{o}_t)) \quad (5)$$

where $f(\cdot)$ represents the single-view feature extractor, and $aggregator(\cdot)$ stands for the feature aggregator, used to merge the features of historical observations, which is realized by using LSTM network or vector concatenation. Considering the number of time steps in an episode is relatively small, we use the latter for simplicity. Suppose there are T time steps in one episode, and the length of a single-view feature is L. Then, the state vector's length is TL. For time step t, if $t < T$, the state vector is padded with zeros to ensure the same length of TL.

In the scenario shown in Fig. 2, the agent's action is defined as the UAV platform selecting the next azimuth angle from the set $\mathbf{\Pi}$ based on the current state and then planning the trajectory to image the target from the expected aspect. For simplicity,

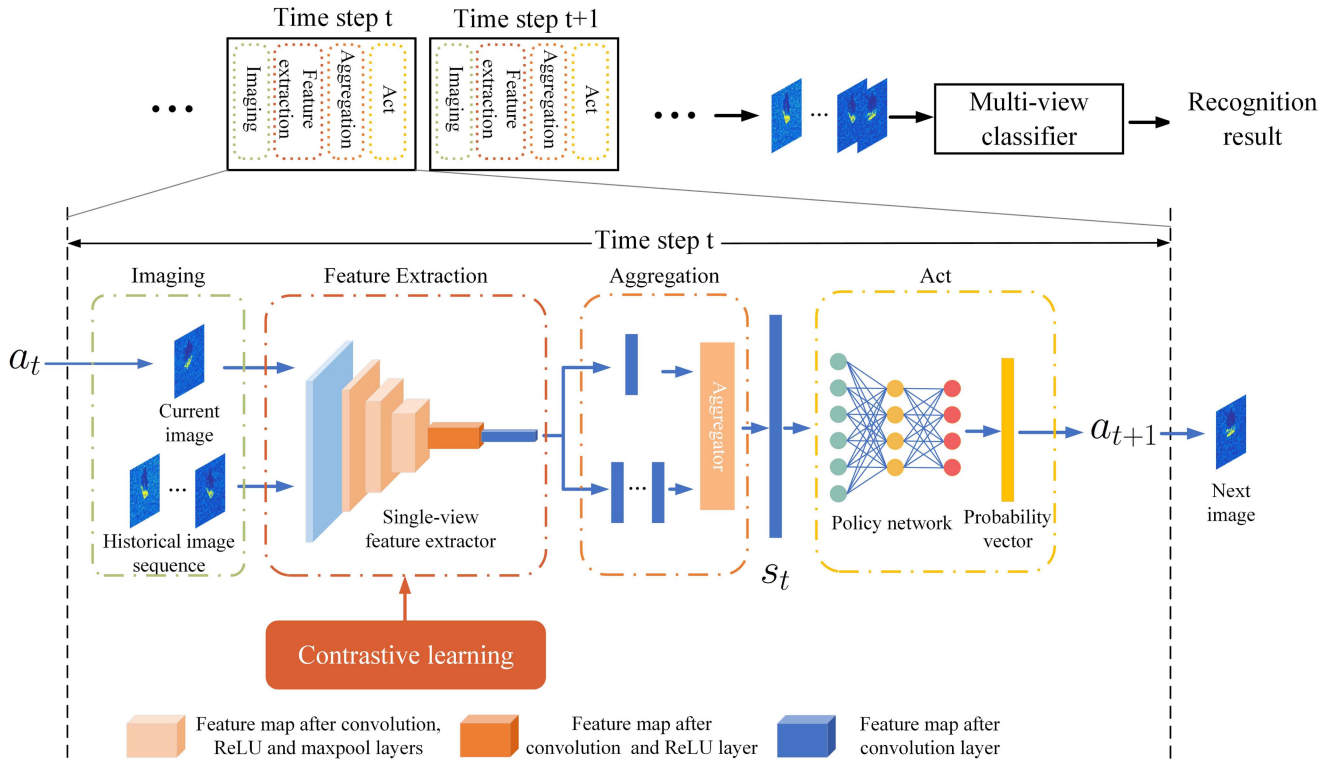

Fig. 5. Inference flowchart of the proposed AaDRL framework. The top row depicts the high-level inference workflow of the AaDRL framework, which is also another demonstration for Fig. 2. The bottom part describes the details of imaging, feature extraction, aggregation, and taking action within a single time step. The single-view feature extractor, pretrained with the contrastive learning method, is used to process the newly and historically acquired target images. According to the feature fusion result, namely the state s_t, the policy network computes a probability vector. At training time, the action a_{t+1} is stochastically sampled by the probabilistic distribution, and the policy net, along with the aggregator, is trained using the PPO-clip algorithm accordingly. In the test phase, the action a_{t+1} corresponds to the maximum's index of the probability vector.

the action space is assumed to be discrete, and the minimum difference between the different azimuth angles is set to be $\Delta\theta$. In this way, we can guarantee that the image obtained after taking action still exists in the mastered training set. Action a is expressed by

$$a \triangleq i\Delta\theta, i = 0, 1, \ldots, \left\lfloor \frac{2\pi}{\Delta\theta} \right\rfloor. \tag{6}$$

Suppose the error of UAV flight control is ignored, then the state transition in the environment becomes deterministic, and the relationship between the azimuths of the target images before and after the action is given by

$$\theta_{t+1} = (\theta_t + a_{t+1}) \bmod 2\pi, \quad \theta_t, \theta_{t+1} \in \mathbf{\Pi}. \tag{7}$$

Considering that the target images collected at all time steps contribute the same to the final recognition result, the discount factor γ is set to 1. At last, based on the MDP modeled above, the AcTR problem in SAR imagery is transformed into the sequential decision-making problem, whose optimization objective is written as

$$\pi^* = \arg\max_{\pi} E_{(s,a)\sim\rho_\pi} \left[\sum_{t=0}^{T-1} \gamma^t r(s_t, a_{t+1}) \right]. \tag{8}$$

π denotes the agent's policy, and ρ_π means the occupancy measure corresponding to π, namely, the joint probability distribution of state-action pair (s, a) influenced by policy π and the environment model. Targeting this optimization objective, we could apply the existing DRL algorithms to help learn the optimal policy π^* based on the training set and transfer it to the test environment.

B. Overview of the AaDRL Framework

Based on the MDP modeled above, an AcTR framework for SAR imagery is proposed, which mainly includes four modules: single-view feature extractor, feature aggregator, classifier, and policy network. The forward inference flowchart of the AaDRL is demonstrated by Fig. 5. Assume there are T time steps in an episode, and the agent is going to make $T-1$ times decisions after the initial observation and obtain $T-1$ new images accordingly. Considering that the azimuth of the initial observation is randomly distributed, its contribution to target recognition is very uncertain. Hence, we only adopt the latter $T-1$ images as the classifier's input.

In the proposed framework, the single-view feature extractor is purely composed of convolutional networks, constructed by modifying the last layer of A-ConvNet [2], which is proved to be efficient in SAR image feature extraction. As illustrated in Fig. 5, in the tth time step, the state s_t is derived through feature extraction and aggregation, and the policy network computes the action probability vector based on the state input and selects the action thereby.

Algorithm 1: The Active Target Recognition Algorithm Derived Under the AaDRL Framework.

Input: the initial observation $\boldsymbol{o}_0$, imaging function $im(\boldsymbol{o}, a)$, the number of time steps T per episode
Output: the predicted target label y

1: state $s_0 = aggregator(f(\boldsymbol{o}_0))$
2: loop: $t = 1, 2, \ldots, T-1$
3: $a_t = \arg\max_a \pi(a|s_{t-1})$
4: $\boldsymbol{o}_t = im(\boldsymbol{o}_{t-1}, a_t)$
5: $s_t = aggregator(f(\boldsymbol{o}_0), f(\boldsymbol{o}_1), \ldots, f(\boldsymbol{o}_t))$
6: return $\boldsymbol{o}_1, \ldots, \boldsymbol{o}_{T-1}$
7: $y \leftarrow classifier(\boldsymbol{o}_1, \ldots, \boldsymbol{o}_{T-1})$

During the training phase, if we rely only on the backpropagation from the RL algorithm to update the feature extractor, the sample efficiency would be too low to derive an effective representation. To this regard, we adopt the pretraining way. In the scenario of the sim-to-real active SAR target recognition problem, there are two premises for policy learning and transfer. First, the image features of different targets under various azimuths can be distinguished by the policy network; second, for the features of training and test images, it is supposed that those with the same target class and azimuth should be matched, i.e., the most alike. In the existing AOR frameworks [16], [27], the single-view feature extraction module is usually pretrained by performing category classification task. However, this kind of practice may not well fit sim-to-real active SAR target recognition problem in this article. With the pretraining through category classification, distinction among target images sharing the same target class while holding different azimuths is blurred. In addition, there is usually a distinct distribution gap between the measured and the simulated data [17]. Hence, the features of the images sharing the same class and azimuth in the training and test environment can hardly match each other, which contradicts the second premise. In contrast, we adopt the contrastive learning method to pretrain the single-view feature extractor. By enlarging the distances among all the image features in the training environment, the policy network can better match the features corresponding to the same target class and azimuth in the training and test environment.

The pseudocode for the forward inference of the AcTR algorithm derived under the AaDRL framework is given by Algorithm 1.

C. Model Pretraining Based on Contrastive Learning

In this article, we pretrain the single-view feature extractor under the contrastive learning framework of SimCLR [33]. During the training, each sample corresponds to a target image of a certain class at a certain azimuth, and the batch size is Q. The individual sample $\boldsymbol{o}$ could be transformed into $\hat{\boldsymbol{o}}_i$ and $\hat{\boldsymbol{o}}_j$ after the processing of augmentation operations ts and ts', respectively, then their features are extracted by the encoder $f(\cdot)$, next, the extracted features $\boldsymbol{h}_i$ and $\boldsymbol{h}_j$ are mapped to $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$ by a projection layer. $\hat{\boldsymbol{o}}_i$ and $\hat{\boldsymbol{o}}_j$ form into a positive sample pair with each other and negative sample pairs with the other $2Q-2$ transformed samples. By minimizing the InfoNCE loss, SimCLR can raise the feature similarity in the positive sample pairs and enlarge the distance in the negative ones. Here, the similarity means the cosine similarity, e.g., the cosine similarity between $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$ is written as $\text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_j)$ and the loss function is expressed by

$$l_{i,j} = -\log \frac{\exp(\text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_j)/\tau)}{\sum_{k=1}^{2Q} \mathbf{1}_{[k\neq i]} \exp(\text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_k)/\tau)} \tag{9}$$

where τ is the temperature coefficient, and $\mathbf{1}_{[k\neq i]}$ denotes that if $k \neq i$, the function value is set to 1, else 0. The InfoNCE loss regarding the whole batch is given by

$$\mathcal{L}_{cl} = \frac{1}{2Q} \sum_{q=1}^{Q} (l_{2q-1,2q} + l_{2q,2q-1}). \tag{10}$$

By minimizing the loss functions above, the single-view feature extractor could learn a proper representation, so as to recognize the targets' characteristics under different azimuths.

The commonly used data augmentation operations for SAR images include random crop, noise adding, occlusion, rotation, and so on. Within the AaDRL framework, the representation learned is also supposed to overcome the cross-domain generalization problem. Since the main difference between the measured and synthetic SAR images lies in the scatters distribution and strength within the target and background, we adopt noise adding as the augmentation operation ts' while setting the other operation ts as keeping the input unchanged.

It should be noted that during the upcoming policy learning process, we keep the parameters of the pretrained feature extractor constant to guarantee its stable capability of differentiating among all kinds of targets at different azimuths.

D. Agent's Policy Learning Based on PPO-Clip Algorithm

The state in MDP could be formed by extracting and aggregating the features from the observed image sequence, based on which the policy network selects the act. In order to confer the ability of autonomous decision-making to the agent, we need to train the policy network based on the interactive experiences between the agent and the environment. In this article, the PPO-clip algorithm [15] is adopted for policy learning, which improves the sample efficiency and training stability of the policy gradient algorithm through the operations of importance sampling and restricting the interval for network parameter updating. Because of its superior performance and wide applicability to various sequential decision-making tasks, the PPO-clip algorithm often serves as a baseline in many types of research on RL. Its optimization objective is written as

$$\max_{\xi} E_{(s,a)\sim\rho_{\pi_{\xi'}}} \left[\min \left(\frac{\pi_\xi(a|s)}{\pi_{\xi'}(a|s)} A^{\pi_{\xi'}}(s,a), \right. \right.$$
$$\left. \left. \text{clip}\left(\frac{\pi_\xi(a|s)}{\pi_{\xi'}(a|s)}, 1-\varepsilon, 1+\varepsilon \right) A^{\pi_{\xi'}}(s,a) \right) \right] \tag{11}$$

where ξ' and ξ denote the policy network's parameter before and after a single iteration, and $A^{\pi_{\xi'}}(s,a)$ is the advantage function under the policy $\pi_{\xi'}$. clip$(\cdot)$ function restricts the ratio of action

selection probability within an interval of $[1-\varepsilon, 1+\varepsilon]$, where ε is a hyperparameter used to adjust the interval width for network parameters updating.

IV. Experiments

In this section, extensive experiments are conducted to verify the effectiveness of the proposed framework in improving the target recognition accuracy. First, we introduce the paired synthetic and measured SAR image dataset named SAMPLE. Next, experiment settings are introduced, and the effectiveness of the proposed framework is verified by comparing it with other active data acquisition policies under different testing conditions. Lastly, the ablation study on the AaDRL framework is conducted to analyze the effect of the contrastive learning method on policy learning and transfer.

A. Dataset and Experimental Settings

1) Dataset Description: The SAMPLE dataset, released by the US Air Force Research Laboratory in 2019, mainly includes synthetic SAR images of various vehicle targets under different observation conditions. Except for the background, target configuration, sensor parameters, observation depression angle, and azimuth angle, etc., are kept consistent with those of the measured SAR images in the MSTAR dataset [34] released by Sandia National Laboratory. Therefore, the SAMPLE dataset provides a good benchmark for studying the differences between simulated and measured SAR images and the recognition algorithms' transfer. The publicly available part of the SAMPLE dataset contains the synthetic SAR image slices of ten ground military vehicle targets (2S1 autonomous rocket launcher, BMP-2, BTR-70 armored personnel carriers, M35, M548 trucks, M1, M2, M60, T-72 tanks, ZSU-234 air defense unit), whose azimuth angles range from 10 to 80°, depression angles range from 15 to 17°. The SAR sensor works at the *X*-band while imaging, and the resolution is 0.3 m. The optical images, measured SAR images, and corresponding synthetic SAR images slices of these ten types of targets are shown in Fig. 6. In order to reduce the interference caused by the target background clutter, all the slices in the dataset are center-cropped to the size of 60×60 pixels.

2) Experimental Settings: Since we focus on SAR target recognition with a few training samples, only M measured samples per class are chosen from the SAMPLE dataset and used to form the training set $\mathbf{D}_{\text{train}}$ for the classifier. Let $N = 10$ be the number of target classes. Although the synthetic images can also be used to train the classifier, its gain is influenced by the discrepancy between the measured and synthetic data. In addition, we focus on raising the SAR target recognition performance from the perspective of active vision instead of improving the baseline performance, so we omitted the synthetic data while training the classifier for simplicity. Given that only the images with azimuth angle between 10 and 80° are made public, the set $\mathbf{\Pi}$ used in our experiments only cover the azimuth angle falling into this interval, and the minimal interval between the neighboring azimuth angles is set to 2°. The $\mathbf{D}_{\text{train}}^{\text{RL}}$ is comprised of $\mathbf{D}_{\text{train}}$ and synthetic SAR images. Since the former only takes a small part of $\mathbf{\Pi}$, we correspond synthetic images to the rest of the azimuths. In contrast, the test set $\mathbf{D}_{\text{test}}^{\text{RL}}$ is purely made of the measured samples, whose target configuration, azimuth angle, and depression angle are reciprocally equivalent to those of the samples in $\mathbf{D}_{\text{train}}^{\text{RL}}$. Because we temporarily focus on the policy generalization from the simulated to the measured environment, the depression angle of all the data used is simply set to 17°. It should be noted that although the azimuth angles of the released part of the SAMPLE dataset are 10 to 80°, there are still some missing angles in this range, so it is not strictly guaranteed that the azimuth interval of adjacent images is 2°. To approximate the ideal condition of uniform azimuthal increasing, we replace the missing images with the ones holding a larger and the closest azimuth angle.

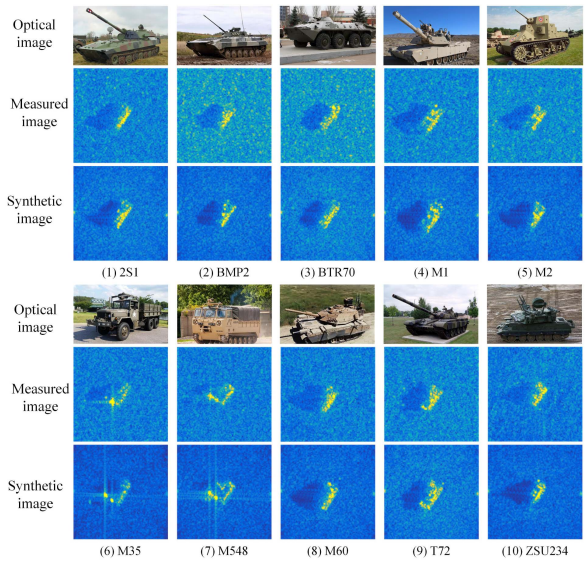

Fig. 6. Optical and SAR image slices concerning vehicle targets of ten classes in the SAMPLE dataset. By matching target configurations and sensor parameters, details in the measured and synthetic data, including shadows, orientation, scatter distribution, and magnitudes, are in good agreement [18]. Since the producer of this dataset did not align the ground planes of the synthetic and the measured images, the backgrounds of the former are somewhat darker.

For each episode in the training phase, a target image with a random azimuth is given to the agent at the beginning, several rewards are fed back from the environment according to the states and agent' actions, and the training loss in the RL algorithm is computed accordingly. During the test phase, the agent samples the target images in $\mathbf{D}_{\text{test}}^{\text{RL}}$ based on the policy learned and sends the last $T-1$ images to classifier to derive the recognition result.

B. Implementation Details

1) Hyperparameter Settings: In the PPO-clip algorithm, the hyperparameters a and b of the reward function shown in (4) are set to 1 and 0.1, respectively. Both the actor's policy network and the critic's value network are a two-layer fully connected network, with an input dimensionality of 128 and a hidden layer dimensionality of 128. The output dimensionality of the former is set to 30, with each output node corresponding to an action choice in the next step, while the output of the latter is a number used to evaluate the input state's value. These two networks are optimized using the Adam optimizer, and the learning rates

TABLE I
PERFORMANCES OF VARIOUS POLICIES UNDER DIFFERENT TEST CONDITIONS

Method		M=2				M=3			
		T=2	T=3	T=4	T=5	T=2	T=3	T=4	T=5
Passive approach	Single-view ResNet18	44.90	44.90	44.90	44.90	59.10	59.10	59.10	59.10
Heuristic approach	Random sampling	46.36±2.00	55.48±1.83	62.64±1.24	66.32±1.85	60.44±1.48	70.38±1.09	75.56±0.96	78.96±0.46
	Sequential sampling	45.42±1.33	52.54±1.98	55.38±1.86	57.90±1.36	59.72±1.65	68.70±1.37	72.46±1.52	75.96±0.86
Neural network-based approach	AaDRL	**60.22±1.70**	**74.00±1.13**	**77.76±1.78**	**84.60±1.00**	**69.96±0.92**	**79.14±1.41**	**87.72±1.42**	**90.12±1.20**
	Framework A	45.78±0.42	58.57±1.62	63.73±0.59	69.11±0.88	59.36±0.46	70.40±0.90	76.23±0.73	80.68±0.79
	Framework B	45.56±0.22	58.86±1.20	62.74±1.54	66.62±1.26	59.26±0.49	70.97±1.16	76.05±1.13	78.38±0.95

Note: The **bold** number denotes the best result. All results are the mean overall accuracy (%) ± standard deviation over 5 random seeds.

used for their updates are $lr_{\text{actor}} = 10^{-3}$ and $lr_{\text{critic}} = 10^{-4}$, respectively. The hyperparameter ε used to control the parameter update interval is 0.2, and λ in the generalized advantage estimation method is set to 0.95. The total number of the agent's interactions with the environment during the training is 300 000, and the data tuples collected in each episode are used for the subsequent ten times of network update.

During the training of the single-view feature extractor based on the SimCLR framework, the projection layer used is a two-layer fully connected network with the temperature coefficient $\tau = 1$ in the InfoNCE loss. In the augmentation operation ts', Gaussian white noise with mean $\mu_1 = 0$ and variance $\sigma_1^2 = 0.04$ is added to the SAR target images. In addition, we found that adding a little noise to the training and test sets for the classifier can effectively improve the testing recognition rate. Therefore, Gaussian white noise with mean $\mu_2 = 0$ and variance $\sigma_2^2 = 0.01$ is added in both the training and testing phases for the classifier.

2) Compared Policies: Since this article is a preliminary exploratory work in the field of AcTR in SAR images, there are very few studies available for comparison. The policies added to the comparison consist of two kinds, the heuristic policy, e.g., the policy of random sampling [35] and sequential sampling [19], and the policies derived under other AcTR frameworks [16], [27] proposed for optical images. Below is the detailed introduction to these baselines.

1) Random sampling: With this policy, the agent randomly selects an angle from the available interval $\mathbf{\Pi}$ as the azimuth of the next observation. To ensure the fairness of comparison, we add a constraint in the random sampling process so that the agent will not get duplicate target images in each episode.
2) Sequential sampling: With this policy, the azimuths of the selected target images are in a uniformly increasing manner. We set the azimuth interval for sequential sampling to 4°.
3) Framework A: In the framework proposed by [27], the state is represented by the accumulated belief in each time step, i.e., the elementwise-product of the single-view posterior beliefs over target identity. Framework A represents the altered AaDRL framework whose single-view feature extractor is replaced by the corresponding part used in [27].
4) Framework B: Similar to [16], Framework B denotes the altered AaDRL framework whose single-view feature extractor is the backbone of the ResNet18 network, first pretrained on ImageNet and then finetuned on $\mathbf{D}_{\text{train}}^{\text{RL}}$.

C. Policy Comparison

This subsection first compares the performance of various policies under different conditions. Subsequently, we present and analyze the training processes of the agent's policy under different AcTR frameworks.

Table I demonstrates the target recognition performance comparison among various approaches under different test conditions. Since the measured SAR target image is very hard to obtain in reality, the measured sample amount M per class is usually a small number in our settings. These approaches are divided into three kinds: passive, heuristic, and active approaches. The passive approach means that the classifier takes in the static single-view image input in both the training and test phases, with no aid from multiple observations and action planner. Here, we use the pretrained ResNet18 as the static classifier and finetune it on the dataset $\mathbf{D}_{\text{train}}$. The heuristic approach includes the policy of random and sequential sampling, whose action selection is intuitive. In the neural network-based approach, we compare the proposed AaDRL framework with the other two kinds, and their differences exist in the single-view feature extraction part. In the test environment with various settings, all the recognition methods are run 50 000 times under five random seeds, namely, 10 000 times for each seed, so we can obtain five average recognition results for each method and calculate the mean overall accuracy and standard derivation thereby. Table I shows that the policy derived under the AaDRL framework overwhelms the other policies under all test conditions.

1) Comparison With Heuristic Policies: We visualize the performance comparison among the first three active data acquisition policies when $M = 2$; the result is given in Fig. 7. The height of the blue rectangles in the figure indicates the single-view target recognition rate of the classifier on the test set $\mathbf{D}_{\text{test}}^{\text{RL}}$. The remaining three colored rectangles correspond to three policies: random sampling, sequential sampling, and the policy derived under the AaDRL framework. The heights of the other three rectangles represent the average multiview classification performances of the classifier under the corresponding three policies. The black vertical line at the top of the rectangles

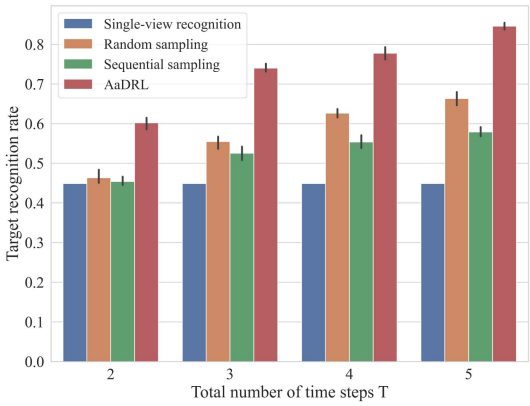

Fig. 7. Performance comparison among different active data acquisition policies when $M = 2$. The policy trained under the proposed AaDRL framework overwhelms the random and sequential sampling policies. Also, see Table I.

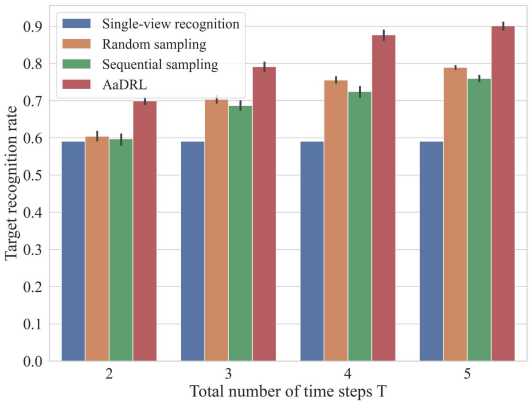

Fig. 8. Performance comparison among different active data acquisition policies when $M = 3$. Using the policy learned under the AaDRL framework, the agent can gain much higher recognition accuracy than the other policies. Also, see Table I.

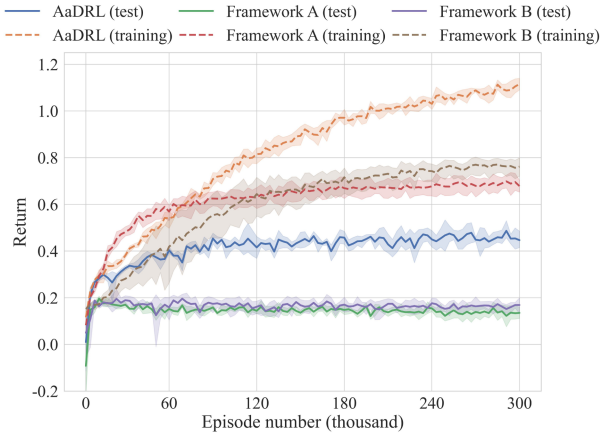

Fig. 9. Return curves for the agent during the training and test phases under different AcTR frameworks. In the training phase, the agents can learn effective control policy under all three frameworks, while in the test phase, for frameworks A and B, the agent's policies fail to transfer to the test environment. In contrast, although it is influenced by the generalization gap, the policy learned under the AaDRL can successfully improve the return value.

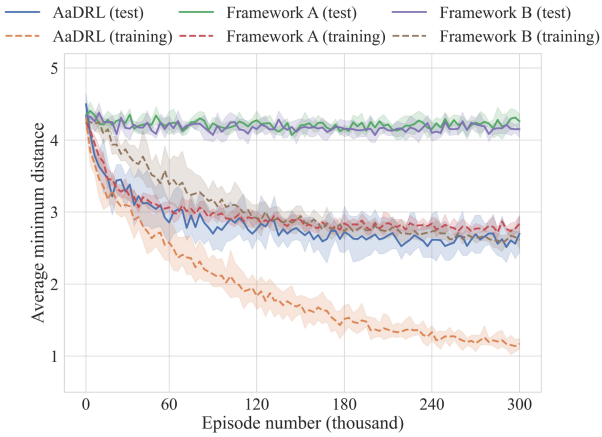

Fig. 10. Average minimum distance curves for the agent in the training and testing phase under different AcTR frameworks. Corresponding to Fig. 9, all three curves of average minimum distance gradually decrease with the ongoing iterations at the training time. However, the policies under frameworks A and B fail to perform the view-matching task in the test environment, but the policy under the AaDRL framework can still help the newly acquired target image to approximate the training samples.

represents the fluctuation range of the results, which is expressed by the confidence interval with a confidence level of 0.95. From the figure, it can be seen that the random sampling policy is slightly better than the sequential sampling policy, whereas the policy learned under the proposed AaDRL framework is significantly better than the other two active data collection policies, improving the recognition rate by more than 10%.

Similarly, Fig. 8 presents the comparison result among the first three active data acquisition policies when $T = 3$, which also suggests that the agent using the policy learned under the proposed framework achieves greater recognition rate gains compared to the policies of random sampling and sequential sampling. Another finding is that as the training sample increases, the single-view recognition performance of the classifier is strengthened, and the advantage of the policy derived under our framework over others is comparatively weakened.

2) Comparison With Other AcTR Frameworks: In the following, we present and analyze the training processes of the agent's policy under different AcTR frameworks. Let the number of samples for each target type be $M = 3$, and the total number of time steps $T = 3$. As shown in Figs. 9–11, we use three evaluating indicators to reflect their effectiveness, i.e., average minimum distance, return, and recognition rate. The minimum distance in the figure denotes the smallest difference between the azimuth of the newly acquired target image and the azimuths of the training samples in each time step. Since the agent will make $T - 1$ decisions in each episode, we take the average of the $T - 1$ smallest distances as the average minimum distance. As mentioned, return is the expected sum of all rewards obtained in one episode. In addition, the recognition rate represents the classifier's performance in the test environment using active data collection policy. In Figs. 9–11, to produce these colored curves, we first set up five random seeds and ran the program under each seed to get five return curves, respectively. Considering that the initial curves obtained fluctuate a lot, these five curves are flattened by the sliding window averaging. Finally, we use

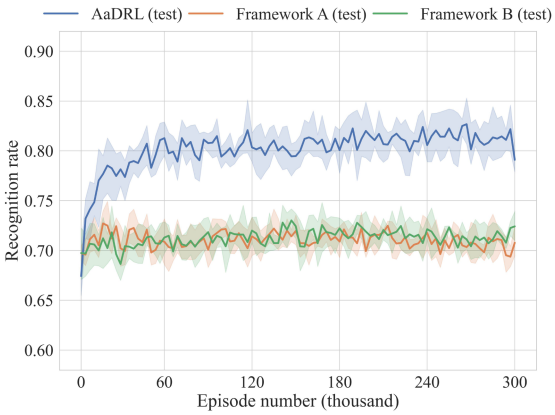

Fig. 11. Recognition accuracy curves for the agent in the test environment under different AcTR frameworks. By using the AaDRL framework, the policy learned can bring the recognition rate gain about 10% when it comes to convergence, while under the other two frameworks, the recognition rate can barely be improved by the agent's policy.

the visualization tool Seaborn to draw the following curves by integrating the results under the five random seeds, where the solid lines in the shaded area represent the mean values of the averaged five curves. The shaded area indicates the fluctuation range of the data, which is expressed by the confidence interval with a confidence level of 0.95. The same type of curves in the following are also obtained according to this method. It should be noted that when a certain training epoch is finished, the updated policy would simultaneously be tested in the test environment, deriving the intermediate test result while training. In this way, we are able to observe and compare the generalization performance of different policies.

Fig. 9 demonstrates the return curves for the agent during the training and test phasees under different AcTR frameworks. In the training environment, as the number of episodes increases, the return values achieved by the policies under three kinds of frameworks all gradually increase, whereas from the extent of improvement, the proposed AaDRL framework is superior to the other two counterparts. In the test phase, for frameworks A and B, the indicator of return nearly remains unchanged with the ongoing iteration, reflecting that the policies learned under these frameworks fail to transfer to the test environment. For the AaDRL framework, although its return improvement is weakened compared to that in the training environment, namely, there is an evident generalization gap in the policy transferring, its policy can still successfully improve recognition accuracy in the test environment.

From the other side, recalling (4), it can be inferred that as the return value rises, the azimuths of the target images newly obtained after the decision-making would get progressively closer to the azimuths of the training samples. This hypothesis is validated by Fig. 10, where all three curves of average minimum distance gradually go down with the increasing episodes in the training environment, showing the agent can perform the view-matching task well under all three frameworks. In the test phase, although influenced by the generalization gap in the policy transferring, the policy under the AaDRL framework can also help the newly acquired target image to approximate the training samples. However, corresponding to the flat curve in Fig. 9, the policies under frameworks A and B fail to help reduce the average minimum distance in the test environment.

Fig. 11 shows the recognition rate curves of the classifier using different active data acquisition policies in the test environment. Based on the analysis in the MDP modeling, the value of the designed reward function is positively correlated with the final recognition performance, we expect the policies derived under frameworks A and B would make little impression in improving the target recognition rate, which is validated by those two flat curves in the figure. By contrast, with the policy under the AaDRL framework, the target recognition rate can gain about 10% when it comes to convergence.

From Figs. 9–11, with the increasing training episodes, all the frameworks successfully help raise the agent's return and lower the average minimum distance in the training phase, indicating the agent could learn a useful policy under each circumstance in the training environment, whereas from the extent of indicator improvement, the proposed AaDRL framework is superior to the other two counterparts. In the test phase, for frameworks A and B, all three indicators nearly remain unchanged as the number of training episodes increases, which means the policies learned under these frameworks fail to transfer to the test environment. In contrast, although it is influenced by the generalization gap, the policy learned under the AaDRL can successfully improve recognition accuracy in the test environment. The key to this discrepancy lies in the representation capability of the single-view extractor, as explained in the related work part. With only the single-view extractor directly borrowed from that used for class identification, the differences among all individual target images holding different azimuths within a single category would be blurred, and it is nearly impossible to match the feature representations of the training and test data sharing the same azimuth. In this way, the agent trained in the training environment can easily mistake the state representation and take the wrong action in the test environment, thus causing the failed policy transfer.

To further illustrate the advancement of the AaDRL framework, we visualize the numerical results in Table I, making the comparison result more intuitive. Except for the improvement brought by inputting the multiple observations, the policies obtained under frameworks A and B contribute little to raising the recognition performance because of the failure in policy transfer. From both Figs. 12 and 13, we can tell that the policy derived under framework A performs slightly better than that of framework B when T is relatively large, and their performances are nearly the same when $T \leq 3$. By using framework A or B, there is at most a 3% increase in recognition rate over the policy of random sampling. In contrast, the recognition performance is greatly improved with the AaDRL's active data collection policy apart from the gain from multiview recognition.

D. Analysis

1) Ablation Study: First, we experimentally verify the necessity of unsupervised pretraining of the single-view feature extractor in the AaDRL framework. The framework used for comparison is generally the same as the proposed framework,

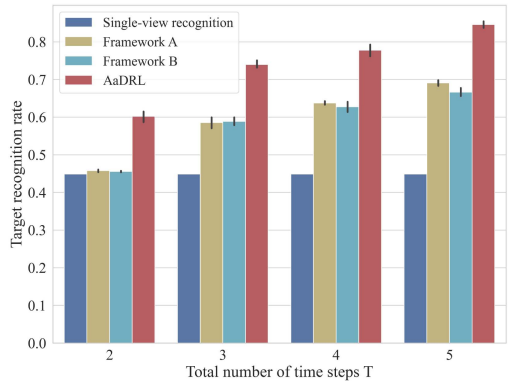

Fig. 12. Performance comparison among policies derived under three kinds of AcTR framework when $M = 2$. The policy trained under the proposed AaDRL framework is much more effective than the other two policies, with an advantage of more than 10%. Also, see Table I.

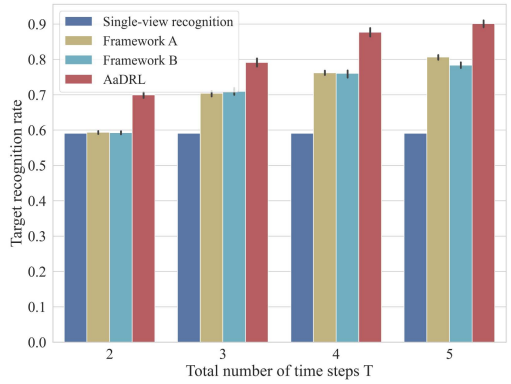

Fig. 13. Performance comparison among different policies derived under three kinds of AcTR framework when $M = 3$. Using the policy learned under the AaDRL framework, the agent can gain much higher recognition accuracy than the other policies. Also, see Table I.

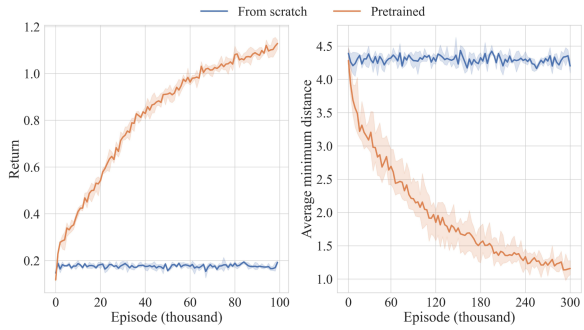

Fig. 14. Impact of the contrastive unsupervised pretraining on the agent's learning under the framework of AaDRL. The comparison above reflects the necessity of using the contrastive learning method to help the agent learn an effective policy from the interactions.

except that its feature extraction module is trained from scratch along with the policy network. Both of them are trained using the same dataset $\mathbf{D}_{\text{train}}^{\text{RL}}$, and the rest of the experimental setup is consistent with that of Fig. 9.

Fig. 14 demonstrates the impact of contrastive unsupervised pretraining on the AaDRL framework. When trained from scratch, the network parameters in the single-view feature extractor are updated by the backpropagation from the PPO-clip algorithm. It can be noticed from Fig. 14 that the agent has not learned an effective policy in this way. There may be two reasons accounting for this failure. First, the dataset used is not abundant enough; besides, the distinction between neighboring SAR image slices is slight. Hence, it is fairly hard to extract an effective representation from the observed image purely depending on the end-to-end learning. In contrast, with the help of contrastive unsupervised pretraining, the feature extractor could offer the policy network an effective representation, enabling the agent to perform the view-matching task as expected.

TABLE II
PERFORMANCE OF ACTR VERSUS THE NOISE'S STANDARD DERIVATION σ_1

σ_1	0.01	0.1	0.2	0.3
Recognition rate	79.89±1.15	76.71±0.60	78.50±1.42	79.20±0.99

Note: All results are the mean overall accuracy (%) ± standard deviation over 5 random seeds.

Since the representation learned is closely related to the loss function in the SimCLR framework, which is decided by the temperature coefficient τ, we conduct an ablation study to see its effect on the final policy's performance in both the training and test environments. Fig. 15(a) and (b) gives the return and average minimum distance curves for the agent during the training and test phases under various temperature coefficients, respectively. The recognition rate curves for the agent in the test environment under different temperature coefficients is shown in Fig. 15(c). These subfigures are drawn in the same manner with that of Figs. 9–11. From the evaluation index of return and average minimum distance, the agent's policy is trained with a higher sample efficiency when $\tau = 0.1$ than in other settings, because it obtains the largest return value and the smallest average minimum distance after the same iterations. However, its advantage is erased when evaluated in the test environment, and the policy with $\tau = 1$ even performs a slightly better than it in terms of the return value or the recognition rate gain. In addition, the large performance degradation of all these policies in the test environment should also be noted. Both the advantage erasion of the best policy and the performance degradation of all the policies can be attributed to the generalization gap between the training and test environment. Hence, if we were to guarantee the effectiveness of the agent's policy in the test time, not only the training sample efficiency, but also the generalization capability should be focused on and well enhanced. Later, we would explain in detail how the single-view representation affects agent policy's training sample efficiency and generalization capability, from which we may find how to design a good visual representation for the agent in the AcTR task.

To further explore the impact of the single-view feature extractor on the AaDRL framework, we gradually adjust the noise's standard derivation σ_1 in the operation ts' while keeping other experimental parameters unchanged, and then different single-view feature extractors can be trained at various σ_1 settings under the SimCLR framework. The eventual agent's policy also varies with the feature extractor. Table II demonstrates the average target recognition results in the test environment corresponding to various σ_1. From the perspective of the mean value, the

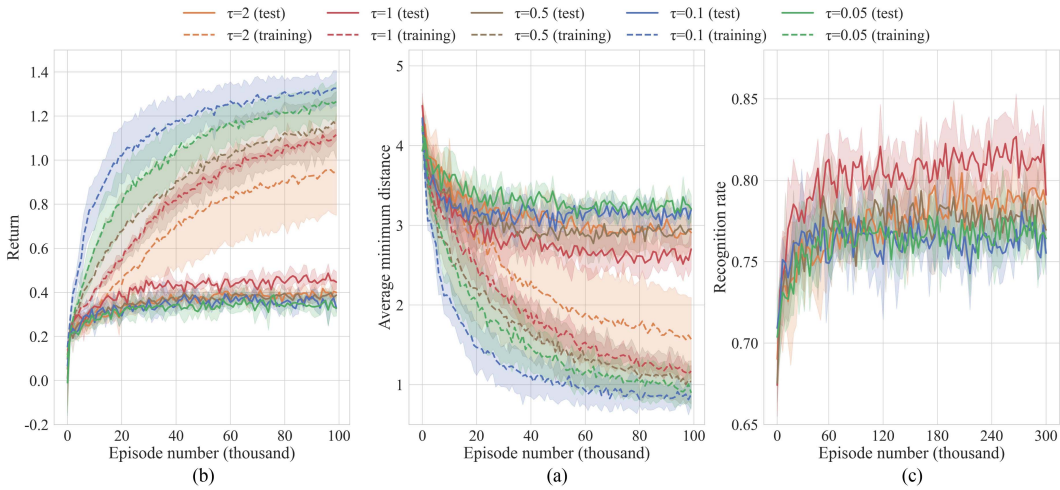

Fig. 15. (a) and (b) Return and average minimum distance curves for the agent during the training and test phases under various temperature coefficients, respectively. (c) Recognition rate curve for the agent in the test environment under different temperature coefficients. In the training step, some agent's policies can achieve higher performance by choosing a proper temperature parameter τ, while their advantage could be erased by the generalization gap between the different environments. To guarantee the effectiveness of the agent's policy in the test time, not only the training sample efficiency, but also the generalization capability should be focused on and well enhanced.

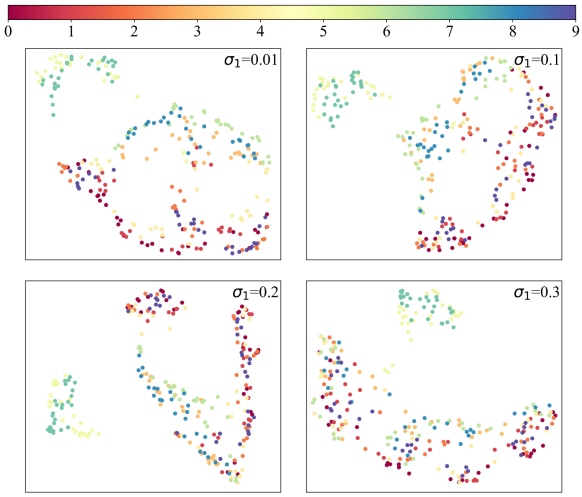

Fig. 16. Feature distributions of the training set $\mathbf{D}_{\text{train}}^{\text{RL}}$ through different single-view feature extractors. With the contrastive learning approach, the feature extraction module can help distinguish the images of various targets at different azimuths. However, there is some feature overlapping in all four cases, which would negatively impact the agent's policy learning.

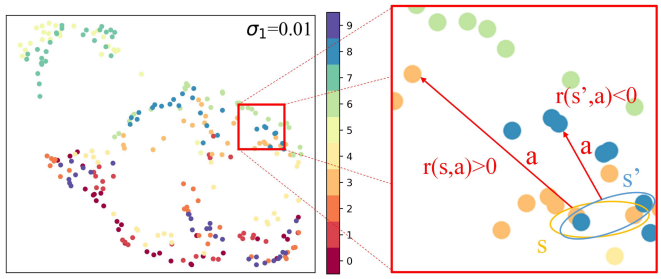

Fig. 17. Impact of feature overlapping on policy learning. The states formed by the mutually overlapped features would be very similar. If the agent takes the same action facing these similar states but obtains hugely different rewards, in the PPO-clip algorithm, the critic network's estimates of these states' values would be influenced, thus lowering the performance of the policy learned.

learned strategy performs best at the time when $\sigma_1 = 0.01$ and worst when $\sigma_1 = 0.1$. The fluctuations in the average target recognition rate for a given σ_1 are caused by the randomness inherent in the sampling and decision-making processes of the agent.

2) Impact of Feature Overlapping: Next, we use the feature visualization tool UMAP [36] to observe the distribution of image features extracted by the single-view feature extractor, and by analyzing the visualization result, we try to figure out the factors constraining the performance of the AaDRL framework. The training set $\mathbf{D}_{\text{train}}^{\text{RL}}$ is processed by different single-view feature extractors, and then UMAP is used to downscale and visualize the resulting image features in Fig. 16. Each subfigure contains ten classes of colored points corresponding to the image features of the ten classes of targets in the training set, and the points sharing the same color correspond to the target features of the same target while holding the different azimuth angles with each other. From the subfigures in Fig. 16, it can be found that the feature extraction module obtained by the contrastive learning approach can help distinguish the images of different targets at different azimuths. However, there is some feature overlapping in all four cases, i.e., the image feature of one type of target at one azimuth gets very close to the image feature of another type of target at a certain azimuth, which would pose a negative impact on agent's policy learning.

Specifically, we use Fig. 17 for illustration, where the red box on the right shows a local zoom of the feature distribution map on the left. Within the red box, the orange oval frame contains two image features of a certain target type, while the blue oval frame contains the two image features of another type of target. Suppose the former two features would form the state s during the training, and the latter two features constitute the state s'. It can be foreseen that these two states would be very similar. When

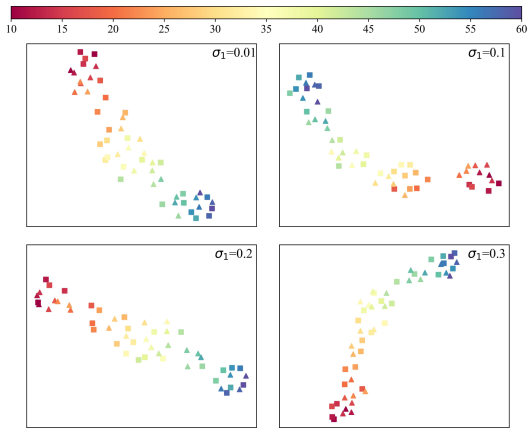

Fig. 18. Feature distributions of T72's measured and synthetic images through different single-view feature extractors. The squares represent the features of simulated images, and the triangles correspond to those of measured images. Meanwhile, the azimuth angles of target images are reflected by different colors. Ideally, the squares and triangles of the same color are supposed to be the closest to each other. However, there is a certain mismatch between them in some cases. This phenomenon inflects that when the agent is deployed in the test environment, it may misunderstand its state and consequently take the wrong action. Therefore, the severer feature mismatch would result in the worse test performance of the agent's policy.

facing s and s', if the agent takes action a on both occasions, it would get the reward $r(s,a)$ and $r(s',a)$, respectively. However, were the distinction between $r(s,a)$ and $r(s',a)$ very huge, say, $r(s,a) > 0$ while $r(s',a) < 0$, in PPO-clip algorithm, the value network's estimates to the state value $V(s)$ and $V(s')$ would be influenced given the high similarity between s and s'. Therefore, frequent feature overlapping will deteriorate the estimating accuracy of the critic network, thus lowering the performance of the policy learned. This viewpoint can also be confirmed by Fig. 9, where the agent's policy cannot enable the average minimum distance to approach 0 or even make it fall below 1 while reaching the convergence. The phenomenon of feature overlapping in the training phase may be the bottleneck that restricts the performance of the agent's policy, since this constraint indirectly leads to a lower upper bound of the agent's performance in the testing phase.

3) Impact of Feature Mismatch: In addition to the feature overlapping, the feature mismatch problem in the testing phase would also challenge the policy learned. Fig. 18 visualizes the image features of the simulated and measured T72 tank SAR images extracted by the single-view feature extraction module, where the squares represent the features of simulated images and the triangles correspond to those of measured images. Meanwhile, the azimuth angles of the target images are reflected by different colors. From the subfigures, it can be seen that with the help of the contrastive learning approach, for the same target, the image feature changes orderly with the increase in azimuth, be it for the simulated or the measured data. However, the key to the successful generalization of the learned policy to the test environment lies in that the features extracted are robust enough to the environment's change, which means that the representations of the simulated and measured images at the same azimuth angle should be as close as possible, i.e., the distance between squares and triangles of the same color is supposed to be the smallest. However, from Fig. 18, it can be seen that the actual situation is not that ideal, where a square of one color can be the closest to a triangle of another color, i.e., there is a certain mismatch between the measured image features and the simulated image features. This phenomenon implies that when the agent faces the test environment, it may misunderstand the azimuth of the target image and consequently take the wrong action. Therefore, the more severe the feature mismatch is, the worse performance of the agent's policy would be in the test environment. Referring to the four subfigures in Fig. 18, we can find that the feature mismatch phenomenon is the most severe under the condition of $\sigma_1 = 0.1$, which to some extent can explain why the agent's policy performs the worst when $\sigma_1 = 0.1$ in Table II.

4) Model Complexity Analysis: To perform the AcTR task, the active view planning module is designed in this article and it would unavoidably increase the model complexity compared to the original target recognition model with static observations. However, according to the statistics, this extra model complexity is acceptable concerning the benefit of active vision. In terms of space complexity, the applied ResNet18 classifier contains 11.2 M parameters, while the parameter number in the policy network is around 0.6 M. The computational burden for making one single action prediction is 34.7 M FLOPs, as for the classifier, its time complexity is 286.3 M FLOPs. At about 300 K iterations between the agent and the environment constructed by the training set $\mathbf{D}_{\text{train}}^{\text{RL}}$, the policy optimization process comes to convergence. In the test phase, the mean inference time for the policy network and the classifier is 1.6 and 5.1 ms, respectively. All these statistics we report are calculated with an image slice of 60×60 pixels and the experiments in this article are conducted on a single RTX3060 GPU.

V. Conclusion

This article proposes an AcTR framework for SAR images based on DRL for the first time, which effectively improves the target recognition accuracy from the perspective of active vision. We use synthetic SAR images and a small number of measured samples to build a relatively complete and close-to-reality training environment for agent training. Second, a view-matching task is designed in the proposed framework to guide the agent to learn how to seek the target image that is as similar as possible to the training sample based on historical observations, and this kind of design can help avoid policy failure in the test environment. In addition, the contrastive learning method helps the agent learn an effective state representation that enables it to recognize the characteristics of different targets under different azimuths, laying a foundation for the agent's policy learning and transfer. Finally, the experiments' results demonstrate that the proposed framework can greatly improve the target recognition accuracy under the small sample condition. In future work, we will seek a more appropriate way to better the state representation for the policy network. In this way, the problem of feature overlapping and mismatch could be alleviated so as to enlarge the gain for SAR target recognition.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their dedicated review. And we thank W. Li, the Ph.D. candidate in our group, for the careful revision to the manuscript.

REFERENCES

[1] A. Moreira, P. Prats-Iraola, M. Younis, G. Krieger, I. Hajnsek, and K. P. Papathanassiou, "A tutorial on synthetic aperture radar," *IEEE Geosci. Remote Sens. Mag.*, vol. 1, no. 1, pp. 6–43, Mar. 2013.

[2] S. Chen, H. Wang, F. Xu, and Y. Jin, "Target classification using the deep convolutional networks for SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 8, pp. 4806–4817, Aug. 2016.

[3] S. Feng, K. Ji, F. Wang, L. Zhang, X. Ma, and G. Kuang, "Electromagnetic scattering feature (ESF) module embedded network based on ASC model for robust and interpretable SAR ATR," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–15, 2022.

[4] J. Pei, Y. Huang, W. Huo, Y. Zhang, J. Yang, and T.-S. Yeo, "SAR automatic target recognition based on multiview deep learning framework," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 4, pp. 2196–2210, Apr. 2018.

[5] H. Zhang, N. M. Nasrabadi, Y. Zhang, and T. S. Huang, "Multi-view automatic target recognition using joint sparse representation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 3, pp. 2481–2497, Mar. 2012.

[6] B. Ding and G. Wen, "Exploiting multi-view SAR images for robust target recognition," *Remote Sens.*, vol. 9, no. 11, 2017, Art. no. 1150.

[7] S.-H. Yu, P. McLaughlin, A. Zatezalo, K.-y. Hsiao, and J. Boskovic, "Integrate knowledge acquisition with target recognition through closed-loop ATR," *Proc. SPIE*, vol. 9474, pp. 145–155, 2015.

[8] J. Pei, Y. Huang, W. Huo, Y. Xue, Y. Zhang, and J. Yang, "Multi-view SAR ATR based on networks ensemble and graph search," in *Proc. IEEE Radar Conf.*, 2018, pp. 0355–0360.

[9] J. Hwangbo, I. Sa, R. Siegwart, and M. Hutter, "Control of a quadrotor with reinforcement learning," *IEEE Robot. Autom. Lett.*, vol. 2, no. 4, pp. 2096–2103, Apr. 2017.

[10] Q. Zhang and D. Zhao, "Data-based reinforcement learning for nonzero-sum games with unknown drift dynamics," *IEEE Trans. Cybern.*, vol. 49, no. 8, pp. 2874–2885, Aug. 2018.

[11] N. Lambert and L. Von Werra, "Illustrating reinforcement learning from human feedback (RLHF)," *Hugging Face Blog*, 2022. Accessed: Sep. 1, 2023. [Online]. Available: https://huggingface.co/blog/rlhf

[12] L. Paletta and A. Pinz, "Active object recognition by view integration and reinforcement learning," *Robot. Auton. Syst.*, vol. 31, no. 1/2, pp. 71–86, 2000.

[13] M. Liu, Y. Shi, L. Zheng, K. Xu, H. Huang, and D. Manocha, "Recurrent 3 D attentional networks for end-to-end active object recognition," *Comput. Vis. Media*, vol. 5, pp. 91–104, 2019.

[14] W. Li, W. Yang, W. Zhang, T. Liu, Y. Liu, and L. Liu, "Hierarchical disentanglement-alignment network for robust SAR vehicle recognition," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 10088–10106, 2023.

[15] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017, *arXiv:1707.06347*.

[16] D. Jayaraman and K. Grauman, "End-to-end policy learning for active visual categorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 7, pp. 1601–1614, Jul. 2018.

[17] N. Inkawhich et al., "Bridging a gap in SAR-ATR: Training on fully synthetic and testing on measured data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 2942–2955, 2021.

[18] B. Lewis, T. Scarnati, E. Sudkamp, J. Nehrbass, S. Rosencrantz, and E. Zelnio, "A SAR dataset for ATR development: The synthetic and measured paired labeled experiment (sample)," *Proc. SPIE*, vol. 10987, pp. 39–54, 2019.

[19] X. Bai, R. Xue, L. Wang, and F. Zhou, "Sequence SAR image classification based on bidirectional convolution-recurrent network," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 11, pp. 9223–9235, Nov. 2019.

[20] S. Li, Z. Pan, and Y. Hu, "Multi-aspect convolutional-transformer network for SAR automatic target recognition," *Remote Sens.*, vol. 14, no. 16, 2022, Art. no. 3924.

[21] R.-H. Huan and Y. Pan, "Target recognition for multi-aspect SAR images with fusion strategies," *Prog. Electromagnetics Res.*, vol. 134, pp. 267–288, 2013.

[22] D. Wilkes and J. Tsotsos, "Active object recognition," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 1992, pp. 136–141.

[23] S. J. Dickinson, H. I. Christensen, J. K. Tsotsos, and G. Olofsson, "Active object recognition integrating attention and viewpoint control," *Comput. Vis. Image Understanding*, vol. 67, no. 3, pp. 239–260, 1997.

[24] B. Schiele and J. L. Crowley, "Transinformation for active object recognition," in *Proc. IEEE 6th Int. Conf. Comput. Vis.*, 1998, pp. 249–254.

[25] J. Denzler and C. M. Brown, "Information theoretic sensor data selection for active object recognition and state estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 2, pp. 145–157, Feb. 2002.

[26] M. F. Huber, T. Dencker, M. Roschani, and J. Beyerer, "Bayesian active object recognition via Gaussian process regression," in *Proc. IEEE 15th Int. Conf. Inf. Fusion*, 2012, pp. 1718–1725.

[27] M. Malmir, K. Sikka, D. Forster, J. R. Movellan, and G. Cottrell, "Deep q-learning for active recognition of germs: Baseline performance on a standardized dataset for active learning," in *Proc. Brit. Mach. Vis. Conf.*, 2015, pp. 161–1.

[28] F. E. McFadden, "Precise pose estimation for synthetic aperture radar images of vehicles," *Opt. Eng.*, vol. 46, no. 10, pp. 107201–107201, 2007.

[29] S. Chen, F. Lu, J. Wang, and M. Liu, "Target aspect angle estimation for synthetic aperture radar automatic target recognition using sparse representation," in *Proc. IEEE Int. Conf. Signal Process., Commun. Comput.*, 2016, pp. 1–4.

[30] Z. Wen, Z. Liu, S. Zhang, and Q. Pan, "Rotation awareness based self-supervised learning for SAR target recognition with limited training samples," *IEEE Trans. Image Process.*, vol. 30, pp. 7266–7279, 2021.

[31] Y. Zhang, Y. Zhuang, H. Li, X. Zhang, and X. Zhao, "A novel method for estimation of the target rotation angle in SAR image," in *Proc. IET Int. Radar Conf.*, 2015, pp. 1–4.

[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[33] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.

[34] T. D. Ross, S. W. Worrell, V. J. Velten, J. C. Mossing, and M. L. Bryant, "Standard SAR ATR evaluation experiments using the MSTAR public release data set," *Proc. SPIE*, vol. 3370, pp. 566–573, 1998.

[35] Z. Wang et al., "Multi-view SAR automatic target recognition based on deformable convolutional network," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2021, pp. 3585–3588.

[36] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," 2018, *arXiv:1802.03426*.

Xinhua Jiang received the B.E. and M.E. degrees in information and communication engineering from the Army Engineering University, Nanjing, China, in 2018 and 2021, respectively. He is currently working toward the Ph.D. degree in information and communication engineering with the National University of Defense Technology, Changsha, China.

He has published papers in respected journals, including IEEE SENSORS JOURNAL, *IEEE Communications Letters*, and China Communications. His primary research interests include active SAR target recognition, deep reinforcement learning, and representation learning.

Tianpeng Liu received the B.E., M.E., and Ph.D. degrees in information and communication engineering from the National University of Defense Technology, Changsha, China, in 2008, 2011, and 2016, respectively.

He is currently an Associate Professor with the College of Electronic Science and Technology, Chengdu, China. He has published numerous papers in respected journals, including IEEE TRANSACTIONS ON AEROSPACE AND ELECTRONIC SYSTEMS and *International Conference on Signal Processing*. His primary research interests include radar signal processing, electronic countermeasure, and cross-eye jamming.

Yongxiang Liu (Member, IEEE) received the B.S. and Ph.D. degrees in information and communication engineering from the College of Electronic Science, National University of Defense Technology, Changsha, China, in 1997 and 2004, respectively.

He is currently a Full Professor with the National University of Defense Technology. He has published numerous papers in respected journals, including IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON GEOSCIENCE, AND REMOTE SENSING. His research interests mainly include radar imaging, SAR image interpretation, and artificial intelligence.

Shuanghui Zhang received the B.S., M.S., and Ph.D. degrees in information and communication engineering from the National University of Defense Technology (NUDT), Changsha, China, in 2011, 2013, and 2016, respectively.

He worked with Nanyang Technological University, Singapore, as a Visiting Ph.D. Student in 2015. He has been with NUDT since 2017, where he is an Associate Professor with the College of Electronic Science and Technology. His research interests include compressive sensing, sparse signal recovery techniques, and Bayesian inference and their applications in radar signal processing.

Huangxing Lin received the B.S. degree in information and communication engineering from Beijing Jiaotong University, Beijing, China, in 2015, and the M.S. and Ph.D. degrees in information and communication engineering from Xiamen University, Xiamen, China, in 2018 and 2022, respectively.

He is currently a Postdoctoral Fellow with the National University of Defense Technology, Changsha, China. His research interests include SAR image interpretation and machine learning.

Li Liu (Senior Member, IEEE) received the Ph.D. degree in information and communication engineering from the National University of Defense Technology (NUDT), Changsha, China, in 2012.

She is currently a Full Professor with NUDT. She has held visiting appointments with the University of Waterloo in Canada, at the Chinese University of Hong Kong, and at the University of Oulu in Finland. Her research interests include computer vision, pattern recognition, and machine learning.

Dr. Liu served as a co-chair of many International Workshops along with major venues like CVPR and ICCV. She served as the leading guest editor of the special issues for IEEE TPAMI and IJCV. She also served as an Area Chair for several respected international conferences. She currently serves as an Associate Editor for *IEEE TCSVT* and *Pattern Recognition*. Her papers currently have over 10 000 citations, according to Google Scholar.