Dropout-based Adversarial Training Networks for Remote Sensing Scene Classification

Xin Wang, Zhipeng Mao, Aiye Shi, Zhilu Zhang, and Huiyu Zhou

***Abstract*—Scene classification in remote sensing (RS) images is a challenging task due to the lack of well labeled data. Recently, deep transfer learning (DTL) has been proposed to handle this task. However, most DTL methods cannot effectively deal with ambiguous features on class boundaries and multi-modal structures of RS data, so their performance is unsatisfactory. To handle the challenges, this letter presents a novel dropout-based adversarial training network for RS scene classification. Specifically, a dropout-based label classifier module is designed to reduce the selection of ambiguous features. Then, a dropout-based domain discriminator module is constructed to capture multi-modal structures of RS images so as to achieve fine-grained alignment between cross-domain distributions. Third, a joint distribution of features and labels is built to further enhance the performance. Experiments on seven public RS data sets show that our model outperforms several state-of-the-arts under different conditions. The code of our method is publicly available at: https://github.com/WangXin81/DATN-Submitted-to-IEEE-GRSL**

***Index Terms*—Remote sensing, scene classification, deep transfer learning, adversarial training, dropout.**

I. Introduction

With the advances of remote sensing (RS) technologies, massive high-resolution RS (HRRS) data has been available, providing abundant valuable information for RS image interpretation. As one key task, scene classification, aiming at classifying different RS scenes into various semantic classes, has played a significant role in the RS community [1].

In recent years, various methods have been proposed for RS scene classification, among which deep learning (DL) methods have attracted increasing interests due to their capabilities of semantic-level feature extraction [2]. To ensure favorable classification performance, DL methods requires complex network architectures and huge labeled samples. However, collecting and annotating massive RS scene data is not only time-consuming, but also extremely expensive. Recently, leveraging knowledge from a labeled source domain to support an unlabeled target domain has become feasible. Under this umbrella, many deep transfer learning (DTL) methods that fuse DL and transfer learning (TL) together have emerged and brought promising outcomes in image classification [3].

X. Wang is supported in part by Six Talents Peak Project of Jiangsu Province under Grant XYDXX-007 and Jiangsu Province Government Scholarship for Studying Abroad. H. Zhou is supported Royal Society-Newton Advanced Fellowship under Grant NA160342. (*Corresponding author: Xin Wang.*)

X. Wang, Z. Mao, A. Shi, and Z. Zhang are with the School of Computer and Information, Hohai University, Nanjing 211100, China (e-mail: wang_xin@hhu.edu.cn).

H. Zhou is a full Professor at School of Computing and Mathematical Sciences, University of Leicester, Leicester LE1 7RH, United Kingdom (e-mail: hz143@leicester.ac.uk).

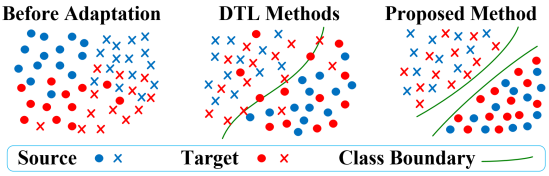

Fig. 1. Feature distributions adapted by different methods. Left: before adaptation. Middle: adaptation by DTL methods which do not consider ambiguous features on class boundaries. Right: adaptation by our method.

Although remarkable progress has been achieved in current researches, there still exist several challenges for HRRS scene classification. First, ground objects in HRRS images have close relationship to adjacent objects and surrounding environments. Various objects form diverse semantic scenes through different spatial distributions. Hence, HRRS scenes usually have large intra-class variations and low inter-class dissimilarities. However, features extracted in most DTL methods seem to be domain-dependent, with weak discriminative abilities. As shown in Fig. 1, feature points from DTL methods may be very close to or sit on the class boundary, resulting in wrong classification results. Second, due to different imaging conditions, such as seasons, weathers, clouds, etc., the radiation intensity and color of scenes belonging to the same category may look different, as shown in Fig. 2. In other words, these scenes have complex multi-modal structures. However, most existing adversarial learning models generally adopt a single domain discriminator, and thus cannot identify such multi-modal information. In this case, data distributions on the source and target domains are confusing. Third, many domain adaptation methods attempt to align feature distributions of the source and target domains without taking label information of samples into consideration [4]. Nevertheless, when the target domain lacks label information, the label information in the source domain becomes extremely critical.

To overcome the above challenges, in this letter, we propose a novel dropout-based adversarial training network (DATN). First, to solve the challenge of ambiguous features, our idea is to generate more discriminative features. Considering that ambiguous features may increase the uncertainty of classification, and ensemble learning [5], as a famous theory in machine learning, has indicated that integrating multiple learners instead of a single learner can achieve better generalization performance, we address to fuse two label classifiers together to reduce the classification uncertainty. However, directly fusing two separate classifiers that do not share weights with each other may greatly increase the network cost. Hence, we embed the idea of dropout into the two label classifiers, so as to reduce the ambiguous features and at the same time maintain the network cost. Second, for the challenge of multi-modal structures, inspired by generative adversarial

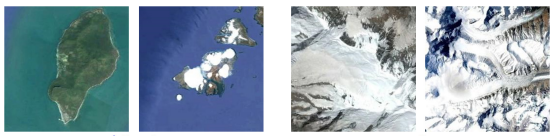

Fig. 2. Multi-modal structures of scenes due to different imaging conditions.

networks [6] that use multiple discriminators for enhancing distribution matching, we design multiple domain discriminators to capture different modes. By combining multi-adversarial domain discriminators together, the cross domain data distributions under multi-modal structures can be matched to the greatest extent, and the misalignment under different modes can be well reduced. Nevertheless, with the increase of the number of discriminators, the model parameters and complexity will increase, and the network becomes difficult to train. Therefore, the idea of dropout is also introduced into the multiple domain discriminators to help mitigating the multi-modal structure problem without increasing the network cost. Third, to better minimize the discrepancy of source and target domains, it is necessary to utilize the image features and labels simultaneously for domain adaptation. With this aim, we employ the Kronecker product [7] to exploit the joint distributions of features and labels for the source and target domains alignment.

The main contributions can be summarized as follows.

1) A dropout-based label classifier (DLC) is proposed in DATN, and by combining such two classifiers together, the ambiguous features on class boundaries can be reduced without increasing any network cost.
2) A dropout-based domain discriminator (DDD) is designed based on the idea of multi-adversarial domain adaptation to capture multi-modal structures of RS data. Meanwhile, it incorporates dropout into domain discriminators, efficiently aligning the cross-domain data distributions whilst ensuring the applicability of our model.
3) Instead of using a single feature distribution, DATN adopts the Kronecker product to produce the joint distribution of features and labels. Minimizing the joint distribution of features and labels can help to learn domain invariant features with stronger discrimination.

II. Proposed Method

Fig. 3 shows the overall architecture of our proposed DATN, which mainly consists of three modules: FG, DLC and DDD.

A. Feature Generator (FG)

Suppose $\mathbf{D}_s = \{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^{N_s}$ is the source domain containing N_s RS images $\mathbf{x}_i^s$ with class labels y_i^s, and $\mathbf{D}_t = \{\mathbf{x}_i^t\}_{i=1}^{N_t}$ is the target domain with N_t unlabeled RS samples $\mathbf{x}_i^t$. The goal of DATN is to predict the labels y^t of images from $\mathbf{D}_t$.

In DATN, the first step is to extract semantically meaningful features from RS data, where ResNet-50 [8] is experimentally chosen as the feature generator due to its relatively low number number of parameters and a good capability of tackling gradient vanishing problems. As shown in Fig. 3, there are total five stages in ResNet-50, and the highest-level features obtained from the last stage ('Conv 5-3') are used as the deep features for RS data. Mathematically, for a sample $\mathbf{x}_i^s$ from the source domain, this process can be viewed as:

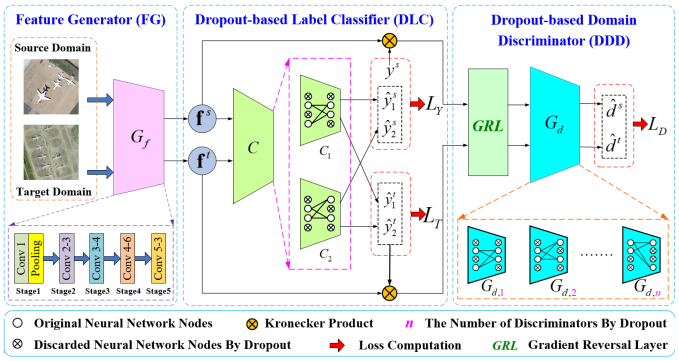

Fig. 3. Architecture of the proposed DATN.

$$\mathbf{f}_i^s = G_f(\mathbf{x}_i^s;\boldsymbol{\theta}_f) \in \mathbb{R}^{1\times 1\times C} \quad (1)$$

where G_f is the feature generator parameterized by $\boldsymbol{\theta}_f$; $\mathbf{f}_i^s$ is the extracted feature vector with C dimensions. Similarly, by feeding $\mathbf{x}_i^t$ into G_f, we can get $\mathbf{f}_i^t = G_f(\mathbf{x}_i^t;\boldsymbol{\theta}_f) \in \mathbb{R}^{1\times 1\times C}$.

B. Dropout-based Label Classifier (DLC)

With FG, we have collected high-level features. Next, DLC is constructed to predict labels of samples from both domains using the learned features. In Fig. 3, the block 'C' in DLC actually corresponds to two drop-based label classifiers, named C_1 and C_2. In each classifier, dropout is adopted to randomly select neurons from an exponential number of networks during training to enhance performance whilst preventing over-fitting.

First, to suppress ambiguous features and simultaneously enhance the discriminative abilities of features, we feed $\mathbf{f}_i^s$ and $\mathbf{f}_i^t$ learned from the source and target domains respectively into C_1 and C_2 parameterized by $\boldsymbol{\theta}_y$. The outputs of the two classifiers are written as:

$$\hat{y}_{i,j}^s = C_j(\mathbf{f}_i^s;\boldsymbol{\theta}_y),\ j=1,2 \quad (2)$$

$$\hat{y}_{i,j}^t = C_j(\mathbf{f}_i^t;\boldsymbol{\theta}_y),\ j=1,2 \quad (3)$$

where $\hat{y}_{i,j}^s$ and $\hat{y}_{i,j}^t$ denote the labels predicted by the j-th label classifier on the source and target domains, respectively.

Second, we compute the source classification losses of the two classifiers as follows:

$$L_{Y,j} = -1/N_s \sum_{i=1}^{N_s} y_i^s \log \hat{y}_{i,j}^s,\ j=1,2 \quad (4)$$

Then, the joint source classification loss can be obtained by:

$$L_Y = L_{Y,1} + L_{Y,2} = -1/N_s \sum_{i=1}^{N_s} y_i^s \log \hat{y}_{i,1}^s - 1/N_s \sum_{i=1}^{N_s} y_i^s \log \hat{y}_{i,2}^s \quad (5)$$

Ideally, the prediction results of the two classifiers should be the same. However, ambiguous features may lead to the inconsistency of the prediction results. To overcome it, we build a novel objective function to minimize the differences between the predicted label distributions of the two classifiers on the target domain, which we call the inconsistency loss:

$$L_t = 1/N_t \sum_{i=1}^{N_t} \left\| \hat{y}_{i,1}^t - \hat{y}_{i,2}^t \right\|_1 \quad (6)$$

where $\|\cdot\|_1$ represents the $\ell 1$ distance.

C. Dropout-based Domain Discriminator (DDD)

To capture multi-modal structures of RS scenes, we design a dropout-based domain discriminator module.

First, before feeding the features extracted by FG into DDD, we integrate the extracted features and the corresponding label information together via the Kronecker product, so as to obtain

TABLE I
DATA SETS USED FOR OUR EXPERIMENTS.

Data source	Data set	Number of classes	Image number	Image size	Spatial resolution(m)	Year
◇	NWPU	45	31500	256×256	0.2~30	2016
◇	AID	30	10000	600×600	0.5~8	2017
※	UCM	21	2100	256×256	0.3	2010
□	PatternNet	38	30400	256×256	0.062~4.693	2018
*	VA	38	59071	256×256	0.07~19.11	2020
☆	VB	38	58944	256×256	0.07~38.22	2020
□	VG	38	59404	256×256	0.075~9.555	2019

◇: Google Earth ※: USGS □: Google Map *: ArcGIS World Imagery ☆: Bing World Imagery

the joint distribution of features and labels. Let $\otimes$ denote the Kronecker product, $\mathbf{P}$ be a $u\times v$ matrix, and $\mathbf{Q}$ be a $l\times k$ matrix. The Kronecker product $\mathbf{P}\otimes\mathbf{Q}$ can be defined as:

$$\mathbf{P}\otimes\mathbf{Q}=\begin{pmatrix} p_{11}\mathbf{Q} & \cdots & p_{1v}\mathbf{Q} \\ \vdots & \ddots & \vdots \\ p_{u1}\mathbf{Q} & \cdots & p_{uv}\mathbf{Q} \end{pmatrix} \tag{7}$$

where p_{uv} is the element in the u-th row and v-th column of $\mathbf{P}$.

For the source domain, since its labels are known, the joint distribution of its feature $\mathbf{f}_i^s$ and label y_i^s can be calculated as $\mathbf{z}_i^s=\mathbf{f}_i^s\otimes y_i^s$. For the target domain, since its true labels are unknown, we use the predicted labels $\hat{y}_{i,1}^t$ and $\hat{y}_{i,2}^t$ to produce the pseudo label information. We express this process as:

$$\hat{y}_i^t=(\hat{y}_{i,1}^t+\hat{y}_{i,2}^t)/2 \tag{8}$$

Then, the joint distribution of target domain feature $\mathbf{f}_i^t$ and pseudo label $\hat{y}_i^t$ can be obtained by $\mathbf{z}_i^t=\mathbf{f}_i^t\otimes\hat{y}_i^t$.

Second, we design a gradient reversal layer (GRL) [9] before the dropout-based domain discriminators are deployed. With GRL, the network parameters can be updated through the standard back propagation.

Third, we construct n dropout-based domain discriminators, and feed the joint distributions $\mathbf{z}_i^s$ and $\mathbf{z}_i^t$ from both of the source and target domains into these domain discriminators separately. The outputs of the q-th $(q=1,2,...,n)$ domain discriminator $G_{d,q}$ parameterized by $\boldsymbol{\theta}_d$ can be expressed as:

$$\hat{d}_{i,q}^s=G_{d,q}(\zeta(\mathbf{z}_i^s);\boldsymbol{\theta}_d)=G_{d,q}(\zeta(\mathbf{f}_i^s\otimes y_i^s);\boldsymbol{\theta}_d) \tag{9}$$

$$\hat{d}_{i,q}^t=G_{d,q}(\zeta(\mathbf{z}_i^t);\boldsymbol{\theta}_d)=G_{d,q}(\zeta(\mathbf{f}_i^t\otimes y_i^t);\boldsymbol{\theta}_d) \tag{10}$$

where $\hat{d}_{i,q}^s$ and $\hat{d}_{i,q}^t$ are the predicted labels by $G_{d,q}$ on source and target domains, respectively. ζ denotes the GRL function.

Fourth, we define the true domain label $d_i^s=1$ for the source domain, while the true domain label $d_i^t=0$ for the target domain. Then, the loss function of these domain discriminators based on the cross-entropy loss is defined as:

$$L_D=-\frac{1}{N_s}\sum_{i=1}^{N_s}\sum_{q=1}^{n}d_i^s\log\hat{d}_{i,q}^s-\frac{1}{N_t}\sum_{i=1}^{N_t}\sum_{q=1}^{n}(1-d_i^t)\log(1-\hat{d}_{i,q}^t) \tag{11}$$

At last, based on Eqs. (5), (6), and (11), the overall objective loss function of our proposed DATN is given as:

$$L=L_Y+\alpha L_t+\beta L_D \tag{12}$$

where α and β are the trade-off parameters. The ultimate goal of network training is to find the optimal values of $\boldsymbol{\theta}_f^*$, $\boldsymbol{\theta}_y^*$, and $\boldsymbol{\theta}_d^*$ for FG, DLC and DDD, by minimizing Eq. (12).

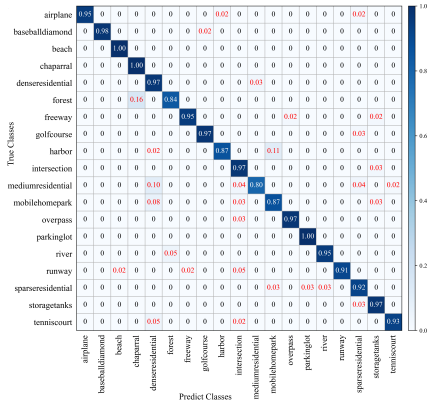

Fig. 4. CM of our method for the first transfer task-N_A → UCM.

To summarize, the whole network training process is equivalent to a two player game, in which one player is DDD used to distinguish whether the sample image comes from the source domain or the target domain, and the other player is FG used to confuse DDD to make DDD hardly distinguish between the source domain and the target domain so as to achieve the purpose of extracting domain invariant features. Ultimately, through continuous optimization for DDD and FG, the whole network becomes 'adversarial'.

III. EXPERIMENTS AND ANALYSIS

A. Data Sets

To evaluate the proposed DATN, we design eight transfer experiments on seven famous RS scene data sets: NWPU-RESISC45 (NWPU) [10], Aerial Image Data Set (AID) [11], UCMerced_LandUse (UCM) [12], PatternNet [13], VArcGIS (VA) [14], VBing (VB) [14], and VGoogle (VG) [15]. The detailed information of these data sets are given in Table I.

First, we combine NWPU and AID to construct a merged data set (called N_A) with 55 categories. There are 19 identical classes between N_A and UCM. We select these categories in N_A as the source domain, while use the 19 identical classes in UCM as the target domain. Thus, we get the first transfer task: N_A → UCM. For convenience, here we directly combine NWPU and AID. In future work, we will construct a merged source domain by other more effective strategies [16], [17].

Second, there are 22 same categories between PatternNet and N_A. We choose these classes in N_A as the source domain, while using the 22 identical classes in PatternNet as the target domain. Thus, we get the second task: N_A → PatternNet.

Third, as the newly proposed data sets, VA, VB and VG have 38 identical classes. We use either of them as the source or target domains, and then carry out six transfer tasks: VA → VB, VA → VG, VB → VA, VB → VG, VG → VA, and VG → VB.

B. Implement Details

All experiments are conducted on NVIDIA GeForce RTX 2080Ti GPU with Pytorch. The development environment is Pycharm on Ubuntu 18.04.1 system. Optimization is performed using Adam with the weight decay penalty of 10^{-4} and a batch size of 64. The learning rate is 3×10^{-4} and the total training epoch number is 600. All image samples are resized to 256×256 pixels. The trade-off parameters α and β are empirically set to 0.8 and 1. The evaluation metrics are overall accuracy (OA), confusion matrix (CM), and $\mathcal{A}$-distance.

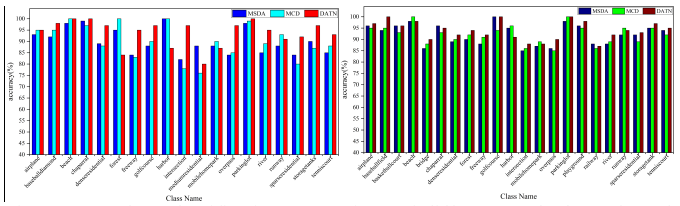

Fig. 5. Per-class classification comparison of different methods. Left: task N_A → UCM. Right: task N_A → PatternNet.

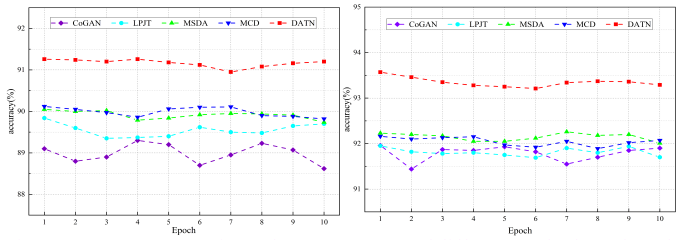

Fig. 6. Accuracy comparison after convergence of different methods. Left: task N_A → UCM. Right: task N_A → PatternNet.

C. Comparative Study

To measure the classification performance of the proposed method, we take the first task, i.e., N_A → UCM, for example, and give the classification accuracy for each class in Fig. 4. It can be seen that our method yields over 90% classification accuracy for most scene categories.

We also compare the proposed method with a number of state-of-the-arts (SOTAs). All SOTAs in our experiments use ResNet-50 pre-trained on ImageNet as their backbone. The results are given in Figs. 5 and 6. Fig. 5 shows the per-class classification comparison of different methods on the first two tasks. As can be seen, our method has the highest accuracy for most classes. Fig. 6 illustrates the accuracy comparison after convergence of different methods. As can be seen, our method is not only more accurate, but also more stable.

The quantitative comparison results are presented in Tables II and III. It can be seen from Table II that, for the task N_A → UCM, our DATN yields 14.83%~1.14% higher accuracy compared to the competing methods. Regarding the task N_A → PatternNet, DATN also achieves outstanding performance with the highest overall accuracy. In particular, compared to MSDA, which has the best performance in the baselines, our DATN obtains 1.34% higher accuracy.

In Table III, there are totally six transfer tasks. As can be seen, our method has the best results in most circumstances. With regard to the average accuracy (Avg) of the six tasks, our method exceeds the best baseline (MCD) by 0.55%, while for VA → VB, the accuracy of ours is higher than MCD, by 1.27%.

At last, in Tables II and III, we can also find that, the first and second groups of experiments have better results than the third ~ eighth ones, indicating that VA, VB and VG have big differences in RS data distributions. Moreover, for the first two groups of experiments, they have the same source domain but different target domains, and the performance of the second one is slightly better than that of the first one, indicating the distribution difference between N_A and PatternNet is less than that between N_A and UCM.

D. Ablation Study

To show the effect of each innovations on DATN, a set of ablation experiments are conducted with different variants.

TABLE II
OVERALL ACCURACY (%) OF DIFFERENT METHODS ON TWO TASKS (THE BEST RESULT IS IN **BOLD**, WHICH IS SIMILAR TO THE FOLLOWING TABLES).

Type	Method	N_A → UCM	N_A → PatternNet
△	ResNet50 [8]	76.43	85.82
	SAN [3]	88.43	91.70
	IDDA [18]	88.91	91.64
	DSAN [4]	89.87	91.92
	CoGAN [9]	89.10	91.96
	LPJT [19]	89.84	91.95
	MSDA [20]	90.05	92.23
	MCD [21]	90.12	92.16
★	DATN(Ours)	**91.26**	**93.57**

△: State-of-the-art Method ★: Proposed Method

TABLE III
OVERALL ACCURACY (%) OF DIFFERENT METHODS ON OTHER SIX TASKS.

Method	VA → VB	VA → VG	VB → VA	VB → VG	VG → VA	VG → VB	Avg
ResNet50	82.41	78.32	77.33	76.04	76.92	78.31	78.22
SAN	88.82	87.01	86.09	83.13	86.24	87.37	86.44
IDDA	88.79	86.94	86.82	83.35	86.31	87.65	86.64
DSAN	89.30	87.04	88.01	83.82	86.70	88.17	87.17
CoGAN	89.55	87.02	87.07	83.79	86.87	89.00	87.22
LPJT	90.20	88.86	87.71	84.85	86.81	88.96	87.90
MSDA	90.13	89.23	88.20	85.65	86.98	89.74	88.32
MCD	90.25	89.19	88.28	**85.91**	87.76	90.07	88.58
DATN	**91.52**	**90.08**	**88.34**	85.87	**88.85**	**90.10**	**89.13**

TABLE IV
ABLATION STUDY ABOUT OVERALL ACCURACY (%) ON TWO TASKS AND COMPUTATIONAL COST. (↓ DENOTE THE SMALL IS BETTER).

Type	Method	N_A → UCM	N_A → PatternNet	*#Param* (↓)	*FLOPs* (↓)
†	DATN-c	88.51	91.76	29.05M	7.5G
	DATN-d	88.42	91.75	29.40M	7.7G
	DATN-j	89.69	92.55	27.98M	6.9G
	DATN-z	91.00	93.24	30.23M	8.0G
★	DATN(Ours)	**91.26**	**93.57**	**27.98M**	**6.9G**

†: Variant Method ★: Proposed Method

As shown in Table IV, DATN-c and DATN-d denote the variants without using dropout in either label classifiers or domain discriminators, respectively; DATN-c only contains one label classifier, while DATN-d only has one domain discriminator. DATN-j denotes a variant without using the joint distribution of features and labels. DATN-z denotes a variant without using dropout, but like DATN, it still contains multiple label classifiers and domain discriminators. As can be seen, compared to DATN, there is a big impact on DATN-d in accuracy, which reflects the importance of capturing multi-modal structures of RS images for classification. Next, DATN-c is also influenced, so it is very important to reduce the ambiguous features on class boundaries. Third, since DATN-j only relies on the single feature distribution, its accuracy decreases by more than 1%. At last, since DATN-z has multiple label classifiers and domain discriminators, it achieves better performance than DATN-c and DATN-d; but gets lower overall accuracy compared to ours, indicating that by using dropout, our DATN produces more discriminative features and thus obtains superior classification performance.

In addition, we utilize the total number of trained parameters *#Param* and the floating-point multiplication-adds *FLOPs* to measure the model complexity and the computational cost. The results of DATN and its variants on the task N_A → PatternNet are given in Table IV. Obviously, the use of dropout in DATN can bring a clear benefit for the computational efficiency.

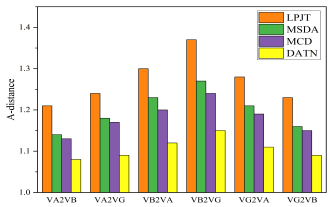

Fig. 7. Comparison $\mathcal{A}$-distance on different tasks.

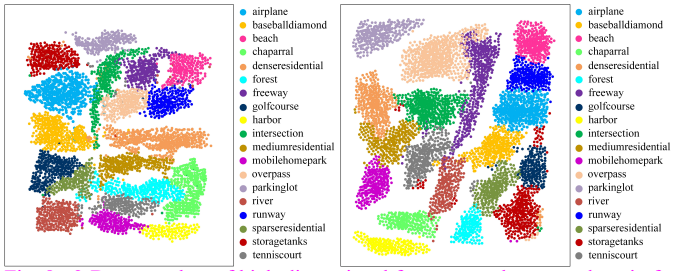

Fig. 8. 2-D scatterplots of high-dimensional features on the target domain for the task N_A → UCM. Left: t-SNE of MCD. Right: t-SNE of our DATN.

E. Distribution Discrepancy

To estimate the distribution differences between the source and target domains, we compute the $\mathcal{A}$-distance (denoted as $d_{\mathcal{A}}$) on the tasks VA → VB, VA → VG , VB → VA, VB → VG, VG → VA, and VG → VB with the features extracted by four different algorithms. The results are shown in Fig. 7. As can be seen $d_{\mathcal{A}}$ of DATN is much lower than those of LPJT, MSDA and MCD, indicating the joint distribution learned by DATN can bridge different domains and reduce the cross-domain gaps more effectively. Also, $d_{\mathcal{A}}$ of VA → VB is smaller than that of VB → VG, reflecting the domains VA and VB are similar. This also explains the higher accuracy of VA → VB in Table III.

F. Feature Visualization

To verify the advantages of DATN, we visualize the features learned on the target domain by t-SNE [19]. Taking N_A → UCM for example, we show the visualization graphs of our DATN and MCD in Fig. 8. As can be seen, MCD, which has the best performance in the baselines, cannot clearly distinguish several classes due to some overlapping features on class boundaries. In contrast, our DATN forms independent semantic clusters, in which different classes are separated on a large scale. As a result, the class boundaries are evident, the distance between classes is larger, and the feature overlaps are less.

G. The Number of Discriminators

DATN involves an adjustable parameter n, i.e., the number of dropout-based domain discriminators. To determine its optimal value, we conduct a set of experiments. We change the values of n from 0 to 200 with the interval of 1. Under each value, DATN is trained for classifying UCM and PatternNet, respectively. The obtained accuracy curve is plotted in Fig. 9. We find that, when the value of n is close to the number of classes, e.g., 19 for UCM or 22 for PatternNet, the accuracy reaches the maximum. Hence, n should be set to the number of classes in the target domain, to achieve the best results.

IV. Conclusion

This letter has presented a novel RS scene classification method based on dropout-based adversarial training networks. We designed FG to extract high-level semantic information,

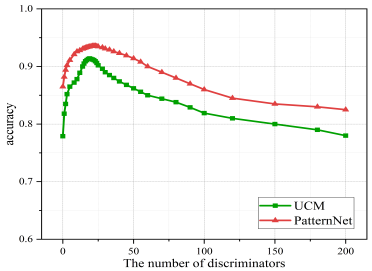

Fig. 9. Performance of DATN with different number of discriminators.

constructed DLC to suppress ambiguous features, and built DDD to capture multi-modal information for RS scene images. The proposed DATN was compared with several standard models on a number of RS data sets and the results demonstrated its effectiveness and superiority.

References

[1] X. Wang, S. Wang, C. Ning and H. Zhou, "Enhanced Feature Pyramid Network With Deep Semantic Embedding for Remote Sensing Scene Classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 9, pp. 7918-7932, Sept. 2021.
[2] C. Ma, X. Mu, R. Lin and S. Wang, "Multilayer Feature Fusion with Weight Adjustment Based on a Convolutional Neural Network for Remote Sensing Scene Classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 2, pp. 241-245, Feb. 2021.
[3] Z. Cao, M. Long, J. Wang and M. I. Jordan, "Partial Transfer Learning with Selective Adversarial Networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2724-2732.
[4] Y. Zhu, F. Zhuang, J. Wang, G. Ke, J. Chen, J. Bian, H. Xiong and Q. He, "Deep Subdomain Adaptation Network for Image Classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 4, pp. 1713-1722, April 2021.
[5] L. Guo, R. Li and B. Jiang, "An Ensemble Broad Learning Scheme for Semisupervised Vehicle Type Classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 12, pp. 5287-5297, Dec. 2021.
[6] Z. Pei, Z. Cao, M. Long and J. Wang, "Multi-adversarial Domain Adaptation," in *Proc. AAAI Conf. Artif. Intell.*, 2018, pp. 3934-3941.
[7] X. Fang, H. Bai, Z. Guo, B. Shen, S. Hoi and Z. Xu, "Domain-Adversarial Residual-Transfer Networks for Unsupervised Cross-Domain Image Classification," *arXiv:1812.11478v1*, 2018.
[8] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770-778.
[9] M. Y. Liu and O. Tuzel, "Coupled Generative Adversarial Networks," in *Adv. Neural Inf. Process. Syst.*, Dec. 2016, pp. 469-477.
[10] G. Cheng, J. Han and X. Lu, "Remote Sensing Image Scene Classification: Benchmark and State of the Art," *Proceedings of the IEEE*, vol. 105, no. 10, pp. 1865-1883, Oct. 2017.
[11] G. Xia et al., "AID: A Benchmark Data Set for Performance Evaluation of Aerial Scene Classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 3965-3981, Jul. 2017.
[12] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *Proc. ACM SIGSPATIAL GIS*, 2010, pp. 270-279.
[13] W. Zhou, S. Newsam, C. Li and Z. Shao, "PatternNet: A Benchmark Dataset for Performance Evaluation of Remote Sensing Image Retrieval," *ISPRS J. Photogramm. Remote Sens.*, vol. 145, pp. 197-209, Nov. 2018.
[14] D. Hou, Z. Miao, H. Xing and H. Wu, "Two Novel Benchmark Datasets from ArcGIS and Bing World Imagery for Remote Sensing Image Retrieval," *Int. J. Remote Sens.*, vol. 42, no. 1, pp. 220-238, Jan. 2021.
[15] D. Hou, Z. Miao, H. Xing and H. Wu, "V-RSIR: An Open Access Web-Based Image Annotation Tool for Remote Sensing Image Retrieval," *IEEE Access*, vol. 7, pp. 83852-83862, Jun. 2019.
[16] X. Lu, T. Gong and X. Zheng, "Multisource Compensation Network for Remote Sensing Cross-Domain Scene Classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 4, pp. 2504-2515, April 2020
[17] S. Saha, S. Zhao and X. X. Zhu, "Multitarget Domain Adaptation for Remote Sensing Classification Using Graph Neural Network," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1-5, 2022, Art no. 6506505, doi: 10.1109/LGRS.2022.3149950."
[18] V. K. Kurmi and V. P. Namboodiri, "Looking back at Labels: A Class based Domain Adaptation Technique," *in Proc. Int. Joint Conf. Neural Netw.*, Jul. 2019, pp. 1-8.
[19] J. Li, M. Jing, K. Lu, L. Zhu and H. T. Shen, "Locality Preserving Joint Transfer for Domain Adaptation," *IEEE Trans. Image Process.*, vol. 28, no. 12, pp. 6103-6115, Dec. 2019.
[20] M. Mancini, L. Porzi, S. R. Buló, B. Caputo and E. Ricci, "Inferring Latent Domains for Unsupervised Deep Domain Adaptation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 2, pp. 485-498, Feb. 2021.
[21] K. Saito, K. Watanabe, Y. Ushiku and T. Harada, "Maximum Classifier Discrepancy for Unsupervised Domain Adaptation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3723-3732.