IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received June 30, 2019, accepted August 3, 2019, date of publication August 14, 2019, date of current version August 28, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2935167

Integrated GANs: Semi-Supervised SAR Target Recognition

FEI GAO[1], QIUYANG LIU[1], JINPING SUN[1], AMIR HUSSAIN[2,3], AND HUIYU ZHOU[4]

[1]School of Electronic and Information Engineering, Beihang University, Beijing 100191, China
[2]Cyber and Big Data Research Laboratory, Edinburgh Napier University, Edinburgh EH11 4BN, U.K.
[3]Taibah Valley, Taibah University, Medina 30001, Saudi Arabia
[4]Department of Informatics, University of Leicester, Leicester LE1 7RH, U.K.

Corresponding author: Jinping Sun (sunjinping@buaa.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 61771027, Grant 61071139, Grant 61471019, Grant 61501011, and Grant 61171122. The work of A. Hussain was supported by the U.K. Engineering and Physical Sciences Research Council (EPSRC) under Grant EP/M026981/1. The work of H. Zhou was supported in part by the U.K. EPSRC under Grant EP/N011074/1, in part by the Royal Society—Newton Advanced Fellowship under Grant NA160342, and in part by the European Union's Horizon 2020 Research and Innovation Program under the Marie–Sklodowska–Curie Grant Agreement 720325.

ABSTRACT With the advantage of working in all weathers and all days, synthetic aperture radar (SAR) imaging systems have a great application value. As an efficient image generation and recognition model, generative adversarial networks (GANs) have been applied to SAR image analysis and achieved promising performance. However, the cost of labeling a large number of SAR images limits the performance of the developed approaches and aggravates the mode collapsing problem. This paper presents a novel approach namely Integrated GANs (I-GAN), which consists of a conditional GANs, an unconditional GANs and a classifier, to achieve semi-supervised generation and recognition simultaneously. The unconditional GANs assist the conditional GANs to increase the diversity of the generated images. A co-training method for the conditional GANs and the classifier is proposed to enrich the training samples. Since our model is capable of representing training images with rich characteristics, the classifier can achieve better recognition accuracy. Experiments on the Moving and Stationary Target Acquisition and Recognition (MSTAR) dataset proves that our method achieves better results in accuracy when labeled samples are insufficient, compared against other state-of-the-art techniques.

INDEX TERMS Synthetic aperture radar (SAR), generative adversarial networks (GANs), semi-supervised learning, generation, recognition.

I. INTRODUCTION

Synthetic aperture radar (SAR) imaging systems adopt coherent imaging principles, which can effectively penetrate clouds and collect rich target information. In recent years, high resolution SAR remote sensing systems, such as Terra SAR-X [1] and COSMO-SkyMed [2], have been widely used in aerial surveying and space reconnaissance. SAR image recognition technology has become an active research area in SAR image remote sensing [3]. Traditional recognition methods for SAR images includes Template Matching [4], Support Vector Machine (SVM) [5], and Adaptive Boosting (AdaBoost) [6]. However, these traditional methods are largely dependent on hand-crafted features. When image data is large and complex, most of the established methods receive unsatisfactory recognition accuracy and efficiency.

Different from the traditional methods with hand-crafted features, methods based on Deep Convolution Neural Networks (DCNN) can learn to extract features from large-scale datasets automatically, which has made great progress in SAR automatic target recognition (ATR) in recent years. The recognition accuracy on the Moving and Stationary Target Acquisition and Recognition (MSTAR) dataset is over 99% [7]. However, these efficient DCNN models are based on the supervised learning which requires a large number of labeled samples. As the number of the labeled samples decreases, the recognition accuracy rapidly declines. It is expensive and time-consuming to collect a large number of labeled samples in SAR image analysis. To address this problem, many semi-supervised learning methods have been proposed [8]–[10]. Among them, deep learning methods,

The associate editor coordinating the review of this article and approving it for publication was Gerardo Di Martino.

based on generative adversarial networks (GANs), have become a hot spot in the research community.

The GANs proposed by Goodfellow *et al.* [11] can learn the distributions of the data and generate new samples. Reference [12] reported new techniques for training GANs (Improved-GAN). Based on these works, Gao et al. proposed semi-supervised GANs for SAR image recognition [13], [14]. The discriminator and the classifier share the same network in these models, but they need different training parameters, which influences each other and may not result in an optimal objective at the same time. Furthermore, the category of the generated SAR images in these models are uncontrollable. Li et al separated the discriminator and the classifier, and replaced the unconditional generator and discriminator with a conditional generator and a conditional discriminator in the Triple-GAN model [15], which solved the problems in optical image recognition. However, the positive samples of the discriminator in Triple-GAN entirely depend on the labeled images, which means the expected distribution of the generator depends only on the labeled images. One of the challenges is that the number of the labeled images is far less than that of optical images. The spatial resolution of SAR images is far lower than that of optical images. In addition, SAR imaging process often accompanies with speckles and noise [16]. These problems aggravate the instability in training GANs leading to the mode collapse problem [17]. Moreover, the unlabeled images with pseudo labels are regarded as negative samples in the adversarial training between the discriminator and the classifier in the Triple-GAN model. However, lots of the pseudo labels are correct. These image-label pairs may be harmful to the training of the generator if the discriminator treats them as negative samples, which may also exaggerate the mode collapse problem.

This paper presents the Integrated GANs (I-GAN) for semi-supervised learning. Fig. 1(a) and (b) show the comparison of the frameworks of Triple-GAN and I-GAN respectively. The I-GAN model can be summarized as three modules and two types of interaction. The three modules consists of a conditional GANs and an unconditional GANs as well as a classifier. The conditional GANs controls the matching of the generated images and the labels. The unconditional GANs improves the stability and the diversity of the generated images. The classifier builds suitable bridges between the conditional and unconditional GANs, whose cross-correlation is used to measure the interaction. There is no direct connection between the two generators or discriminators. Each generator fools two discriminators with different functions simultaneously, while each discriminator corresponds to two generators with different functions simultaneously. The unconditional discriminator can take both labeled and unlabeled images as positive samples, but only measure whether the distribution of the generated images converges to the distribution of the real images; the conditional discriminator only takes the labeled images as positive samples, but it can also reveal whether or not the images and the labels

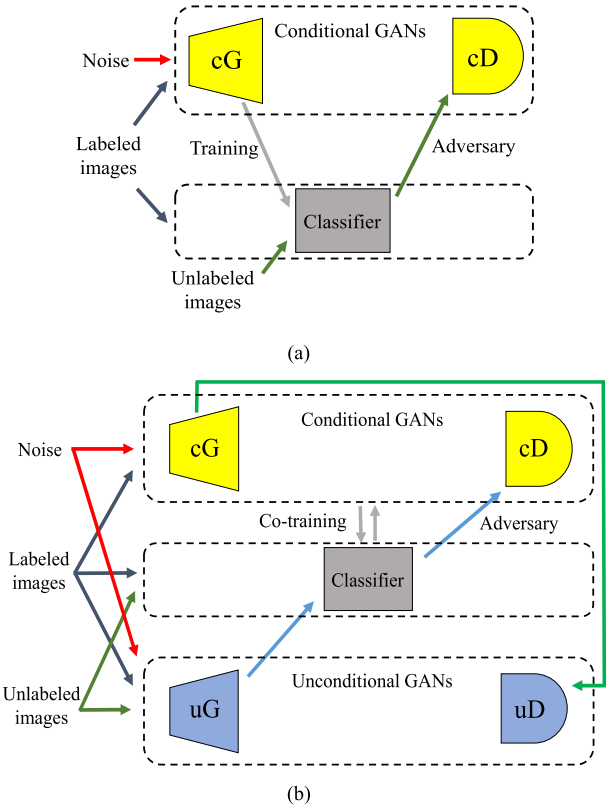

FIGURE 1. The comparison of the brief framework of the Triple-GAN and the I-GAN. (a) Triple-GAN (b) I-GAN (ours). cG, cD, uG, and uD denote the conditional generator, the conditional discriminator, the unconditional generator, and the unconditional discriminator respectively.

properly match. The other interaction is co-training [18] of the classifier and the conditional GANs. In the training process, we select the unlabeled SAR images with high confidence from the classifier and assign pseudo labels to add positive samples of the conditional discriminator. Further, we select the labeled generated images with high confidence from the conditional discriminator to increase the number of training samples for the classifier. Different from the Triple-GAN, our model have several strengths. In the adversarial training, we replace the role of the unlabeled SAR images in Triple-GAN with the images generated by the unconditional generator, which solves the problem where a large number of correct image-label pairs are regarded as negative samples by the conditional discriminator. Associating the classifier with the conditional GANs, our model utilizes the co-training method, while only the conditional GANs help the classifier in the Triple-GAN model.

Overall, our main contributions are summarized as follows: (1) We investigate the problems of the existing models based on GANs for SAR image recognition, and propose a novel GANs model for semi-supervised learning. (2) We improve the quality of the generated SAR images when the labeled samples are insufficient. (3) Our experiments on the MSTAR dataset show that the I-GAN achieves the state-of-the-art results in the SAR image recognition tasks.

The rest of this paper is organized as follows: Section II briefly introduces several related works of SAR image

recognition and generative model. In Section III, we describe our model and the training algorithm in detail. Section IV introduces the MSTAR dataset and the partitions of the training set. In Section V, we conduct the image generation and recognition experiments and compare the results with the other methods. Finally, the paper is concluded in Section VI

II. RELATED WORK

A. SAR IMAGE RECOGNITION

SAR image recognition can be divided into three stages: preprocessing, feature extraction, and classification. The preprocessing contains denoising, speckle reduction, and data enhancement. The representative preprocessing techniques include spatial filtering, wavelet transform, and spatial geometric transform. Conventional feature extraction approaches, e.g. [19], are based on hand-crafted feature such as peak intensity, center distance, and Hu moment. Machine learning methods such as Principal Component Analysis (PCA) [20] and Linear Discriminant Analysis (LDA) [21] are also widely used in feature extraction. Early classification algorithms are mainly based on Template Matching [4]. Others algorithms including SVM [5], [22], AdaBoost [6], Sparse Representation [23], and Scattering Center Models [24] are also applied to SAR image classification and yield promising results. However, the performance of these methods depend on the hand-crafted features. Hand-crafted feature extractors are usually low of efficiency when the data is large and complex.

In contrast to the traditional algorithms, DCNN based methods can extract hierarchical features automatically and achieve superior performance. In recent years, researchers have developed many efficient models for large-scale datasets: ResNet [25], DenseNet [26], SeNet [27]. With the growing prominence of DCNN, deep learning methods have been applied to SAR image recognition [28], [50]. Chen *et al.* [7] proposed a five-layer all-convolutional network and achieved 99.13% recognition accuracy in the MSTAR datasets. In order to alleviate overfitting caused by insufficient labeled SAR images, several algorithms have been proposed to achieve higher recognition accuracy with less labeled samples. Huang *et al.* [29] designed the reconstruction loss function based on a feedback bypass to transfer knowledge from a plenty of unlabeled SAR scene images. Shang *et al.* [30] introduced deep memory to SAR images recognition, which utilized an information recorder to store samples' spatial features. As a novel deep learning method, GANs has been applied to semi-supervised SAR images recognition tasks in recent studies [14].

B. GENERATIVE MODEL

Over the past decades, many deep generative models (DGMs) have been proposed, such as Restricted Boltzmann Machines (RBMs), Deep belief networks (DBNs), Variational Autoencoder (VAE) and their numerous variants. These generative models have been applied to remote sensing images recently. RBMs [31] are a special type of Markov random field which contains one layer of stochastic hidden units. DBNs [32] consist of several RBMs. It utilizes layer-by-layer pretraining to solve the gradient dispersion problem caused by the superposition of multi-layered neurons. VAE [33] introduces stochastic gradient descent to generative models to estimate complex posterior distributions. These DGMs models have been applied to the image generation task recently, e.g. Auxiliary Deep Generative Models (ADGMs) [34], which proposed auxiliary variables to build an expressive variational distribution.

Generative adversarial networks [11], based on zero-sum game theory, makes significant breakthroughs in DGMs. Neither Markov chains nor inference for implicit variables is needed during the training process. Although gradient disappearance and mode collapse make the generator and the discriminator difficult to reach Nash equilibrium, the GANs have demonstrated impressive performance to generate realistic images.

Because of the instability existing in the training of the standard GANs and considering the advantage of DCNN in image processing tasks, Radford et al proposed Deep Convolutional Generative Adversarial Networks (DCGANs) [35], which replaced the multi-layer perceptron in the standard GANs with a DCNN architecture. Arjovsky et al proved that Jensen-Shannon Divergence would lead to the disappearance of gradient in the training process of GANs and substituted the Wasserstein Distance [36], [37]. Mao *et al.* [38] also proved that the cross-entropy loss function would affect the stability of GANs and could be replaced by the least-square loss function.

The standard GANs and these improvements are unsupervised generation models. The generated images have no category information. The Conditional-GAN (C-GAN) [39] is a supervised generation model, which adds labels as extra information to the input layers of the generator and the discriminator. Reed *et al.* [40] introduced labels information to both input and hidden layers. Odena *et al.* [41] designed an auxiliary classifier attached to the discriminator to strengthen the image-label matching.

Recently, many algorithms have been proposed to deploy the GANs for image recognition. The Categorical GANs (CatGAN) [42] achieved unsupervised and semi-supervised learning based on mutual information between the observed examples and their predicted categorical class distribution. The Improved-GAN [12] proposed by Salimans et al replaced the discriminator of the standard GANs with a $(K + 1)$ classes multi-classifier to recognize K classes of multiple targets. Li et al analyzed the problems in the Improved-GAN and designed the Triple-GAN [15] which worked with the C-GAN to control the category of the generated images. This model proposed adversarial training between the classifier and the discriminator to improve the recognition performance. Saatchi and Wilson [43] used stochastic gradient Hamiltonian Monte Carlo to marginalize the weights of the generator as well as the discriminator and achieve the state-of-the-art recognition results in most optical datasets.

III. METHODOLOGY

A. PREVIOUS MODELS

Standard GANs are formulated as a two-player minimax game which consists of a generator and a discriminator [11]. The discriminator is a binary classifier that identifies the real and generated images, while the generator tries to generate the images with a distribution similar to that of the real images to fool the discriminator. During the training process, the generator takes random noise $Z \sim p_z(Z)$ as input, and learns the distribution of the real images $X \sim p(X)$ to represent a mapping from the noise to the data space. The discriminator takes the real images $X \sim p(X)$ and the generated images $X_g \sim p_g(X)$ as input, and assigns a high or low score respectively. The generator tries to make the discriminator assign $X_g \sim p_g(X)$ a high score. The generator and the discriminator are both composed of neural networks. The loss function of the training process is a minimax problem as follows:

$$\min_G \max_D V(D,G) = \mathrm{E}_{x\sim p(x)}\left[\log\left(D(x)\right)\right] + \mathrm{E}_{z\sim p_z(z)}\left[\log\left(1-D\left(G(z)\right)\right)\right] \quad (1)$$

where $x \in X$, $z \in Z$. D and G denote the discriminator and the generator respectively. $D(x)$, $G(z)$ denote their output. $E_{x\sim p(x)}\left[\log\left(D(x)\right)\right]$ denotes the expectation of $\log\left(D(x)\right)$ when x follows the distribution $p(x)$. The parameters of D and G are updated in turn during the training process. Given G and its distribution p_g, optimal D is $D(x) = p(x)/\left(p(x)+p_g(x)\right)$. The gradient of D guides the updating of G. Iff $p(x) = p_g(x)$, the global optimum is achieved. At this point, the performance of D and G cannot be further improved. The discriminator cannot distinguish the two distributions p and p_g, i.e. $D(x) = D\left(G(z)\right) = 1/2$.

Several papers suggest that Jensen-Shannon (JS) divergence of the standard GANs causes instability during the learning process [36]–[38]. The instability can be summarized as two aspects. First, when the discriminator converges to optimality, $JS\left(p_{data}, p_g\right)$ is close to the constant $\log 2$ [36]. So the generator gets vanishing gradients and cannot be updated anymore. Another aspect is that the fake samples on the correct side of the decision boundary get vanishing gradients, even if they are still far away from the real data. Reference [38] introduces minimizing Pearson χ^2 divergence to address these problems. The least squares loss function penalizes the fake samples based on their distances to the decision boundary, which can generate more gradients when updating the generator. The objective functions of the Least Squares Generative Adversarial Networks (LSGANs) can be defined as follows:

$$\begin{aligned}\min_D V(D) &= \mathrm{E}_{x\sim p_{data}(x)}\left[\left(D(x)-1\right)^2\right] + \mathrm{E}_{z\sim p_z(z)}\left[\left(D\left(G(z)\right)\right)^2\right] \\ \min_G V(G) &= \mathrm{E}_{x\sim p_z(x)}\left[\left(D\left(G(z)\right)-1\right)^2\right] \end{aligned} \quad (2)$$

The C-GAN adds labels information to the generator and the discriminator [39]. The conditional generator takes noise $Z \sim p_z(Z)$ and the corresponding label $Y \sim p(Y)$ as input, and outputs the labeled generated images $\left(X_{cg}, Y_{cg}\right) \sim p_{cg}(X,Y)$. The conditional discriminator takes the labeled real images $(X_l, Y_l) \sim p(X,Y)$ as well as the labeled generated images $\left(X_{cg}, Y_{cg}\right) \sim p_{cg}(X,Y)$ as input, and assigns high and low scores respectively. The loss function of the training process is a minimax problem as follows:

$$\begin{aligned}&\min_G \max_D V(D,G) \\ &= \mathrm{E}_{(x,y)\sim p(x,y)}\left[\log\left(D(x,y)\right)\right] \\ &\quad + \mathrm{E}_{z\sim p_z(z),y\sim p(y)}\left[\log\left(1-D\left(G(z,y),y\right)\right)\right]\end{aligned} \quad (3)$$

The Triple-GAN [15] is a three-player semi-supervised image generation and recognition model, which consists of a conditional discriminator, a conditional generator and a classifier. The generator assists the training of the classifier. They play the minimax game with the discriminator respectively at the same time. The input and output of the generator are identical to those of the standard C-GAN. The classifier takes the unlabeled real images $X_u \sim p(X)$ as input and outputs the real images with pseudo labels $(X_u, Y_u) \sim p_c(X,Y)$ assigned to the discriminator. The input and output of the discriminator are composed of three parts. The first is to assign $(X_l, Y_l) \sim p(X,Y)$ a high score. The second is to assign $\left(X_{cg}, Y_{cg}\right) \sim p_{cg}(X,Y)$ a low score. The third is to assign $(X_u, Y_u) \sim p_c(X,Y)$ a low score. While the generator and the classifier try to fool the discriminator so that $\left(X_{cg}, Y_{cg}\right) \sim p_{cg}(X,Y)$ and $(X_u, Y_u) \sim p_c(X,Y)$ can get a high score. At the same time, the classifier is also trained with the cross entropy calculated from $(X_l, Y_l) \sim p(X,Y)$ and $\left(X_{cg}, Y_{cg}\right) \sim p_{cg}(X,Y)$. The loss function of the training process is a minimax problem as follows:

$$\begin{aligned}&\min_{C,G} \max_D V(C,D,G) \\ &= \Re_l + \alpha_{cg}\Re_{cg} + \mathrm{E}_{(x,y)\sim p(x,y)}\left[\log\left(D(x,y)\right)\right] \\ &\quad + \alpha\mathrm{E}_{(x,y)\sim p_c(x,y)}\left[\log\left(1-D(x,y)\right)\right] \\ &\quad + (1-\alpha)\,\mathrm{E}_{z\sim p_z(z),y\sim p(y)}\left[\log\left(1-D\left(G(z,y),y\right)\right)\right]\end{aligned} \quad (4)$$

where C denotes the classifier, $\Re_l$, $\Re_{cg}$ denote the cross-entropy loss of the classifier calculated from $(X_l, Y_l) \sim p(X,Y)$ and $\left(X_{cg}, Y_{cg}\right) \sim p_{cg}(X,Y)$ respectively, $\Re_l = \mathrm{E}_{(x,y)\sim p(x,y)}\left[-\log(p_c(y|x)\right]$, $\Re_{cg} = \mathrm{E}_{(x,y)\sim p_{cg}(x,y)}\left[-\log(p_c(y|x)\right]$. α controls the weight of $\left(X_{cg}, Y_{cg}\right) \sim p_{cg}(X,Y)$ and $(X_u, Y_u) \sim p_c(X,Y)$ to the discriminator. α_{cg} controls the weight of $\left(X_{cg}, Y_{cg}\right) \sim p_{cg}(X,Y)$ to the classifier. $\alpha, \alpha_{cg} \in (0,1)$ depend on experimental environments.

B. FIVE PLAYERS MODEL

In order to make full use of both labeled and unlabeled images, we propose the I-GAN model for semi-supervised SAR image generation and recognition. Our model is a

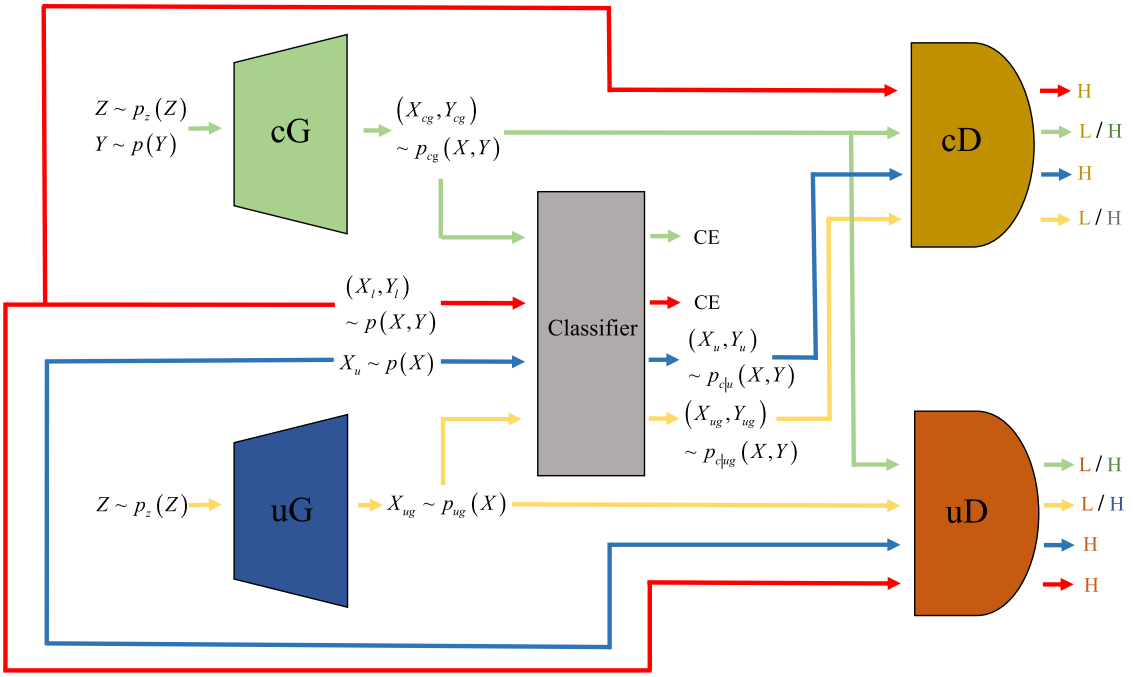

FIGURE 2. The proposed framework and the description of I-GAN. X, Y and Z denote the image, the label and the random noisy respectively. HC and CE mean the high confidence and the cross entropy respectively. H and L denote the high and low scores respectively. The color of H and L is consistent with the optimization goal of the corresponding generator, discriminator or classifier. The red, blue, green, and yellow paths respectively represent the processing of the labeled SAR image, the unlabeled SAR image, the labeled noise, and the unlabeled noise.

five-player model, whose detailed framework is shown in Fig. 2. The two generators try to fool the two discriminators at the same time, making them assign high score to the generated images. The classifier assists uG to fool cD in the game. The input and output of each player are described as follows:

- cG takes the random noise $Z \sim p_z(Z)$ and the corresponding label $Y \sim p(Y)$ as input, and outputs the labeled generated images $(X_{cg}, Y_{cg}) \sim p_{cg}(X,Y)$. The output will be utilized to fool cD and uD. In addition, we select half of $(X_{cg}, Y_{cg}) \sim p_{cg}(X,Y)$ which cD assigns a higher score than the others as $(X'_{cg}, Y'_{cg}) \sim p_{cg}(X,Y)$ to increase the labeled samples for classifier training.
- uG takes the random noise $Z \sim p_z(Z)$ as input, and outputs the unlabeled generated images $X_{ug} \sim p_{ug}(X)$. The output will be utilized to fool uD. Additionally, $X_{ug} \sim p_{ug}(X)$ will be given pseudo labels by the classifier to fool cD.
- The classifier is trained with the labeled images and then utilized to give pseudo labels to the unlabeled images. The labeled images are composed of $(X_l, Y_l) \sim p(X,Y)$ and $(X'_{cg}, Y'_{cg}) \sim p_{cg}(X,Y)$. The unlabeled images are composed of two parts. The first is the unlabeled real images $X_u \sim p(X)$. The classifier outputs the real images with pseudo labels $(X_u, Y_u) \sim p_{c|u}(X,Y)$, where $p_{c|u}(X,Y) = p_u(X)\, p_{c|u}(Y|X)$. We select half of $(X_u, Y_u) \sim p_{c|u}(X,Y)$ with a higher softmax confidence than the others as $(X'_u, Y'_u) \sim p_{c|u}(X,Y)$ to enrich the positive samples for cD. The second part is $X_{ug} \sim p_{ug}(X)$. They are given pseudo labels by the classifier as $(X_{ug}, Y_{ug}) \sim p_{c|ug}(X,Y)$, where $p_{c|ug}(X,Y) = p_{ug}(X)\, p_{c|ug}(Y|X)$. $(X_{ug}, Y_{ug}) \sim p_{c|ug}(X,Y)$ will be input to cD for adversarial training.
- cD aims at assigning a low score to $(X_{cg}, Y_{cg}) \sim p_{cg}(X,Y)$, $(X_{ug}, Y_{ug}) \sim p_{c|ug}(X,Y)$ and a high score to $(X_l, Y_l) \sim p(X,Y)$, $(X'_u, Y'_u) \sim p_{c|u}(X,Y)$.
- uD gives $(X_{cg}, Y_{cg}) \sim p_{cg}(X,Y)$ and $X_{ug} \sim p_{ug}(X)$ a low score, and gives $(X_l, Y_l) \sim p(X,Y)$ and $X_u \sim p(X)$ a high score.

We use the least-square loss function in the discriminators, where 1 and 0 denote high and low scores respectively. Considering that there are so many players in the I-GAN model, and there is no direct relationship between the two discriminators or the two generators, the minimax game does not have an explicit formulation. So, we show the loss functions of five players separately.

The loss functions of cD and uD can be defined as:

$$\begin{aligned}
&\min_{cD} V(cD) \\
&= \alpha \mathrm{E}_{(x,y)\sim p(x,y)}\left[(cD(x,y)-1)^2\right] \\
&\quad + (1-\alpha)\mathrm{E}_{(x,y)\sim p_{c|u}(x,y)}\left[(cD(x,y)-1)^2\right] \\
&\quad + \beta \mathrm{E}_{z\sim p_z(z), y\sim p(y)}\left[(cD(cG(z,y),y))^2\right] \\
&\quad + (1-\beta)\,\mathrm{E}_{z\sim p_z(z),(G(z),y)\sim p_{c|ug}(x,y)}\left[\left(cD(uG(z),y)^2\right)\right]
\end{aligned} \tag{5}$$

$$\begin{aligned}&\min_{uD} V(uD)\\&= \gamma \mathrm{E}_{(x,y)\sim p(x,y)}\left[(uD(x)-1)^2\right]\\&\quad+(1-\gamma)\mathrm{E}_{x\sim p(x)}\left[(uD(x)-1)^2\right]\\&\quad+(1-\beta)\,\mathrm{E}_{z\sim p_z(z),y\sim p(y)}\left[(uD(cG(z,y)))^2\right]\\&\quad+\beta\mathrm{E}_{z\sim p_z(z)}\left[(uD(uG(z)))^2\right]\end{aligned}\tag{6}$$

where $\mathrm{E}_{(x,y)\sim p(x,y)}\left[(cD(x,y)-1)^2\right]$ and $\mathrm{E}_{(x,y)\sim p_{c|u}(x,y)}\left[(cD(x,y)-1)^2\right]$ represent that cD assigns a high score to $(X_l,Y_l)\sim p(X,Y)$ and $\left(X'_u,Y'_u\right)\sim p_{c|u}(X,Y)$ respectively. α and $1-\alpha$ controls the weight of $(X_l,Y_l)\sim p(X,Y)$ and $\left(X'_u,Y'_u\right)\sim p_{c|u}(X,Y)$. $\alpha\in[0.5,1]$ indicates that original labeled samples is more important. $\mathrm{E}_{z\sim p_z(z),y\sim p(y)}\left[(cD(cG(z,y),y))^2\right]$ and $\mathrm{E}_{z\sim p_z(z)}\left[(uD(uG(z)))^2\right]$ mean that cD gives $\left(X_{cg},Y_{cg}\right)\sim p_{cg}(X,Y)$ and $\left(X_{ug},Y_{ug}\right)\sim p_{c|ug}(X,Y)$ a low score respectively. β and $1-\beta$ control the autocorrelation and cross-correlation coefficient of the conditional GANs and the unconditional GANs. In order to enhance the reliability of image-label pairs, cD should pay more attention to cG and uD should pay more attention to uG, so $\beta\in[0.5,1]$. Both α and β depend on experimental environments. $\mathrm{E}_{(x,y)\sim p(x,y)}\left[(uD(x)-1)^2\right]$ and $\mathrm{E}_{x\sim p(x)}\left[(uD(x)-1)^2\right]$ indicate that uD assigns a high score to $(X_l,Y_l)\sim p(X,Y)$ and $X_u\sim p(X)$ respectively. $\gamma\in[0,1]$ controls the weight of $(X_l,Y_l)\sim p(X,Y)$ and $X_u\sim p(X)$. It is obvious that they are equally important to uD. Assuming that the number of them is N_l,N_u respectively, then $\gamma=N_l/(N_l+N_u)$. $\mathrm{E}_{z\sim p_z(z),y\sim p(y)}\left[(uD(cG(z,y)))^2\right]$ and $\mathrm{E}_{z\sim p_z(z)}\left[(uD(uG(z)))^2\right]$ mean that uD gives $\left(X_{cg},Y_{cg}\right)\sim p_{cg}(X,Y)$ and $X_{ug}\sim p_{ug}(X)$ a low score respectively.

Accordingly, the loss functions of cG and uG can be defined as:

$$\begin{aligned}&\min_{cG} V(cG)\\&=\beta\mathrm{E}_{z\sim p_z(z),y\sim p(y)}\left[(cD(cG(z,y),y)-1)^2\right]\\&\quad+(1-\beta)\,\mathrm{E}_{z\sim p_z(z),y\sim p(y)}\left[(uD(cG(z,y))-1)^2\right]\end{aligned}\tag{7}$$

$$\begin{aligned}&\min_{uG} V(uG)\\&=(1-\beta)\,\mathrm{E}_{z\sim p_z(z),(G(z),y)\sim p_{c|ug}(x,y)}\left[(cD(uG(z),y)-1)^2\right]\\&\quad+\beta\mathrm{E}_{z\sim p_z(z)}\left[(uD(uG(z))-1)^2\right]\end{aligned}\tag{8}$$

These loss functions suggest that the generators try to fool the discriminators to obtain a high score. $\mathrm{E}_{z\sim p_z(z),y\sim p(y)}\left[(cD(cG(z,y),y)-1)^2\right]$ and $\mathrm{E}_{z\sim p_z(z),y\sim p(y)}\left[(uD(cG(z,y))-1)^2\right]$ represent that cG aims to make cD and uD assign a high score to $\left(X_{cg},Y_{cg}\right)\sim p_{cg}(X,Y)$ respectively. $\mathrm{E}_{z\sim p_z(z),(G(z),y)\sim p_{c|ug}(x,y)}\left[(cD(uG(z),y)-1)^2\right]$ and $\mathrm{E}_{z\sim p_z(z)}\left[(uD(uG(z))-1)^2\right]$ mean that uG tries to make cD and uD give $\left(X_{ug},Y_{ug}\right)\sim p_{c|ug}(X,Y)$ and $X_{ug}\sim p_{ug}(X)$ a high score respectively.

The classifier adopts a cross-entropy loss function for the labeled images. For the unlabeled images, we apply the adversarial loss to $X_{ug}\sim p_{ug}(X)$ instead of $X_u\sim p(X)$ to avoid the problem where correct image-label pairs are regarded as negative samples by cD. The loss functions of the classifier can be defined as:

$$\begin{aligned}\min_{C} V(C) &= \Re_l+\alpha_{cg}\Re_{cg}\\&\quad+k\,(1-\beta)\,\mathrm{E}_{z\sim p_z(z),(G(z),y)\sim p_{c|ug}(x,y)}\\&\quad\times\left[(cD(uG(z),y)-1)^2\right]\end{aligned}\tag{9}$$

where $\Re_l=\mathrm{E}_{(x,y)\sim p(x,y)}\left[-\log(p_c(y|x)\right]$ and $\Re_{cg}=\mathrm{E}_{(x,y)\sim p_{cg}(x,y)}\left[-\log(p_c(y|x)\right]$ denote the cross-entropy loss calculated by $(X_l,Y_l)\sim p(X,Y)$ and $\left(X'_{cg},Y'_{cg}\right)\sim p_{cg}(X,Y)$ respectively. $\alpha_{cg}\in[0,1]$ controls the importance of $\left(X'_{cg},Y'_{cg}\right)\sim p_{cg}(X,Y)$ relative to $(X_l,Y_l)\sim p(X,Y)$. $\mathrm{E}_{z\sim p_z(z),(G(z),y)\sim p_{c|ug}(x,y)}\left[(cD(uG(z),y)-1)^2\right]$ is the adversarial loss, which represent that the classifier aims to make cD assign $\left(X_{ug},Y_{ug}\right)\sim p_{c|ug}(X,Y)$ a high score.k denotes the attenuation of the adversarial loss to least-square loss. Both α_{cg} and k depend on experimental environments.

C. PROPOSED ARCHITECTURE

In this section, we introduce the specific architecture of each player and several techniques used in I-GAN, where the size of the image is 64×64. Fig. 3(a) shows the architecture of the discriminators. cD and uD have similar architectures. They consist of three convolution layers and two full connected layers. The input is a $64\times64\times1$ grayscale image. The output is the score assigned by the discriminator. The kernel size, channel number, convolution stride and activation function are illustrated in Fig. 3(a). The difference between cD and uD is that only cD adds label information to each layer. Fig. 3(b) shows the architecture of the generators. cG and uG have similar architectures. They consist of one full connected layer and four deconvolution layers. The input is 100-dimensional Gaussian noise. After the full connected layer, the input is reshaped to make up a $4\times4\times512$ feature map. The difference between cG and uG is that only cG adds label information to each layer. Fig. 3(c) shows the architecture of the classifier, which consists of three convolution layers and three full connected layers.

In order to improve the stability of image generation and the accuracy of recognition, several practical techniques are used in our algorithm. The dropout is added to the hidden layer of the discriminators, the generators and the classifier to alleviate over-fitting. The instance normalization is used in the convolution layer of the discriminators and the generators. The batch normalization is used in the fully connected layer of the classifier and the generators. The Gaussian noise is added to the input of the discriminators and the classifier to enhance system robustness. The learning rate of the discriminators, the generators and the classifier decreases exponentially in each epoch.

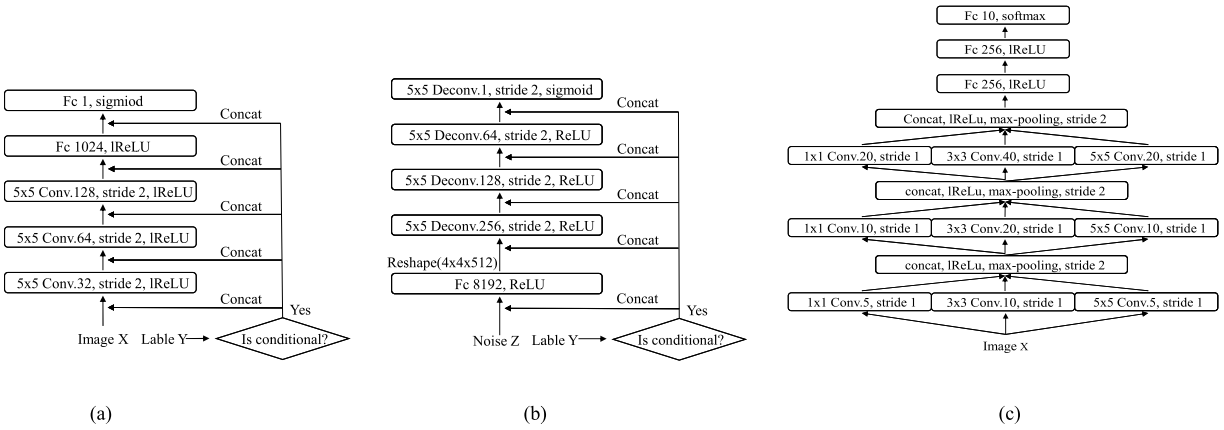

FIGURE 3. The architecture of the discriminators, the generators and the classifier. (a) cD and uD; (b) cG and uD; and (c) The classifier. Conv, Deconv, and Fc denote 2Dconvolution layer, 2Ddeconvolution layer, and fully connected layer respectively.

Algorithm 1 shows the whole procedure of training the I-GAN and the iteration of the semi-supervised network, where θ_{cd}, θ_{ud}, θ_{cg}, θ_{ug}, θ_c denote the trainable parameters of cD, uD, cG, uG, and the classifier respectively.

Algorithm 1 Training of the I-GAN

for number of training iterations do

1. Sample a mini-batch of $z \sim p_z(z)$ and $y \sim p(y)$
2. cG outputs $(x_{cg}, y_{cg}) \sim p_{cg}(x, y)$, and uG outputs $x_{ug} \sim p_{ug}(x)$
3. Sample a batch of $x_u \sim p(x)$
4. The classifier inputs $x_u \sim p(x)$ and $x_{ug} \sim p_{ug}(x)$, and outputs $(x_u, y_u) \sim p_{c|u}(x, y)$ and $(x_{ug}, y_{ug}) \sim p_{c|ug}(x, y)$
5. Select half of $(x_{cg}, y_{cg}) \sim p_{cg}(x, y)$ and $(x_u, y_u) \sim p_{c|u}(x, y)$ with high confidence by cD and the classifier respectively
6. Sample a batch of $(x_l, y_l) \sim p(x, y)$
7. Compute the loss functions and the gradient
8. Update cD, uD, cG, uG, and the classifier in order along their stochastic gradient:

$$\nabla_{\theta_{cd}} V(cD), \nabla_{\theta_{ud}} V(D), \nabla_{\theta_{cg}} V(cG), \nabla_{\theta_{ug}} V(G), \nabla_{\theta_c} V(C)$$

end for

IV. EXPERIMENTAL SET-UP

The MSTAR dataset is derived from the Defense Advanced Research Project Agency (DARPA) and the Air Force Research Laboratory (AFRL), which contains 0.3m*0.3m resolution's SAR images obtained from X-band, HH-polarized and bunching mode sensors [44]. Ten classes of vehicle targets are used in our experiment: 2S1, ZSU234, BMP2, BRDM2, BTR60, BTR70, D7, ZIL131, T62, and T72. These targets are widely used in SAR image recognition. Their SAR images and the corresponding optical images are shown in Fig. 4.

In the experiment, we extract the areas of 64 × 64 pixels in the center of the original images for training and recognition tasks. The training set contains 2747 SAR targets under a 17° depression angle and the test set contains 2425 SAR targets under a 15° depression angle. The generalization ability of the model can be verified by using different depression angles in the training and testing sets. Table 1 lists the details of each class.

TABLE 1. Details of our training set and test set.

Class	Serial	Training Set		Testing set	
		Depression	Number	Depression	Number
2S1	B_01	17°	299	15°	274
ZSU234	D_08		299		274
BRDM2	E_71	17°	298	15°	274
BTR60	K10YT_7532		256		195
BMP2	SN_9563		233		195
BTR70	C_71		233		196
D7	92V_13015		299		274
ZIL131	E_12		299		274
T62	A_51	17°	299	15°	273
T72	#A64		232		196
Sum			2747		2425

TABLE 2. Partitions of the training set.

Labeled number	Unlabeled number	Labeled rate	Labeled batchsize	Unlabeled batchsize
100	2647	3.6%	25	200
200	2547	7.3%	40	160
400	2347	14.6%	50	150
600	2147	21.8%	75	150

In our semi-supervised experiment, we choose part of images in the training set as the labeled data and the rest in the training set as the unlabeled data. In order to verify the generation and recognition performance of the I-GAN under different numbers of labeled samples, we select 10, 20, 40 and 60 SAR images per class as labeled data respectively. Table 2 shows the partitions of the training set. It should be noted that we only perform one random selection in all the experiments based on the same number of the labeled SAR images, which means that the same labeled SAR images are used in all the experiments. It can reduce the influence of the training set's partitions on the experimental results.

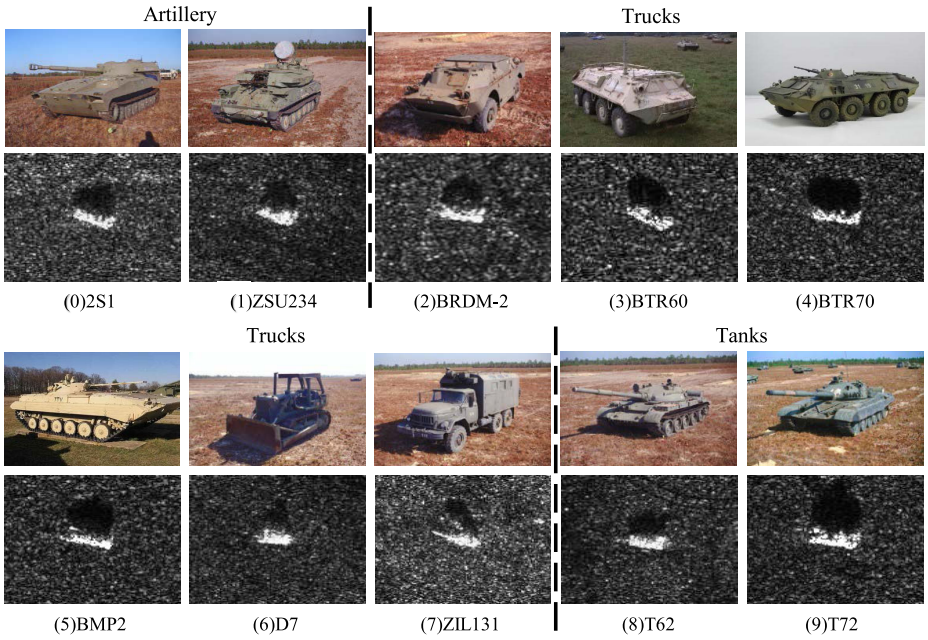

FIGURE 4. Optical images and SAR images of 10 classes of vehicle objects in the MSTAR database.

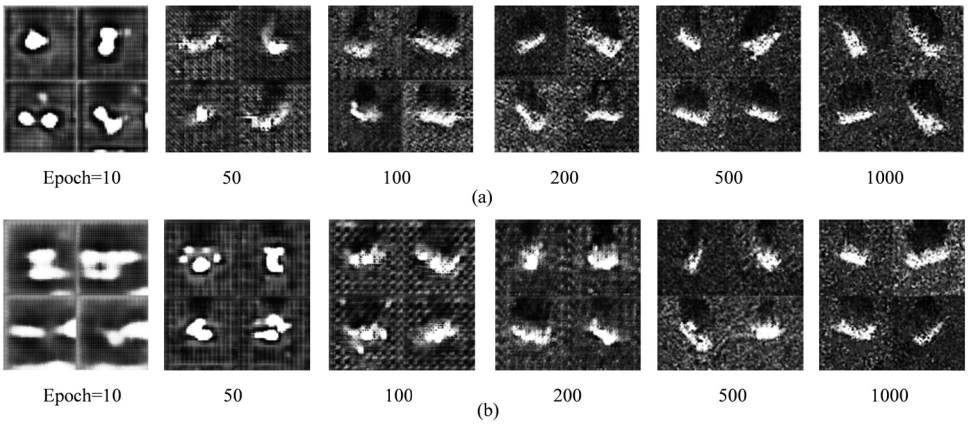

FIGURE 5. The generated images from the conditional generator with the increase of the training time: (a) 600 labeled samples; (b) 100 labeled samples.

Our main configuration of the computer is: CPU: Intel(R) Xeon(R) E5504, GPU: GTX1070, operating system: Ubuntu 16.04, running software: Python 2.7, Tensorflow 1.1.0.

V. RESULTS AND DISCUSSIONS

A. IMAGE GENERATION

In the I-GAN model, we design two generators with different functions. Only cG can generate fake SAR images based on different categories. Because the labeled samples are far fewer than the unlabeled samples in our experiments, training cG is difficult. The fake SAR images generated by cG in the training process are shown in Fig. 5, where the numbers of the labeled SAR images are 600 and 100 respectively. It can be seen that before the 500th epoch, the quality of the generated images gradually improves with the training proceeding, and cG with more labeled samples has better performance. In the last 500 epochs, it is difficult for humans to distinguish the difference between the real and generated images. We only show the generating process of these two sample sizes because they use the most and the least labeled samples, corresponding to the best and the worst experimental results respectively.

In order to verify that our cG can control the categories of the generated images, Fig. 6 shows the generated images class by class in the 1000th epoch with 600 labeled samples.

Classical evaluation of the SAR image generation is often based on subjective evaluation. As the commonly used metrics, Inception score (IS) and Fréchet Inception distance (FID) rely on the inception network pretrained using the ImageNet dataset. They are not appropriate to evaluate

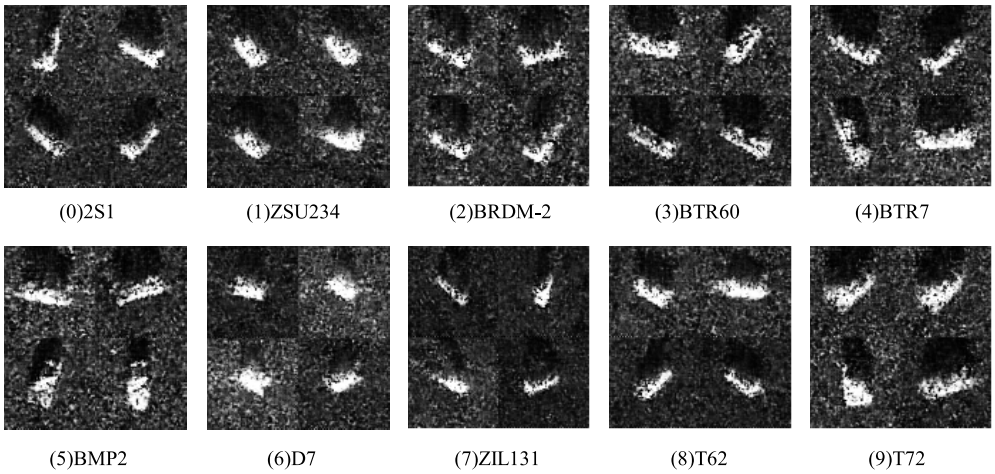

FIGURE 6. The generated images of 10 classes in the 1000th epoch.

the quality of SAR image generation. In addition, the method of adjusting the image size by bilinear interpolation in IS and FID may damage the scattering characteristics of the SAR images. Because of the limitations of IS and FID methods, we utilize the GAN-train and the GAN-test proposed in [45] to evaluate the quality of the generated images. To introduce the GAN-train and the GAN-test, we first define the dataset composed of the generated images as Dg, the SAR image dataset for training the GANs as Dt, and the validation set of the SAR images as Dv. Dt and Dv are identical to the training and the testing sets presented in Section IV respectively. The GAN-train and the GAN-test need an extra classifier as a rater. To distinguish it from the classifier used in the I-GAN, its architecture is shown in Fig. 7. The GAN-train is the classification outcome trained on Dg and tested on Dv. The GAN-test is the classification outcome trained on Dt and tested on Dg. [45] reports that GAN-test evaluates the similarity between the distributions of the generated image-label pairs and those of the real image-label pairs. The GAN-train mainly evaluates the diversity of the generated images, but the difference between the distributions of the generated image-label pairs and those of the real image-label pairs can also affect the GAN-train. Since Dg and Dt have the same distribution's expectation, the GAN-test is straightforward. If the GAN-test is significantly low, the GANs does not capture the features in Dt properly. If the GAN-test is significantly high while the GAN-train is significantly low, the mode collapse problem occurs in the GANs training. In addition, the GAN-train is related to the size of Dg. When the size of Dg is small, the GAN-train will be low and significantly influenced by the randomness of the generated data. As the size of Dg increases, the GAN-train gradually increases and converges. Therefore, the size of Dg should be large enough in the evaluation. To determine the appropriate size of Dg on the MSTAR dataset, the relationship between the size of Dg and the GAN-train of the I-GAN is shown in Fig. 8. It seems that the GAN-train converges when the number of the generated images in Dg exceeds 3000. Hence, our following experiment is based on Dg composed of 9000 generated images to ensure that the GAN-train is in the convergence region. In addition, the number of the generated images in each class is equal in our experiments.

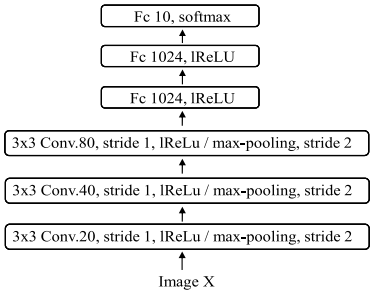

FIGURE 7. The extra classifier used in GAN-train and GAN-test.

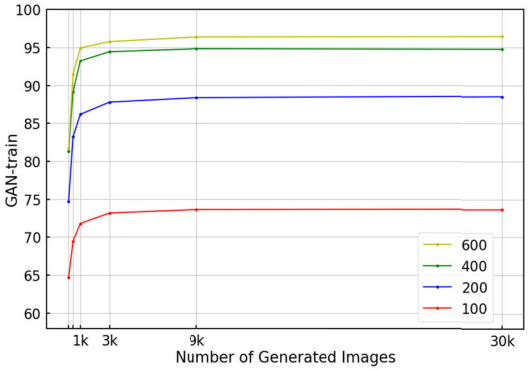

FIGURE 8. The GAN-train of the I-GAN under different sizes of Dg. Different curves represent the I-GAN trained by different number of labeled SAR images.

Table 3 shows the GAN-train and the GAN-test of the C-GAN, the AC-GAN, the Triple-GAN and our method. It should be emphasized that although Dt contains 2747 SAR

TABLE 3. The GAN-train and the GAN-test of the C-GAN, the AC-GAN, the Triple-GAN, and our method under different partitions of the training dataset.

Labeled number	GAN-train				GAN-test			
	C-GAN	AC-GAN	Triple-GAN	Ours	C-GAN	AC-GAN	Triple-GAN	Ours
100	45.33	50.63	41.76	**73.61**	99.79	99.88	**99.93**	98.53
200	69.27	70.87	67.03	**88.36**	99.69	99.80	**99.92**	99.65
400	85.92	87.21	84.65	**94.83**	99.81	99.85	**99.94**	99.89
600	89.12	89.41	89.16	**96.39**	99.77	99.82	**99.96**	99.93

images, only part of the samples are labeled. We have illustrated the partitions of the training set in Table 2. As supervised generation models, C-GAN and AC-GAN are trained only with the labeled samples. The Triple-GAN and our method are trained with both labeled and unlabeled samples. The training parameters and the architecture of the classifier, the discriminator as well as the generator used in all methods are the same. In addition, all of the methods are based on least-square loss function.

In the experiment with 100 labeled SAR images, the GAN-test of our method is 98.53%, but the GAN-train is 73.61%, far higher than the other methods. The generated images of I-GAN have better diversity because the unsupervised GANs and co-training model help the feature extraction performance under a small number of the labeled samples. At the same time, as a few incorrect image-label pairs are regarded as the positive samples by cD in the co-training model, cG outputs a small number of SAR images with wrong labels, which influences the GAN-test. Although the GAN-test of the other methods is higher, their GAN-train is far less than the proposed, which means the diversity of the generated images is quite low and mode collapse occurs. As the number of the labeled samples increases, the GAN-test of our method increases rapidly. That is because the pseudo label given by the classifier becomes more accurate using the co-training model. While the C-GAN and Triple-GAN only rely on the labeled images as positive samples, their GAN-test do not increase significantly. In all the experiments, the GAN-test is generally high, because the DCNN architecture and the least-square loss function can ensure the stability in the training process. The GAN-train of our method is prominently higher than those of C-GAN, AC-GAN and Triple-GAN, as the semi-supervised generation method learns the features of the unlabeled SAR images and increases the diversity of the generated images. Although the Triple-GAN is also a semi-supervised generation model, its GAN-train tends to be lower than that of the C-GAN when the number of the labeled samples decreases. This is because only the labeled SAR images are treated as positive samples in the Triple-GAN model, and some correct image-label pairs regarded as negative samples by the discriminator, resulting in poor results. Therefore, the method proposed in Triple-GAN has little help to the diversity of the generated images, while our semi-supervised generation method can better capture the features of the unlabeled SAR images. The classifier of the I-GAN is trained with both of the generated images and the real images in our co-training model, and higher GAN-train can directly enhance the final recognition performance.

As described in this paper, generation of better images in I-GAN are mainly attributed to two semi-supervised learning strategies compared with the Triple-GAN. To begin with, we add unconditional auxiliary GANs so that cG can learn features from unlabeled samples. Due to the unconditional GANs, we solve the problem of many correct image-label pairs being treated as negative samples by cD. This is due the discriminators only assigning a low score to the generated images in our model. The second improvement is the co-training between conditional GANs and the classifier. We select high-confidence unlabeled SAR images to form positive samples of the conditional discriminator. Since we increase the positive samples for the conditional GANs, our semi-supervised method can achieve higher GAN-train with insufficient labeled samples. Apart from the two strategies, we replace the cross-entropy loss function with the least-square loss function. We implement the ablation experiments with 100 and 200 labeled samples because the difference is more prominent as the labeled samples are fewer. Table 4 shows the GAN-train details and verifies the effectiveness of the improvement aspects. T, L, U, C, and Imp denote Triple-GAN, least-square loss, unconditional GANs, co-training, and improvement respectively. As can be seen, although it has been proved in [38] that least-square loss function is better than cross-entropy loss function, the performance of T + L is also limited by insufficient labeled samples. Since our two semi-supervised learning methods can utilize the unlabeled samples more effectively, their superiority is more significant. From the results of the ablation experiments, the unconditional GANs can be seen to address the problems in Triple-GAN. The co-training is successful to make the GAN-train achieve substantial outcomes.

It should be noted that due to the limitation of the number of the labeled samples and the feature extraction ability of the GANs, the quality of our generated image may still be inferior to that of the real SAR images. However, the experiment for the GAN-train and the GAN-test indicates that our generated images can fool a common DCNN classifier significantly. Under 600 labeled samples, we can achieve 96% recognition

TABLE 4. The GAN-train in ablation experiments under 100 and 200 labeled samples.

Target	Labeled number = 100							Labeled number = 200						
	T	T + L		T + L + U		T + L+ U + C		T	T + L		T + L + U		T + L+ U + C	
	GAN-train	GAN-train	Imp	GAN-train	Imp	GAN-train	Imp	GAN-train	GAN-train	Imp	GAN-train	Imp	GAN-train	Imp
2S1	55.78	52.43	-3.35	46.35	-9.43	80.9	25.12	74.64	73.12	-1.52	70.01	-4.63	89.23	14.59
ZSU234	75.79	80.66	4.87	69.53	-6.26	99.34	23.55	82.6	85.1	2.50	76.52	-6.08	99.03	16.43
BRDM2	24.76	24.88	0.12	32.06	7.3	63.14	38.38	47.02	43.49	-3.53	63.14	16.12	84.43	37.41
BTR60	32.99	32.22	-0.77	40.17	7.18	56.32	23.33	54.96	53.93	-1.03	56.11	1.15	75.73	20.77
BMP2	26.33	32.82	6.49	39.57	13.24	47.78	21.45	32.74	34.27	1.53	45.73	12.99	60.34	27.6
BTR70	41.59	39.89	-1.70	52.72	11.13	80.78	39.19	61.14	62.67	1.53	67.01	5.87	91.5	30.36
D7	49.03	49.33	0.30	50.91	1.88	74.7	25.67	92.76	96.66	3.90	97.87	5.11	98.91	6.15
ZIL131	21.72	23.97	2.25	45.68	23.96	62.16	40.44	57.91	58.09	0.18	62.47	4.56	91.97	34.06
T62	33.33	28.14	-5.19	43.04	9.71	71.86	38.53	63.98	69.6	5.62	73.87	9.89	87.91	23.93
T72	50.17	49.58	-0.59	61.99	11.82	94.73	44.56	89.63	83.67	-5.96	86.65	-2.98	95.84	6.21
Average	41.15	41.39	0.24	48.2	7.05	73.17	32.02	65.74	66.06	0.32	69.94	4.2	87.49	21.75
Overall	41.6	41.76	0.16	48.16	6.56	73.61	32.01	66.54	67.03	0.49	70.6	4.06	88.36	21.82

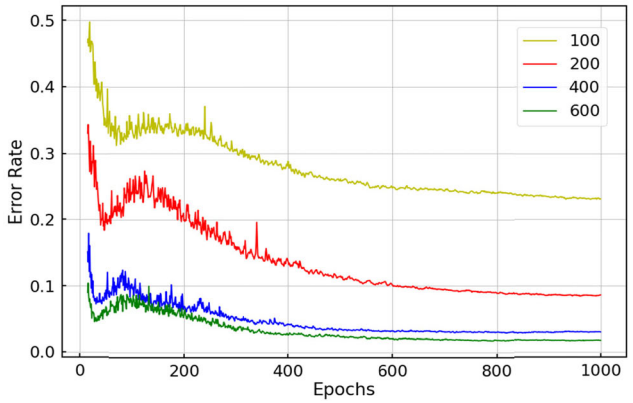

FIGURE 9. Error rate curves of I-GAN trained with different numbers of the labeled SAR images.

accuracy for real SAR images using the classifier only trained with the generated images, and 99% recognition accuracy for the generated images using the classifier only trained with the real SAR images. Even if our generated images may not reach the level of the real SAR images, we can still use them as a supportive tool to train the classifier. In this case, our generated images are more practical than the traditional model. Considering that the quality of the generated images may be lower than that of real images, and that there are a small number of image-label pairs mismatched in the process of training, instead of directly sending $(X_{cg}, Y_{cg}) \sim p_{cg}(X, Y)$ and $(X_l, Y_l) \sim p(X, Y)$ into the classifier, we adopt three measures: (1) We only select half of $(X_{cg}, Y_{cg}) \sim p_{cg}(X, Y)$ which cD assigns a higher score than the others to the classifier; (2) We ensure the importance of $\left(X'_{cg}, Y'_{cg}\right) \sim p_{cg}(X, Y)$ relative to $(X_l, Y_l) \sim p(X, Y)$ through the hyperparameter α_{cg} in the loss function of the classifier; (3) We use different Gaussian random noise and labels to generate new images in each iteration, so every wrong image-label pair can only be used at most once.

B. RECOGNITION RESULTS

In this section, we first show the error rate curves of the I-GAN under different numbers of the labeled SAR images in Fig. 9 to observe the learning process of the classifier in I-GAN model. Each curve is averaged by 10 times training. To make the curve convenient to be observed, we only show the epochs from the 15th to 1000th.

We compare the recognition accuracy of the I-GAN and the Triple-GAN [15] to verify the effectiveness of our proposed method. For fair comparison, the network's architecture and the training parameters used in Triple-GAN are the same as those of our method. The average recognition accuracy and the features visualization by T-SNE [46] are shown in Table 5 and Fig. 10 respectively.

To further analyze the performance of the two models, we calculate the Kappa coefficient [47] and the training time, which are shown in Table 6. The Kappa coefficient is generally used to measure the consistency between the model results and the actual results. It is defined as:

$$\kappa = \frac{p_0 - p_e}{1 - p_e} \tag{10}$$

where p_0 denotes the relative observed agreement between the output of the classifier and the real label. p_e is the hypothetical probability of the chance agreement. $\kappa \in [-1, 1]$, the closer κ is to 1, the better the agreement between the model and the actual results is.

It can be seen that the Average accuracy, Overall accuracy and Kappa coefficients of the I-GAN are higher than those of Triple-GAN in all the experiments. That is because in the Triple-GAN model, the conditional GANs helps to train the classifier, but the classifier is not helpful to the conditional GANs. However, co-training model is used for linking the classifier with conditional GANs in our method. In addition, the correct image-label pairs regarded as negative samples exacerbates the mode collapse problem in the Triple-GAN model, while our cD only assigns a low score to the generated images with the help of the unconditional GANs. So, the quality of the generated images in our method is higher than that of the Triple-GAN, which makes the classifier to be of better performance. As to time consumption, since the framework of the I-GAN is more complex than that of the Triple-GAN, the I-GAN needs more training time in all the experiments.

We also compare the recognition performance of our method with that of the other semi-supervised methods,

TABLE 5. The recognition accuracy (%) of the Triple-GAN and our method under different partitions of the training dataset.

Labeled number	100		200		400		600	
method	Triple-GAN	I-GAN	Triple-GAN	I-GAN	Triple-GAN	I-GAN	Triple-GAN	I-GAN
2S1	68.50 ±2.94	73.39 ±3.83	86.75 ±2.54	93.18 ±1.15	95.59 ±1.20	98.69 ±0.49	97.23 ±0.62	98.32 ±0.79
ZSU234	98.76 ±0.43	99.24 ±0.36	98.98 ±0.45	99.53 ±0.30	99.86 ±0.19	99.96 ±0.11	99.96 ±0.11	100.00 ±0
BRDM2	58.90 ±4.00	71.42 ±2.40	76.06 ±3.04	82.63 ±2.18	93.91 ±0.64	94.09 ±1.15	96.32 ±1.07	96.13 ±1.14
BTR60	56.16 ±1.44	62.72 ±1.71	76.15 ±2.47	83.23 ±1.41	88.72 ±1.76	91.64 ±1.33	92.00 ±1.31	95.64 ±1.14
BMP2	58.41 ±3.47	54.36 ±5.25	68.87 ±2.49	79.23 ±2.35	82.87 ±2.51	86.10 ±1.83	88.98 ±2.10	93.95 ±2.17
BTR70	64.75 ±4.62	77.19 ±3.09	85.25 ±1.93	90.10 ±4.51	97.91 ±0.66	99.59 ±0.32	99.44 ±0.38	99.95 ±0.16
D7	75.04 ±1.35	82.26 ±1.68	98.91 ±0.30	99.05 ±0.35	98.83 ±0.57	99.20 ±0.15	98.69 ±0.60	98.94 ±0.50
ZIL131	76.35 ±1.69	70.58 ±2.9	80.11 ±2.94	91.06 ±3.01	98.76 ±0.73	99.64 ±0	98.87 ±0.32	99.57 ±0.23
T62	58.03 ±3.93	79.63 ±5.52	82.60 ±3.61	92.12 ±2.27	95.02 ±0.78	98.35 ±0.81	96.99 ±1.56	99.41 ±0.39
T72	80.51 ±3.41	91.33 ±2.70	97.81 ±1.05	100.00 ±0	99.29 ±0.43	99.95 ±0.16	100.00 ±0	100.00 ±0
Average	69.54 ±1.32	**76.21 ±1.35**	85.15 ±0.71	**91.01 ±0.69**	95.07 ±0.39	96.72 ±0.37	96.85 ±0.26	**98.19 ±0.34**
Overall	70.14 ±1.33	**76.85 ±1.29**	85.56 ±0.76	**91.39 ±0.64**	95.45 ±0.33	97.03 ±0.34	97.08 ±0.26	**98.30 ±0.31**

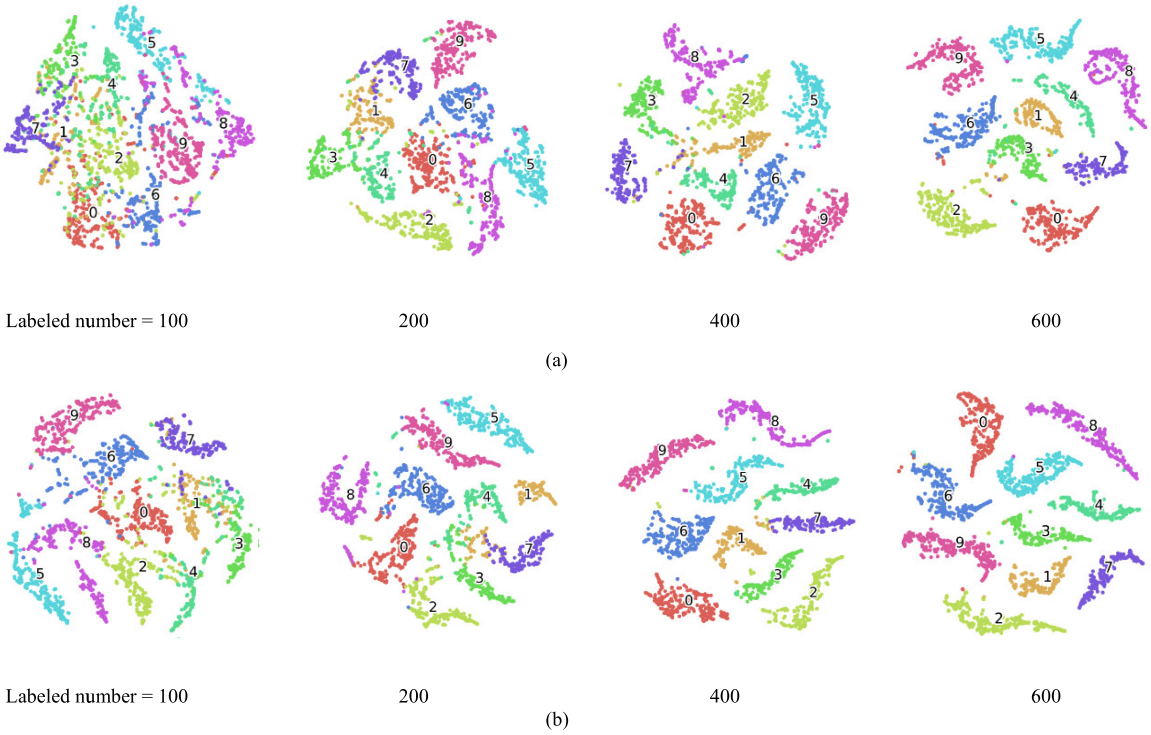

FIGURE 10. Features generated by t-SNE: (a) Triple-GAN (b) I-GAN. Different colors denote different categories. The number of the labeled samples is attached to each picture.

including the Mean Teacher [48], the Virtual Adversarial Training (VAT) [49], the Improved-GAN [12], and the Bayesian-GAN [43]. The Mean Teacher takes the moving average of the student classifier's parameters as the teacher classifier's parameters, and adds the difference between the outputs of the two classifiers into the loss function. The VAT adopts the idea of adversarial training, and requires the classifier to make the same prediction before and after applying adversarial noise. The Improved-GAN changes the architecture of the discriminator used in the standard GANs, which is applied to the recognition task. The Bayesian-GAN proposes a practical Bayesian formula and applies the GANs to end-to-end unsupervised learning and semi-supervised learning. The architecture of the classifier in the traditional semi-supervised learning method is the same as that used in our method. The Adam optimizer is adopted in the classifier, where the

TABLE 6. The Kappa coefficient and the training time of the Triple-GAN method and our method.

Labeled number	Kappa coefficient		Training time (s / epoch)	
	Triple-GAN	I-GAN	Triple-GAN	I-GAN
100	0.668 ±0.015	**0.742 ±0.014**	**1.58**	2.19
200	0.839 ±0.008	**0.904 ±0.007**	**1.92**	2.83
400	0.949 ±0.004	**0.967 ±0.004**	**3.06**	4.67
600	0.967 ±0.003	**0.981 ±0.003**	**3.63**	5.73

TABLE 7. The recognition accuracy (%) of the Mean Teacher, the VAT, the Improved-GAN, the Bayesian-GAN, and our method with different numbers of the labeled SAR images.

Method	Number of labeled SAR images			
	n = 100	n = 200	n = 400	n = 600
Mean Teacher	66.94 ±1.77	80.01 ±1.62	91.78 ±1.07	94.96 ±0.70
VAT	69.05 ±1.21	83.78 ±0.79	93.71 ±0.71	95.02 ±0.65
Improved-GAN	57.52 ±2.22	74.56 ±1.77	89.07 ±1.10	92.49 ±0.68
Bayesian-GAN	72.08 ±2.31	85.90 ±1.41	96.47 ±0.95	97.78 ±0.34
Ours	76.85 ±1.29	91.39 ±0.64	97.03 ±0.34	98.30 ±0.31

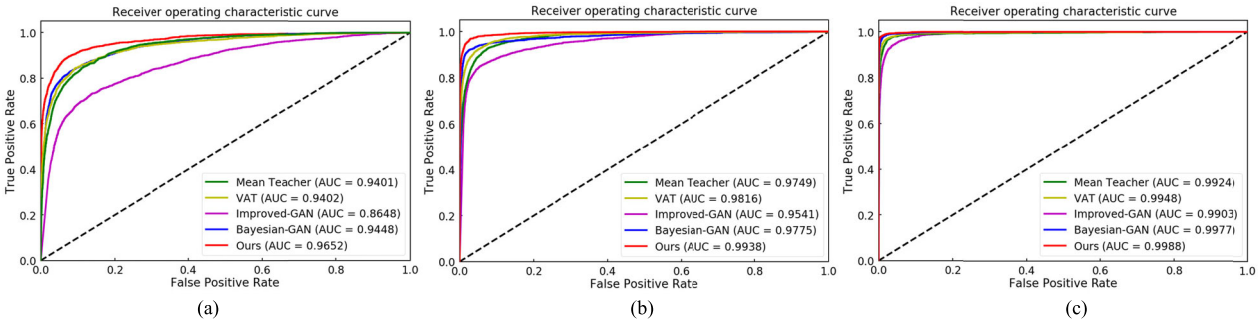

FIGURE 11. ROC curves of Mean Teacher, VAT, Improved-GAN, Bayesian-GAN, and our method. (a) 100 labeled samples; (b) 200 labeled samples; and (c) 400 labeled samples.

learning rate is initialized at 0.001 and decreases exponentially. The architecture of the generator and the discriminator of the method based on GANs are the same as that of our method. The Adam optimizer is adopted for the generator and the discriminator, where the learning rate is initialized at 0.0002 and decreases exponentially. The recognition accuracy of each method under different sample sizes is shown in Table 7. Each experiment is repeated for 10 times.

As can be seen in Table 7, the recognition accuracy of all the methods is improved with the number of the labeled SAR images increasing. The recognition performance of our method is better than that of the other methods. The fewer labeled SAR images, the more significant difference of the recognition performance. Compared with the traditional semi-supervised method, the GANs-based method can learn the features of the unlabeled samples and generate more data to train the networks. We analyze and address the problems in the traditional GANs model, thus the I-GAN achieves higher recognition accuracy. In the experiment with 600 labeled samples, the recognition accuracy of our method can achieve 98.30%, close to the level of some supervised methods which use all 2747 labeled SAR images. That is because our I-GAN model can generate high-quality images to train the classifier through the semi-supervised learning. In the experiment with only 100 labeled samples, it is a challenge to train and obtain the best network parameters. The over-fitting of the supervised learning is severe, and the performance of many semi-supervised methods degrades rapidly due to the insufficient utilization with the unlabeled SAR images. Our method can still reach 76.85% under this condition, while the Improved-GAN is only of 57.52%. This indicates that although our method has a complex framework with many parameters, our method does achieve well.

To further compare the generalization capability of the I-GAN with that of the other methods, we plot the receiver operating characteristic (ROC) curves. As shown in Fig. 11,

the area under curve (AUC) of all the methods improve and tend to 1 with the number of the labeled SAR images increasing. Our method achieves better AUC compared with the other methods. The difference is more significant to distinguish under 100 and 200 labeled samples. The high-quality generated images help to achieve a larger true positive rate (TPR), which enables our method to achieve better AUC.

VI. CONCLUSION

This paper has presented a novel 5-player semi-supervised GANs model for SAR image generation and recognition. In this framework, we combined the ability of the unconditional GANs for unsupervised feature extraction and the ability of the conditional GANs for supervised image-label matching. The separation of the discriminator and the classifier enables them to adopt different loss functions and learning rates, so they can maintain their own performance and assist each other simultaneously. Experiments on the MSTAR dataset have verified that the joint training among the five players can improve the feature extraction ability of the network efficiently under the insufficient labeled SAR images. Our model can generate high quality SAR images by category, which is a feasible tool to train the classifier. Therefore, our classifier has achieved the state-of-the-art recognition performance compared with the other semi-supervised learning methods.

ACKNOWLEDGMENT

The authors would like to thank the Associate Editor and the Anonymous Reviewers for their constructive suggestions.

REFERENCES

[1] H. Breit, T. Fritz, U. Balss, M. Lachaise, A. Niedermeier, and M. Vonavka, "TerraSAR-X SAR processing and products," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 2, pp. 727–740, Feb. 2010.

[2] F. Covello, F. Battazza, A. Coletta, E. Lopinto, C. Fiorentino, L. Pietranera, G. Valentini, and S. Zoffoli, "COSMO-SkyMed an existing opportunity for observing the earth," *J. Geodyn.*, vol. 49, nos. 3–4, pp. 171–180, Apr. 2010..

[3] R. Guida, A. Iodice, D. Riccio, and U. Stilla, "Model-based interpretation of high-resolution SAR images of buildings," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 1, no. 2, pp. 107–119, Jun. 2008.

[4] L. M. Novak, G. J. Owirka, and W. S. Brower, "Performance of 10- and 20-target MSE classifiers," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 36, no. 4, pp. 1279–1289, Oct. 2000.

[5] F. Gao, W. Lv, Y. Zhang, J. Sun, J. Wang, and E. Yang, "A novel semisupervised support vector machine classifier based on active learning and context information," *Multidimensional Syst. Signal Process.*, vol. 27, no. 4, pp. 969–988, Oct. 2016.

[6] Y. Sun, Z. Liu, S. Todorovic, and J. Li, "Adaptive boosting for SAR automatic target recognition," *IEEE Trans. Aerosp. Electron. Syst.,*, vol. 43, no. 1, pp. 112–125, Jan. 2007.

[7] S. Chen, H. Wang, F. Xu, and Y. Q. 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.

[8] C. Persello and L. Bruzzone, "Active and semisupervised learning for the classification of remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 11, pp. 6937–6956, Nov. 2014.

[9] T. Jebara, J. Wang, and S.-F. Chang, "Graph construction and *b*-matching for semi-supervised learning," in *Proc. Int. Conf. Mach. Learn.*, 2009, pp. 441–448.

[10] Z.-H. Zhou and M. Li, "Tri-training: Exploiting unlabeled data using three classifiers," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 11, pp. 1529–1541, Nov. 2005.

[11] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[12] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training GANs," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 2234–2242.

[13] F. Gao, F. Ma, J. Wang, J. Sun, E. Yang, and H. Zhou, "Semi-supervised generative adversarial nets with multiple generators for SAR image recognition," *Sensors*, vol. 18, no. 8, p. 2706, Aug. 2018.

[14] F. Gao, Y. Yang, J. Wang, J. Sun, E. Yang, and H. Zhou, "A deep convolutional generative adversarial networks (DCGANs)-based semi-supervised method for object recognition in synthetic aperture radar (SAR) images," *Remote Sens.*, vol. 10, no. 6, p. 846, Jun. 2018.

[15] C. Li, K. Xu, J. Zhu, and B. Zhang, "Triple generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 4088–4098.

[16] F. Gao, X. Xue, J. Sun, J. Wang, and Y. Zhang, "A SAR image despeckling method based on two-dimensional S transform shrinkage," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 5, pp. 3025–3034, May 2016.

[17] L. Theis, A. van den Oord, and M. Bethge, "A note on the evaluation of generative models," 2015, *arXiv:1511.01844*. [Online]. Available: https://arxiv.org/abs/1511.01844

[18] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proc. Conf. Comput. Learn. Theory*, 1998, pp. 92–100.

[19] F. Gao, F. Ma, J. Wang, J. Sun, E. Yang, and H. Zhou, "Visual saliency modeling for river detection in high-resolution SAR imagery," *IEEE Access*, vol. 6, pp. 1000–1014, Nov. 2017.

[20] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.

[21] Z. Yue, F. Gao, Q. Xiong, J. Wang, T. Huang, E. Yang, and H. Zhou, "A novel semi-supervised convolutional neural network method for synthetic aperture radar image recognition," *Cogn. Comput.*, pp. 1–12, Mar. 2019.

[22] F. Gao, T. Huang, J. Sun, J. Wang, A. Hussain, and E. Yang, "A new algorithm for SAR image target recognition based on an improved deep convolutional neural network," *Cogn. Comput.*, vol. 5, pp. 1–16, Jun. 2018.

[23] Q. Zhu, Y. Zhong, S. Wu, L. Zhang, and D. Li, "Scene classification based on the sparse homogeneous-heterogeneous topic feature model," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 5, pp. 2689–2703, 2018.

[24] B. Ding, G. Wen, X. Huang, C. Ma, and X. Yang, "Target recognition in synthetic aperture radar images via matching of attributed scattering centers," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 7, pp. 3334–3347, Jul. 2017.

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

[26] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4700–4708.

[27] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.

[28] F. Gao, T. Huang, J. Wang, J. Sun, A. Hussain, and E. Yang, "Dual-branch deep convolution neural network for polarimetric SAR image classification," *Appl. Sci.*, vol. 7, no. 5, p. 447, May 2017.

[29] Z. Huang, Z. Pan, and B. Lei, "Transfer learning with deep convolutional neural network for SAR target classification with limited labeled data," *Remote Sens.*, vol. 9, no. 9, p. 907, Sep. 2017.

[30] R. Shang, J. Wang, L. Jiao, R. Stolkin, B. Hou, and Y. Li, "SAR targets classification based on deep memory convolution neural networks and transfer parameters," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 8, pp. 2834–2846, Aug. 2018.

[31] K. Li, G. Cheng, S. Bu, and X. You, "Rotation-insensitive and context-augmented object detection in remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 4, pp. 2337–2348, Apr. 2018.

[32] P. Zhong, Z. Gong, S. Li, and C.-B. Schönlieb, "Learning to diversify deep belief networks for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 55, no. 6, pp. 3516–3530, Jun. 2017.

[33] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2013.

[34] L. Maaløe, C. K. Sønderby, S. K. Sønderby, and O. Winther, "Auxiliary deep generative models," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2017.

[35] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2016.
[36] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2017.
[37] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of Wasserstein GANs," in *Proc. Adv. Neural Inf. Process. Syst.* vol. 30, 2017, pp. 5767–5777.
[38] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, W. Zhen, and S. P. Smolley, "Least squares generative adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2794–2802.
[39] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014, *arXiv:1411.1784*. [Online]. Available: https://arxiv.org/abs/1411.1784
[40] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative adversarial text to image synthesis," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2016.
[41] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier GANs," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2017.
[42] J. T. Springenberg, "Unsupervised and semi-supervised learning with categorical generative adversarial networks," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2016.
[43] Y. Saatchi and A. G. Wilson, "Bayesian GAN," in *Proc. Adv. Neural Inf. Process. Syst.*, San Diego, CA, USA, vol. 30, 2017, pp. 3622–3631.
[44] E. R. Keydel, S. W. Lee, and J. T. Moore, "MSTAR extended operating conditions: A tutorial," *Proc. SPIE*, vol. 2757, pp. 228–242, Jun. 1996.
[45] K. Shmelkov, C. Schmid, and K. Alahari, "How good is my GAN?" in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 213–229.
[46] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.
[47] J. Cohen, "A coefficient of agreement for nominal scales," *Edu. Psychol. Meas.*, vol. 20, no. 1, pp. 37–46, 1960.
[48] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Proc. Adv. Neural Inf. Process. Syst.*, San Diego, CA, USA, vol. 30, 2017, pp. 1195–1204.
[49] T. Miyato, S.-I. Maeda, S. Ishii, and M. Koyama, "Virtual adversarial training: A regularization method for supervised and semi-supervised learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1979–1993, Aug. 2019.
[50] H. Chen, F. Zhang, B. Tang, Q. Yin, and X. Sun, "Slim and efficient neural network design for resource-constrained SAR target recognition," *Remote Sens.*, vol. 10, no. 10, p. 1618, 2018.