Using Semi-Supervised Generative Adversarial Networks to Target Recognition in SAR Images

Shijia Chai

***Abstract*—A semi-supervised generative adversarial networks (GANs) based synthetic aperture radar (SAR) target recognition method is proposed in this paper, aiming for solving the problem of automatic target recognition (ATR) in SAR images when the training set contains only few labeled data and relatively adequate unlabeled data. As one of the semi-supervised learning technique, semi-supervised GANs based approach is suitable for this problem and has achieved impressive performance on many tasks. The whole network consists of a generator and a discriminator. The generator learns the distribution of the target from the labeled and the unlabeled data, and generates the fake data. The discriminator predicts the class of the labeled data, and distinguishes the fake data from the real one. To avoid the model falling into the failure mode, we introduce a parameter to control the tradeoff between the supervised and the unsupervised loss, and decay the parameter during the training. To demonstrate the effectiveness of the proposed method, experiments on the Moving and Stationary Target Acquisition and Recognition (MSTAR) data set are conducted. Comparison with several state-of-the-art methods indicates that the proposed method obviously promotes the recognition accuracy.**

***Index Terms*—Automatic target recognition, synthetic aperture radar, semi-supervised learning, generative adversarial networks.**

I. INTRODUCTION

SYNTHETIC aperture radar (SAR) can work day-and-night, and is scarcely affected by meteorological conditions, making it suitable for reconnaissance and surveillance. Due to the various imaging mechanisms of optical and SAR sensors, the understanding of SAR images is quite different from the analysis of optical images. Automatic target recognition (ATR) in SAR images is a challenging task and has attracted a lot of interests in recent years. SAR-ATR can be divided into three stages: detection, discrimination, and classification [1], and we focus on the classification stage in this paper. Many approaches have been proposed to solve the classification problem in SAR-ATR, and the MSTAR data set is usually used to test the performance of these approaches. SAR-ATR approaches mainly put effort into two aspects: how to extract effective features and how to design strong classifiers. In one hand, many features are specially designed for SAR-ATR, such as salient point features [2], and scatter cluster features [3]. In the other hand, various classifiers are designed for target recognition in SAR images, including sparse representation based method [4], and manifolds based method [5]. As the development of the deep learning theory, deep neural networks are widely employed to solve the SAR-ATR problem, and achieve state-of-the-art performance under the MSTAR data set, profiting from the powerful feature extraction ability of neural networks. However, compared with large data sets, such as ImageNet, the data number in the MSTAR data set is still small, which may yield severe overfitting when training the networks. A lot of previous studies make effort on alleviating the overfitting issue by removing full connected layers [6], or introducing transfer learning and regularization terms [7].

Different from previous studies, we take account SAR target recognition with few labeled data and relatively adequate unlabeled data. This problem is worthy studying in part because quality annotation for SAR targets is costly and in part because few related studies are conducted although semi-supervised learning techniques are widely used in machine learning field. Semi-supervised learning is inherently suitable for solving this problem, and prevalent traditional semi-supervised learning approaches include transductive support vector machine (TSVM) [8] and graph based semi-supervised method [9]. As a generative model, GANs [10] can be used for semi-supervised learning and have achieved impressive performance [11]-[13]. In this paper, semi-supervised GANs are used to solve SAR target recognition problem by designing appropriate networks. Directly applying the existing approach, such as [13], will lead to model failure. To overcome this issue, we modify the loss of the model by introducing a tradeoff parameter that controls the weight between the supervised and the unsupervised loss. We will detail the method in section II. The first contribution of this work is that it is the first attempt to apply semi-supervised GANs to SAR-ATR, and the second contribution is that we design a new loss which is proven effective for the problem to be solved.

II. SEMI-SUPERVISED GANS BASED SAR-ATR

GANs [10] consist of two adversarial networks, a generator G and a discriminator D. G generates fake data and tries to fool D, and D tries to distinguish real data from fake data generated from G. This process can be formulated as the following min-max problem:

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log p(y=1|\boldsymbol{x},D)] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1-p(y=1|G(\boldsymbol{z}),D))] \tag{1}$$

where $\boldsymbol{x}$ and $\boldsymbol{z}$ represent the real and the noise data respectively, and y is the label of the data indicating if the

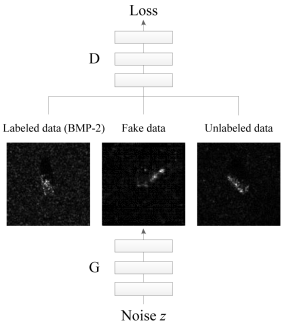

Fig. 1. The model architecture of the proposed method.

TABLE I
THE NETWORK STRUCTURE FOR THE MSTAR DATA SET

Generator G	Discriminator D
Input $\boldsymbol{z} \in \mathbb{R}^{100}$	Input 88 × 88 gray image
11 × 11 × 128 fc, ReLU	5 × 5 conv 16, ReLU
Batch normalization	2 × 2 max-pool, stride 2
4 × 4 fs-conv 64, stride 2, ReLU	5 × 5 conv 32, ReLU
Batch normalization	2 × 2 max-pool, stride 2
4 × 4 fs-conv 32, stride 2, ReLU	6 × 6 conv 64, ReLU
Batch normalization	2 × 2 max-pool, stride 2
4 × 4 fs-conv 1, stride 2, Sigmoid	5 × 5 conv 128, ReLU
Multiply 255	Dropout with keep probability 0.5
	3 × 3 conv 11, Softmax

data is real or fake. By simultaneously training G and D, G can capture the distribution of the real data. A widely used GAN model is the deep convolutional generative adversarial networks (DCGANs) [14], which consists of two convolutional neural networks.

GANs can be applied to semi-supervised learning, which focus on the issue of how to use the unlabeled data. Consider a K categories classification problem, let $\boldsymbol{x}$ and $\boldsymbol{z}$ denote the real and the noise data respectively, and y denote the label of the real data, which ranges from 1 to K. In order to utilize the unlabeled data, semi-supervised GANs adds a new class label $y = K+1$, which corresponds to the fake data [13]. The objective of D turns to: 1) for labeled data, predict its class; 2) for labeled or unlabeled data, recognize it as real data; and 3) for data generated from G, recognize it as fake data. This process can be formulated as the following min-max problem:

$$\begin{aligned}\min_G \max_D V(D,G) = &\mathbb{E}_{\boldsymbol{x},y\sim p_{\text{data}}(\boldsymbol{x},y)}[\log p(y=k|\boldsymbol{x},D)] \\ &+ \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log(1-p(y=K+1|\boldsymbol{x},D))] \\ &+ \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log p(y=K+1|G(\boldsymbol{z}),D)]\end{aligned} \tag{2}$$

The architecture of the semi-supervised GANs based SAR-ATR model is illustrated in Fig. 1. The whole model consists of a generator G and a discriminator D. Here we take the MSTAR data set as an example and the network structure is listed in Table I, where "5 × 5 conv 16" refers to a convolution layer with 5 × 5 kernel and 16 output feature maps, "2 × 2 max-pool" indicates a maximum pooling layer with 2 × 2 kernel, "fc" and "fs-conv" represent the full connected layer and the fractionally-strided convolution layer [14] respectively. The structure of G is designed similar with DCGANs [14] for its well performance of generating fake data.

For training the network, a mini-batch containing m labeled data, m unlabeled data, and m noise data is taken at a time in one iteration and the stochastic gradient descent is applied to minimize the loss function computed from the mini-batch. Denote $\mathbf{X}_l = [\boldsymbol{x}_l^1, \ldots, \boldsymbol{x}_l^m]$, $\mathbf{X}_u = [\boldsymbol{x}_u^1, \ldots, \boldsymbol{x}_u^m]$, and $\mathbf{Z} = [\boldsymbol{z}^1, \ldots, \boldsymbol{z}^m]$ as the labeled, unlabeled, and noise data in the mini-batch. Let $\mathbf{Y}_l = [\boldsymbol{y}_l^1, \ldots, \boldsymbol{y}_l^m]$ represents the ground truth label of $\mathbf{X}_l$, here the label is the one-hot version of the class label. The network contains two losses: the generator loss G_L and the discriminator loss D_L. G_L is obtained through:

$$G_L = -\frac{1}{m}\sum_{i=1}^{m}\log(1 - R_l(D(G(\boldsymbol{z}^i))) + \epsilon) \tag{3}$$

where $R_l(\cdot)$ is the operation to extract the last element of the input, and ϵ is a small positive number assuring the numerical stability of the method. D_L consists of two parts, the supervised part D_S and the unsupervised part D_U, and can be written as follows:

$$D_L = wD_S + D_U \tag{4}$$

where w controls the tradeoff between the two losses. D_S is computed as:

$$D_S = \frac{1}{m}\sum_{i=1}^{m} H(\boldsymbol{y}_l^i, R_K(D(\boldsymbol{x}_l^i))) \tag{5}$$

where $H(\boldsymbol{p}, \boldsymbol{q})$ computes the cross entropy between $\boldsymbol{p}$ and $\boldsymbol{q}$, $R_K(\cdot)$ extracts the first K elements of the input to form a K-dimensional vector. D_U is calculated as:

$$\begin{aligned}D_U = D_U^U + D_U^F = &-\frac{1}{m}\sum_{i=1}^{m}\log(1 - R_l(D(\boldsymbol{x}_u^i)) + \epsilon) \\ &-\frac{1}{m}\sum_{i=1}^{m}\log(R_l(D(G(\boldsymbol{z}^i))) + \epsilon)\end{aligned} \tag{6}$$

where D_U^U and D_U^F represent the losses generated from unlabeled and fake data respectively.

Instead of setting the tradeoff parameter w as a constant value, which will lead to model failure for semi-supervised GANs based SAR-ATR, we gradually decay it. More specifically speaking, w is set as a large value initially, and reduced by a factor of α after each T iterations, which can be formulated as:

$$D_L(t) = w(t)D_S(t) + D_U(t) \tag{7}$$

where t indicates the t th iteration, and $w(t)$ is computed as:

$$w(t) = w(0)\alpha^{\lfloor t/T \rfloor} \tag{8}$$

where $\lfloor x \rfloor$ denotes the floor operation. The idea of decaying the tradeoff parameter is straight forward. At the beginning, the supervised loss should be paid more attention. As the training continues, the discriminator can classify the labeled data well, and at that time unlabeled data should play more important role as a consequence. It is time to stop the training when $w(t)$ is very small (close to zero).

III. EXPERIMENT AND DISCUSSION

A. Data Description and Parameter Setting

The effect of the proposed method is evaluated using the MSTAR data set under standard operating conditions (SOC) and extended operating conditions (EOC). In the SOC test

TABLE II
THE TRAINING AND THE TEST SET UNDER SOC

		Training images			Testing images	
Class	Serial No.	Depr.	$\#\mathbb{X}_l$	$\#\mathbb{X}_u$	Depr.	$\#\mathbb{T}$
2S1	b01	17°	10	289	15°	274
BMP2	9563	17°	10	223	15°	195
BRDM2	E-71	17°	10	288	15°	274
BTR70	c71	17°	10	223	15°	196
BTR60	k10yt7532	17°	10	246	15°	195
D7	92v13015	17°	10	289	15°	274
T62	A51	17°	10	289	15°	273
T72	132	17°	10	222	15°	196
ZIL131	E12	17°	10	289	15°	274
ZSU23/4	d08	17°	10	289	15°	274

TABLE III
THE TEST SET UNDER EOC

Class	Serial No.	Depr.	$\#\mathbb{T}$
2S1	b01	30°	288
BRDM2	E71	30°	287
T72	A64	30°	288
ZSU23/4	d08	30°	288

scenario, the serial numbers in the test set are the same as those in the training set, and depression angles of the data in the training and the test set are similar. In the SOC experimental setup, the proposed method is tested on the ten-class targets classification problem. Details about the training and the test set are listed in Table II, where Depr. is short for depression angle, $\#\mathbb{X}_l$, $\#\mathbb{X}_u$, and $\#\mathbb{T}$ represent the sample number of the labeled data set $\mathbb{X}_l$, the unlabeled data set $\mathbb{X}_u$, and the test data set $\mathbb{T}$ respectively. In the EOC test scenario, samples in the training and the test set have large differences. We evaluate the proposed method on the four-class targets classification problem with large depression angle changes. Details about the test set under EOC are listed in Table III, and the training set consists of the corresponding four targets with 17° depression angles, the details of whom are presented in Table II. Under both SOC and EOC scenarios, 10 images of each class in the training set are selected as the labeled data, and the rest are taken as the unlabeled data. We select the labeled data so that they can cover all aspect angles as well as possible, since SAR images are sensitive to the variance of aspect angles. Ten 88 $\times$ 88 patches are randomly sampled from the original 128 $\times$ 128 image chips for data augmentation, therefore each class includes 100 data for the training purpose after the augmentation. In the following experiment, all compared state-of-the-art methods, as well as the approach without decaying the tradeoff parameter w, use the same training data, including the augmented one.

Parameters of the proposed method is set as follows: the mini-batch size m is set as 100, the initial tradeoff parameter $w(0)$ is set as 5, the decay factor α and the decay step T is set as 0.8 and 1000 respectively. The learning rates of D and G are initialized as 0.0001 and 0.0002, and we decay the both by a factor of 0.99 after every 5000 iterations. D is trained once and G is trained twice in each iteration. We stop the training when $w(t)$ is less than 0.03, that requires 23000 iterations.

B. Performance of the Proposed Method

This subsection is devoted to validating the effect of the proposed method. Under the SOC test scenario, an average accuracy of 85.65% is achieved when using 10 labeled data in each class and the confusion matrix of the proposed method is shown in Table IV. In this Table, each row gives the ground-truth class of the target and each column denotes the target class predicted by the proposed method. Fig. 2 illustrates the curves of the supervised discriminator loss D_S, the unsupervised discriminator loss generated from unlabeled data D_U^U, the unsupervised discriminator loss generated from fake data D_U^F, and the generator loss G_L respectively. It can be seen from Fig. 2 that D_S decreases gradually as the increasing of the iteration number, which indicates that D improves the ability of classifying the labeled data. The value of D_U^U is small, and after a period of oscillation, the loss decreases in general. As shown in Fig. 2(c) and Fig. 2(d), for the fake data, the discriminator loss decreases while the generator loss increases, which indicates that as the training continues the discriminator D gets better than the generator G. Fig. 3 depicts the fake data generated from G after 0, 10000, and 20000 iterations respectively. It is clear that at the beginning the generated data is merely noise, and over the course of training the generated data is more similar with the real data. The result in Fig. 3 demonstrates that the generator G gets better during the training, and by combining with the conclusion obtained from Fig. 2(c) and Fig. 2(d), it can be drawn that the discriminator becomes even better, which is what we expect.

The necessity of decaying the tradeoff parameter w is explained in Fig. 4. Fig. 4(a) plots the curve of the generator loss G_L when w is set as 1. It can be seen that G_L increases steeply at the 14000th iteration, and the generated samples at that time are illustrated in Fig. 4(b), which indicates that G fails. Fig. 4(c) and Fig. 4(d) show the G_L curve and the generated samples when the model fallen into the failure mode at the 15000th iteration with $w = 5$. Fig. 4(e) and Fig. 4(f) give the corresponding results when w is set as 10, and the failure occurs at the 17000th iteration. The recognition accuracies drop a lot when the model falls into the failure mode and are 76.29%, 77.36%, and 77.81% under the above three circumstances. By comparing the result in Fig. 2 and Fig. 3 with that in Fig. 4, it can be drawn that setting w as a constant will lead to the model failure, and the failure can be avoided by decaying w. We attribute the model failure to the overfitting of the discriminator, that is validated by the result shown in Fig. 5, that plots the curves of the discriminator loss of unlabeled data and fake data with $w = 5$. It can be seen from Fig. 5 that when the model falls into the failure mode at the 15000th iteration when the generator loss G_L increases steeply, the discriminator loss of unlabeled data D_U^U and the discriminator loss of fake data D_U^F dramatically drop to almost zero, which indicates the overfitting of the discriminator. At that moment, the discriminator completely wins the minimax two-play game, and the generator cannot generates well samples since then. The consistent results are obtained when w is set as 1 and 10 respectively.

Under the EOC test scenario, the proposed method obtains

TABLE IV
THE CONFUSION MATRIX OF THE PROPOSED METHOD UNDER SOC

Class	2S1	BMP2	BRDM2	BTR70	BTR60	D7	T62	T72	ZIL131	ZSU23/4	$P_{cc}(\%)$
2S1	221	1	27	3	0	1	11	8	2	0	80.66
BMP2	17	147	0	5	6	1	6	13	0	0	75.38
BRDM2	8	1	249	9	0	0	0	0	4	3	90.88
BTR70	7	9	8	172	0	0	0	0	0	0	87.76
BTR60	1	7	0	24	137	3	10	5	4	4	70.26
D7	1	0	0	0	0	244	8	0	4	17	89.05
T62	12	0	0	0	0	6	238	13	0	4	87.18
T72	0	4	0	0	0	4	16	171	0	1	87.24
ZIL131	11	0	1	0	1	9	1	1	242	8	88.32
ZSU23/4	1	0	0	0	0	11	3	0	3	256	93.43
Total											85.65

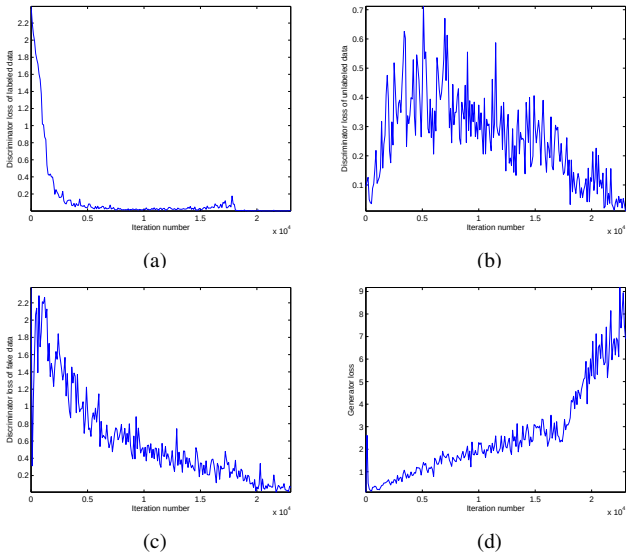

Fig. 2. (a) The curve of D_S. (b) The curve of D_U^U. (c) The curve of D_U^F. (d) The curve of G_L.

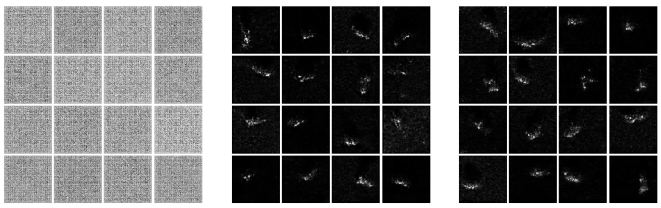

Fig. 3. (a) Generated data after 0 iterations. (b) Generated data after 10000 iterations. (c) Generated data after 20000 iterations.

an average accuracy of 75.50% when using 10 labeled data in each class, and the corresponding confusion matrix is presented in Table V. The accuracy of the proposed method under EOC is not as well as that under SOC, since SAR images are very sensitive to the various of depression angles.

Data augmentation is helpful to alleviate the overfitting issue and learn well model. The training data is augmented by 10 times in the above experiment. Average accuracies of the proposed method when the augmented ratio is set as 10, 20, and 30 respectively are presented in Table VI. The result

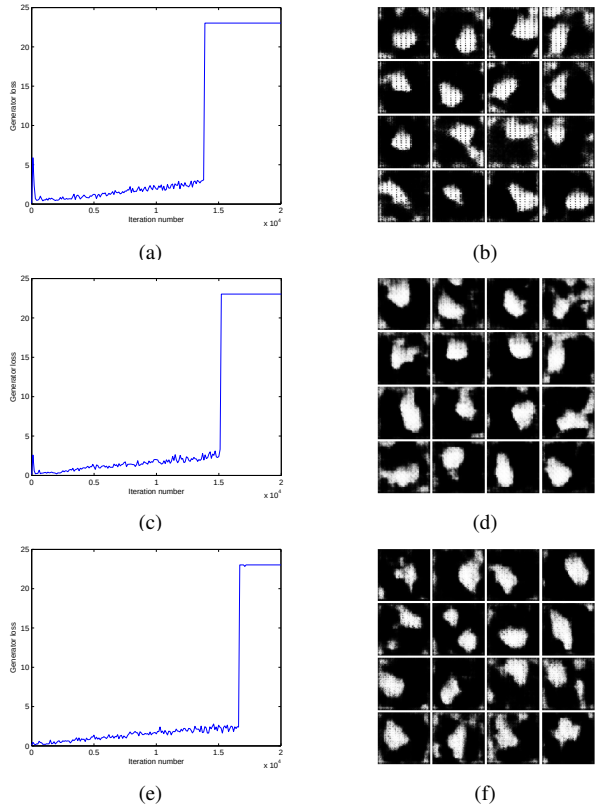

Fig. 4. (a) The G_L curve when $w = 1$. (b) The generated data when the model falls into the failure mode with $w = 1$. (c) The G_L curve when $w = 5$. (d) The generated data when the model falls into the failure mode with $w = 5$. (e) The G_L curve when $w = 10$. (f) The generated data when the model falls into the failure mode with $w = 10$.

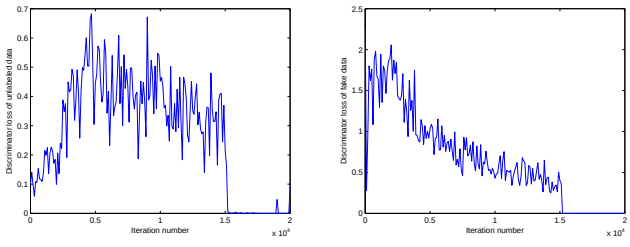

Fig. 5. (a) The D_U^U curve when $w = 5$. (b) The D_U^F curve when $w = 5$.

TABLE V
THE CONFUSION MATRIX OF THE PROPOSED METHOD UNDER EOC

Class	2S1	BRDM2	T72	ZSU23/4	$P_{cc}(\%)$
2S1	168	95	24	1	58.33
BRDM2	17	261	4	5	90.94
T72	80	3	165	40	57.29
ZSU23/4	8	0	5	275	95.49
Total					75.50

TABLE VI
AVERAGE ACCURACIES OF THE PROPOSED METHOD UNDER VARIOUS AUGMENTED RATIOS

	SOC-10	SOC-20	SOC-30
10 times	85.65%	89.86%	93.07%
20 times	85.73%	89.81%	93.11%
30 times	85.86%	89.94%	92.99%

TABLE VII
AVERAGE ACCURACIES OF VARIOUS METHODS UNDER SOC

	SOC-10	SOC-20	SOC-30
Semi-Graph	56.45%	69.86%	72.49%
A-ConvNets	70.76%	81.36%	87.34%
CNN-TL	74.10%	83.09%	89.07%
SiameseNets	72.04%	86.43%	90.60%
CatGAN	77.90%	84.66%	89.44%
SSGANs	85.65%	89.86%	93.07%

TABLE VIII
AVERAGE ACCURACIES OF VARIOUS METHODS UNDER EOC

	Semi-Graph	A-ConvNets	CNN-TL
Acc	62.90%	70.89%	71.76%
	SiameseNets	CatGAN	SSGANs
Acc	70.46%	71.24%	75.50%

indicates that 10 times magnification is enough for improving the performance of the method, and a larger value of the augmented ratio will not boost the accuracy further. It is worth pointing out that solely using data augmentation fails to prevent the model from falling into the failure mode, while the proposed method can solve this issue.

C. Comparison with the State-of-the-art Methods

We compare the proposed semi-supervised GANs (SSGANs) method with five related methods: Semi-Graph [9], a traditional graph based semi-supervised learning method; A-ConvNets [6], which achieves the state-of-the-art performance under the MSTAR dataset; CNN-TL [7], that can utilize unlabeled samples via transfer learning and reconstructed bypass regularization, and gives better result than A-ConvNets when using small amount of training data; SiameseNets [15], a widely used few-shot learning method, and CatGAN [11], a prevalent semi-supervised GANs based method. Although A-ConvNets and SiameseNets are not designed for semi-supervised learning, we take them as the benchmark for revealing how many samples the proposed method needs to obtain the similar recognition rate with the method that does not use semi-supervised techniques. Average accuracies of various methods are compared in Table VII, where the symbol n in SOC-n indicates the number of the labeled training data in each category. Comparison results under EOC with 10 labeled training data in each category are presented in Table VIII. It can be seen from the result that the proposed method consistently achieves the highest recognition rate under both SOC and EOC test scenarios, which demonstrates the effectiveness of the method. We observe that the improvement of the proposed method than the compared methods in the EOC test scenario is not as large as that under the SOC experimental setup, especially for 2S1 and T72. One probable explanation of this phenomenon is that the more striking distribution discrepancy between the training and the test sets makes learning exact data distribution from unlabeled samples become harder.

IV. CONCLUSION

GANs are a recent popular technique for semi-supervised learning. We apply GANs to SAR-ATR when the training set contains only few labeled data and relatively adequate unlabeled data. Unlabeled samples are also used to learn the distribution of the target, that can improve the performance of the discriminator. A decayed tradeoff parameter is introduced for preventing the model from falling into the failure mode, and experimental results validate its effectiveness.

REFERENCES

[1] D. E. Dudgeon and R. T. Lacoss, "An overview of automatic target recognition," *Lincoln Lab. J.*, vol. 6, no. 1, pp. 3-10, 1993.

[2] J. Chen, B. Zhang, and C. Wang, "Backscattering feature analysis and recognition of civilian aircraft in TerraSAR-X images," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 4, pp. 796-800, Apr. 2015.

[3] Z. Pan, X. Qiu, Z. Huang, and B. Lei, "Airplane recognition in TerraSAR-X images via scatter cluster extraction and reweighted sparse representation," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 1, pp. 112-116, Jan. 2017.

[4] G. Dong, G. Kuang, N. Wang, L. Zhao, and J. Lu, "SAR target recognition via joint sparse representation of monogenic signal," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 7, pp. 3316-3328, Jul. 2015.

[5] X. Liu, Y. Huang, J. Pei, and J. Yang, "Sample discriminant analysis for SAR ATR," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 12, pp. 2120-2124, Dec. 2014.

[6] 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.

[7] 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, pp. 1-21, Sep. 2017.

[8] T. Joachims, "Transductive inference for text classification using support vector machines," in *Proc. Internat. Conf. Mach. Learn.*, Jun. 1999, pp. 200-209.

[9] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Scholkopf, "Learning with local and global consistency," in *Proc. Adv. Neural Inf. Proces. Syst.*, Dec. 2004, pp. 284-291.

[10] 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. Adv. Neural Inf. Proces. Syst.*, Dec. 2014, pp. 2672-2680.

[11] J. T. Springenberg, "Unsupervised and semi-supervised learning with categorical generative adversarial networks," in *Proc. Internat. Conf. Learn. Representations*, May 2016, pp. 1-20.

[12] K. Sricharan, R. Bala, M. Shreve, H. Ding, K. Saketh, and J. Sun, "Semi-supervised conditional GANs," *arXiv preprint arXiv: 1708.05789*, 2017.

[13] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training GANs," in *Proc. Adv. Neural Inf. Proces. Syst.*, Dec. 2016, pp. 2234-2242.

[14] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," in *arXiv preprint arXiv: 1511.06434*, 2016.

[15] S. Zagoruyko and N. Komodakis, "Learning to compare image patches via convolutional neural networks," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2015, pp. 4353-4361.