Domain Adaptation for Object Classification in Point Clouds via Asymmetrical Siamese and Conditional Adversarial Network

Huan Luo, Lingkai Li, Lina Fang, Hanyun Wang, Cheng Wang, *Senior Member, IEEE*, Wenzhong Guo, *Senior Member, IEEE*, and Jonathan Li, *Senior Member, IEEE*

***Abstract*— Nowadays, researchers have developed various deep neural networks for processing point clouds effectively. Due to the enormous parameters in deep learning-based models, a lot of manual efforts have to be invested into annotating sufficient training samples. To mitigate such manual efforts of annotating samples for a new scanning device, this letter focuses on proposing a new neural network to achieve domain adaptation in 3-D object classification. Specifically, to minimize the data discrepancy of intraclass objects in different domains, an Asymmetrical Siamese (AS) module is designed to align the intraclass features. To preserve the discriminative information for distinguishing interclass objects in different domains, a Conditional Adversarial (CA) module is leveraged to consider the classification information conveyed from the classifier. To verify the effectiveness of the proposed method on object classification in heterogeneous point clouds, evaluations are conducted on three point cloud datasets, which are collected in different scenarios by different laser scanning devices. Furthermore, the comparative experiments also demonstrate the superior performance of the proposed method on the classification accuracy.**

***Index Terms*— 3-D object classification, asymmetrical Siamese (AS) network, domain adaptation, feature alignment, point clouds.**

I. Introduction

RECENT advances in deep learning technologies have enlighten researchers to develop a variety of neural networks for processing point cloud data. These neural networks have been shown to be effective at assisting machine in understanding the geometric shapes of 3-D objects [1], [2]. However, manually annotating massive samples for training neural networks is often time-consuming and labor-intensive [3]. In addition, due to the data discrepancy in the heterogeneous point clouds which are collected by different laser scanning systems, training datasets need to be created for each type of laser scanning system [4].

To ease the annotation burden for a new task, domain adaption, as a branch of machine learning, is studied extensively [5]. The goal of domain adaption is to transfer knowledge from a source domain containing sufficient annotated samples to a target domain where annotated samples are scarce or unavailable. Some pioneering studies focused on proposing domain adaption approaches to process the heterogeneous point clouds [6], [7], [8]. To achieve point cloud representation across different domains, a domain adaptation network was proposed to simultaneously implement global-level and local-level feature alignments [6]. Inspired by domain adaptation in 2-D images, a multihead network was designed to integrate the classification loss and deformation reconstruction loss for globally aligning feature descriptions, which achieves self-supervised learning in point cloud classification [7]. To reduce the domain discrepancies caused by different laser scanners, a local adversarial learning is proposed to leverage the local surface prior to assist the semantic label transfer across different domains [8]. Different laser scanning systems inevitably induce the measured point clouds with different point densities and different point distributions, resulting in severe intraclass variations in heterogeneous point clouds. Meanwhile, the local similarities of objects in different scenarios lead to the interclass similarities in heterogeneous point clouds. However, current studies neglect to explicitly preserve the intraclass similarities and interclass dissimilarities, both of which influence the performance of domain adaptation in heterogeneous point clouds.

To jointly consider the intraclass similarities and interclass dissimilarities into domain adaptation of heterogeneous point clouds, this letter mainly focuses on proposing an Asymmetrical Siamese and Conditional Adversarial Network (ASCA-Net). The new ASCA-Net contains two main components, i.e., Asymmetrical Siamese (AS) module and Conditional Adversarial (CA) module. Specifically, on the one hand, we propose to design an AS module to minimize the discrepancy of intraclass objects by aligning their latent space extracted from source domain and target domain to be as close as possible. The discrepancy minimization of intraclass objects from different domains is beneficial to preserve the intraclass similarities in the domain adaptation procedure. In addition, due to the scarce or unavailable supervised information in target domain, we propose an unsupervised method to effectively

Manuscript received 5 July 2022; revised 28 August 2022; accepted 19 September 2022. Date of publication 22 September 2022; date of current version 4 October 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 61801121, Grant 62002064, and Grant U21A20472; and in part by the National Key Research and Development Plan of China under Grant 2021YFB3600503. *(Corresponding author: Hanyun Wang.)*

Huan Luo, Lingkai Li, and Wenzhong Guo are with the College of Computer and Data Science, the Fujian Provincial Key Laboratory of Network Computing and Intelligent Information Processing, and the Key Laboratory of Spatial Data Mining and Information Sharing, Ministry of Education, Fuzhou University, Fuzhou 350003, China (e-mail: hluo@fzu.edu.cn).

Lina Fang is with the Academy of Digital China (Fujian), Fuzhou University, Fuzhou 350003, China.

Hanyun Wang is with the School of Surveying and Mapping, Information Engineering University, Zhengzhou 450000, China (e-mail: why.scholar@gmail.com).

Cheng Wang is with the Fujian Key Laboratory of Sensing and Computing for Smart Cities, Xiamen University, Xiamen 361005, China.

Jonathan Li is with the Department of Geography and Environmental Management, and the Department of Systems Design Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada.

Digital Object Identifier 10.1109/LGRS.2022.3208589

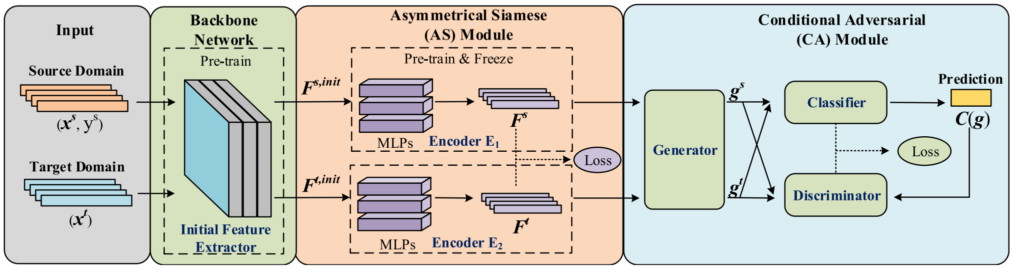

Fig. 1. Network architecture of the ASCA-Net. With the point cloud objects from the source and target domains as inputs, a pretrained backbone network is first used to extract the initial features. Then, an AS module is trained to minimize the feature discrepancy of intraclass objects. Finally, the CA module accomplishes domain adaptation by preserving the discriminative information for interclass objects.

find the intraclass objects from different domains. On the other hand, we propose to design a CA module to consider the discriminative information conveyed from the classifier predictions into domain adaptation procedure. Such discriminative information largely assists in guaranteeing the dissimilarities of interclass objects in the domain adaptation procedure. Intuitively, intraclass similarities and interclass dissimilarities can largely improve the performance of object classification in heterogeneous point clouds. The main contributions of this letter to classify objects in heterogeneous point clouds can be summarized as follows.

1) To jointly and explicitly preserve the intraclass similarities and interclass dissimilarities, we propose a new domain adaptation network called ASCA-Net, which can effectively classify objects in heterogeneous point clouds.
2) To minimize the discrepancy of intraclass objects from different domains, we design an AS module to leverage AS network to align the intraclass features in an unsupervised manner.
3) We conduct extensive experiments on three datasets collected by different laser scanners. The comparative experiments demonstrate the superior performance of ASCA-Net over other methods on classifying objects in heterogeneous point clouds.

II. Methodology

Fig. 1 presents the network architecture of the ASCA-Net to achieve object classification across different domains. To be specific, the inputs of the ASCA-Net contain 3-D objects from a source domain $D_s = \{(\boldsymbol{x}_i^s, y_i^s)\}_{i=0}^{N_s}$ and a target domain $D_t = \{(\mathbf{x}_i^t)\}_{i=0}^{N_t}$. Here, $\boldsymbol{x}_i$ and y_i represent point cloud object and its class label, respectively. N_s and N_t are the number of objects in source domain D_s and target domain D_t, respectively. Each object consists of N points, where each point is represented by 3-D coordinates (x, y, z). The goal of the ASCA-Net is to leverage the labeled objects in source domain D_s to predict the class label y^t for the unlabeled object $\boldsymbol{x}^t$ in target domain D_t. As shown in Fig. 1, the ASCA-Net is composed of three components: backbone network, AS module, and CA module.

Specifically, we first pretrain a backbone network to generate the initial feature representations of objects from the source and target domains (see Section II-A). Then, the AS module is trained to minimize the discrepancy between intraclass objects in heterogeneous point clouds (see Section II-B). Finally, the CA module is proposed to accomplish adversarial domain adaptation by considering conditional information conveyed from classifier (see Section II-C).

A. Backbone Network

The backbone network is responsible for extracting initial features for each object in different domains. Any off-the-shelf networks for point cloud feature extraction can be used as the backbone network. Here, we leverage RandLA-Net [9] as our backbone network to gradually aggregate local features for object description. The backbone network consists of four dilated residual blocks, each of which stacks multiple local spatial encoding units and attentive pooling units with a skip connection. Specifically, local spatial encoding unit encodes neighboring geometries of points which are selected by the random sampling. Attentive pooling unit aggregates the set of neighboring point features by automatically assigning different weights according to the attention mechanism. After applying dilated residual blocks, we aggregate features to one point feature, $\boldsymbol{F}_{\text{init}}$, for globally describing an object.

B. AS Module

To reduce the feature dissimilarities among the intraclass objects in heterogeneous point clouds, we propose an AS module to conduct feature alignment in the feature space. The goal of AS module is to push the latent feature space of intraclass objects in source and target domains to be as close as possible.

Inspired by the AS network [10], we design an AS module consisting of two encoders and a metric learning mechanism. We use MLPs as the encoders and the two encoders have an identical network architecture. Specifically, an encoder E_1 for source domain is first trained by adding a softmax layer to implement the classification task with the supervised information y^s. Once the training procedure terminates, the weights of the encoder will be frozen and the encoder E_1 is able to project each object in source domain into a high-dimensional codeword $\mathbf{F}^s$. Then, an encoder E_2 also maps the object from target domain into a high-dimensional codeword $\mathbf{F}^t$. Here, the lengths of $\mathbf{F}^s$ and $\mathbf{F}^t$ are identically set to 1024. We hope that the distribution of intraclass objects in different domains should be consistent by optimizing the feature matching distance. Notably, we only update the weights of the encoder in E_2 using a feature matching loss in the training stage. In the feature matching loss, the similarity between two high-dimensional feature vectors can be calculated by $L2$-Norm as follows:

$$\mathcal{L}_{\text{AS}} = (\boldsymbol{x}^s, \boldsymbol{x}^t) = \sum_{i=0}^{N_c} \|\boldsymbol{F}^s - \boldsymbol{F}^t\|_2 \tag{1}$$

where $\boldsymbol{F}^s$ and $\boldsymbol{F}^t$ represent the encoded features of objects $\boldsymbol{x}^s$ and $\boldsymbol{x}^t$, respectively. Note that the smaller value of $\mathcal{L}_{\text{fm}}$ implies the smaller discrepancy between intraclass objects in different domains.

To effectively calculate the feature matching loss in (2), we need to find the pairwise intraclass objects in advance. Due to the scarce or unavailable supervised information provided in target domain, we propose an unsupervised method to search the pairs of intraclass objects which should be mapped in the same region. Concretely, in the source domain, we first calculate the feature centroid, $\boldsymbol{F}^s_c$, for class c as

$$\boldsymbol{F}^s_c = \frac{1}{N_s}\sum_{\boldsymbol{x}_i \in D_s} \boldsymbol{F}^s_{\text{init},i} \tag{2}$$

where $\boldsymbol{F}^s_{\text{init},i}$ is the object x_i's initial feature representation obtained by the backbone network.

After that, the Euclid distance between each object in target domain and the feature centroid, $\boldsymbol{F}^s_c$, is computed. According to the Euclid distance, the pairwise objects $(\boldsymbol{F}^s_c, \boldsymbol{F}^t_{\text{init},i})$ can be determined by searching the nearest centroid of the object, x^t_i. Finally, we can optimize the matching loss function to minimize the distance between source domain and supervised target domain in different feature spaces. Note that, the pairwise intraclass objects are dynamically determined according to the updated parameters of backbone network.

C. CA Module

To preserve discriminability for intraclass objects from different domains, we design a CA module to condition domain adaption on the discriminative information conveyed in the classifier predictions. Inspired by Conditional Domain Adversarial Network (CDAN) [5], the CA module contains three components: generator G, discriminator D, and classifier C. The generator G is built by a three-layer MLP, and it is used to generate transferable feature representations to confuse the discriminator. The discriminator D aims to distinguish which domain an object comes from. The classifier C predicts class labels for objects. Hence, the CA module can be represented as a minimax optimization problem with two loss terms, i.e., the classification loss, $\mathcal{L}_C$, and the discrimination loss, $\mathcal{L}_D$.

Specifically, the classification loss, $\mathcal{L}_C$, which is used to train classifier C, is calculated by the labeled objects from source domain, and it can be formulated as follows:

$$\mathcal{L}_C = \sum_{\boldsymbol{x}^s_i \in D_s} L\left(C\left(\mathbf{g}^s_i\right), y^s_i\right) \tag{3}$$

where $L(\cdot,\cdot)$ computes the cross-entropy loss. $\mathbf{g}^s_i$ is the feature representation of object $\boldsymbol{x}^s_i$, which is obtained by generator G. $C(\cdot)$ is the class probability predicted by classifier C.

The discrimination loss, $\mathcal{L}_D$, is used to train the discriminator, D, on objects with the domain information, and it can be formulated as follows:

$$\mathcal{L}_D = \sum_{\boldsymbol{x}_i \in D_s \bigcup D_t} L(D([\mathbf{g}_i, C(\mathbf{g}_i)]), Y_i) \tag{4}$$

where Y_i is the label indicating whether the object $\boldsymbol{x}_i$ belongs to source domain or target domain. $[\cdot,\cdot]$ is an operation that concatenates two vectors. Thus, the inputs of discriminator D contain not only the feature representations generated by G, but also the classifier predictions obtained by C, which provides the discriminative information to distinguish intraclass objects during the domain adaptation. The design of $\mathcal{L}_D$ constrains the generator G to generate the discriminative features for interclass objects and to preserve the consistent features for intraclass objects.

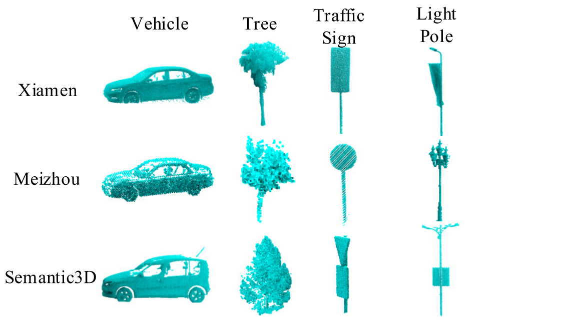

Fig. 2. Examples of *traffic sign*, *tree*, *light pole*, and *vehicle* in Xiamen, Meizhou, and Semantic3D datasets.

The training procedure of CA module iteratively implements a minimax game of two stages described as follows.

1) Train the generator G on source domain to minimize the classification loss, $\mathcal{L}_C$, while maximizing the discrimination loss, $\mathcal{L}_D$, on source and target domains for confusing the discriminator G. The minimization of $\mathcal{L}_C$ and maximization of $\mathcal{L}_D$ enable generator G to generate the discriminative features for interclass objects.

   Formally, we formulate the objective function as follows:

$$\min \mathcal{L}_{\text{CA}} = \min_{G,C}(\mathcal{L}_C - \lambda\mathcal{L}_D) \tag{5}$$

   where λ is a hyperparameter which controls the weights of the classification loss and the discriminator loss.

2) Freeze the trained generator G and update the weights of the discriminator D to minimize the discrimination loss, $\mathcal{L}_D$, as

$$\min_D \mathcal{L}_D. \tag{6}$$

   The minimization of $\mathcal{L}_D$ improves the ability of discriminator D to distinguish whether the feature belongs to source domain or target domain. The total loss of ASCA-Net is formulated as follows:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{AS}} + \alpha\mathcal{L}_{\text{CA}} \tag{7}$$

   where the factor α controls the weights of $\mathcal{L}_{\text{AS}}$ and $\mathcal{L}_{\text{CA}}$.

III. Experiment

A. Dataset

To verify the effectiveness of the proposed ASCA-Net on object classification in heterogeneous point clouds, extensive experiments are conducted on three datasets: Semantic3D [11], Xiamen [12], and Meizhou. Specifically, Semantic3D dataset is acquired by a Terrestrial Laser Scanning (TLS) system in urban outdoor scenes. Xiamen dataset is collected by the Reigl VMX450 Mobile Laser Scanning (MLS) system in Xiamen city with two laser scanners, and Meizhou dataset is collected by the Reigl VMX450 MLS system in Meizhou island with only one laser scanner. As shown in Fig. 2, intraclass objects in different datasets have obvious discrepancies. Table I presents the number of objects in five classes: *vehicle, tree, traffic sign, light pole,* and *other*. To evaluate the performance of

TABLE I

NUMBER OF OBJECTS IN THREE DATASETS

	vehicle	tree	traffic sign	light pole	other	Total
Semantic3D	67	51	18	25	20	161
Xiamen	91	339	118	53	53	601
Meizhou	132	178	128	123	47	561

object classification in heterogeneous point clouds, we conduct experiments on six domain adaption tasks, i.e., Semantic3D to Xiamen (SvsX), Xiamen to Semantic3D (XvsS), Meizhou to Semantic3D (MvsS), Semantic3D to Meizhou (SvsM), Xiamen to Meizhou (XvsM), and Meizhou to Xiamen (MvsX). In each task, we treat two datasets as source domain and target domain, respectively.

B. Experiments Setup

To train our proposed ASCA-Net, we augment the used datasets by the rotation operation and use Farthest Point Sampling (FPS) to sample objects with N points.

1) Backbone Network: To obtain initial features for objects, the used RandLA-Net contains four dilated residual blocks, which gradually aggregate the local features. Specifically, the size of the point clouds are sampled in every dilated residual block, i.e., $[K \rightarrow (N/5) \rightarrow (N/5*5) \rightarrow (N/5*5*6) \rightarrow (N/5*5*6*10)]$. Meanwhile, the dimension of the feature to describe an object is increased, i.e., $(3 \rightarrow 64 \rightarrow 256 \rightarrow 512 \rightarrow 1024)$.

2) AS Module: The AS module consists of two encoders, each of which is built by a two-layer MLPs with 1024 nodes in each layer.

3) CA Module: In the CA module, the generator G is built by a three-layer MLP of $(512 \rightarrow 256 \rightarrow 64)$; the discriminator G is built by a three-layer MLP with 1024 nodes in each layer; the classifier C is built by two fully connected layers of $(32 \rightarrow n_{\text{class}})$, where n_{class} is the number of classes in the classification task.

We implement the proposed ASCA-Net with Tensorflow and set the optimizer to Adam Optimizer. The learning rates of all modules and weight decay are set at 0.0001 and 0.05, respectively. We empirically set the factor α in (7) at 1.

C. 3-D Object Classification in Heterogeneous Point Clouds

The quantitative evaluations of our proposed ACSA-Net on object classification in heterogeneous point clouds are reported in Table II. As shown in Table II, the ACSA-Net achieves the average $F1$-Score on the six tasks at 0.805, 0.801, 0.805, 0.956, 0.973, and 0.862, respectively. This exhibits that our proposed ACSA-Net can effectively align objects from different domains in the feature space. In addition, the precision for vehicle, tree, traffic sign, and light pole reaches an average of 0.950, 0.915, 0.793, and 0.867, respectively. The precision of vehicle and tree is a little higher than that of traffic sign and light pole. This is because more severe interclass similarities and intraclass discrepancies exist in traffic sign and light pole.

TABLE II

CLASSIFICATION RESULTS ON DIFFERENT DATASETS

	Method	vehicle	tree	traffic sign	light pole	other	Avg
StoX	Precision	.992	.899	.730	.750	.666	.807
	Recall	.903	.999	.703	.800	.615	.804
	F1-Score	.945	.947	.717	.774	.639	.805
XtoS	Precision	.861	.958	.639	.877	.681	.803
	Recall	.958	.804	.846	.939	.445	.798
	F1-Score	.907	.874	.728	.907	.538	.801
MtoS	Precision	.868	.818	.739	.795	.805	.805
	Recall	.958	.991	.715	.952	.408	.805
	F1-Score	.911	.896	.726	.866	.541	.805
StoM	Precision	.995	.999	.935	.879	.976	.957
	Recall	.982	.999	.867	.939	.995	.956
	F1-Score	.988	.999	.899	.908	.985	.956
XtoM	Precision	.999	.994	.983	.994	.900	.974
	Recall	.887	.999	.992	.987	1.00	.973
	F1-Score	.939	.997	.988	.990	.948	.973
MtoX	Precision	.987	.820	.735	.910	.915	.873
	Recall	.909	.999	.825	.927	.602	.852
	F1-Score	.946	.901	.778	.918	.726	.862

TABLE III

AVERAGE CLASSIFICATION PRECISION ON DIFFERENT DATASETS

	StoX	XtoS	MtoS	StoM	XtoM	MtoX
MCD [13]	.650	.756	.740	.705	.813	.798
PointDAN [6]	.724	.720	.764	.752	.851	.809
no-AS&CA	.623	.769	.644	.640	.779	.682
no-AS	.632	.785	.668	.780	.800	.721
ASCA-Net (DGCNN)	.770	.772	.777	.945	.972	.860
ASCA-Net (RandLA)	**.807**	**.803**	**.805**	**.957**	**.974**	**.873**

To demonstrate the superiority of our proposed method, we make a comparison with two methods, i.e., the MCD method [13] and the PointDAN method [6]. As shown in Table III, we can see that the average precision of our proposed ASCA-Net preforms higher than MCD and PointDAN methods, which proves that joint consideration of intraclass similarity and interclass dissimilarity can improve the performance of domain adaption. To further evaluate the effectiveness of different proposed modules in the proposed ASCA-Net, we also compare it with two methods, i.e.: 1) the RandLA method [9] which is adopted as our backbone network (no-AS&CA) and 2) the CA domain adaptation method [5] which removes the AS module in the ACSA-Net architecture (no-AS); As shown in Table III, no-AS&CA method performs poor on classifying objects in target domain when the supervised training information only comes from source domain. This reflects that the objects in different domains have large data discrepancy, and the classification model only trained on source domain lacks of sufficient generalization to perform well in the target domain. The superior performance of our proposed method demonstrates that our proposed ACSA-Net can effectively eliminate the inconsistency in the intraclass objects and preserve discriminative information in the interclass objects from different domains. In addition, we implement the experiments on the ASCA-Net by replacing the backbone network with DGCNN [14]. Although its performance slightly decreases due to the lower classification precision of DGCNN, the comparable results of ASCA-Net (DGCNN) in Table III demonstrate the effectiveness of our proposed network architecture.

D. Visualization

As shown in Fig. 3, we use t-SNE to visualize the feature distributions of the XvsM task achieved by different methods, i.e., no-AS&CA, no-AS, and ASCA-Net methods. Furthermore, we add some point cloud objects in the t-SNE

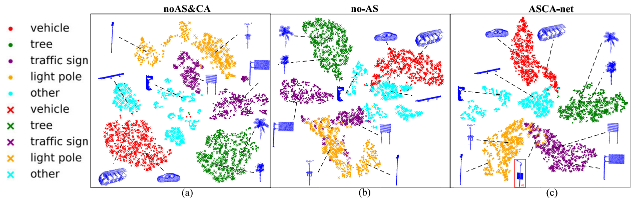

Fig. 3. t-SNE visualization of different methods applied in the XvsM task. Object in source domain and target domain is represented as dot and cross, respectively.

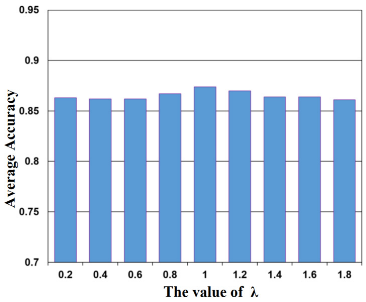

Fig. 4. Impact of hyper-parameter λ on the object classification accuracy.

visualization to better exhibit the domain adaption results. For the no-AS&CA method and the only-CA method, the objects in the source and target are discretely distributed in the feature space and across-domain features are not well-aligned. As shown in Fig. 3(c), the proposed ASCA-Net better aligns intraclass objects and better discriminates interclass objects in different domains than other methods, which guarantees the superior performance on object classification in the heterogeneous point clouds. Additionally, the light pole in region 1 is misclassified as signpost because of similar geometric shapes.

E. Parameter Sensitivity Analysis

To examine the impact of the hyper-parameter λ on the performance of 3-D object classification in heterogeneous point clouds, the evaluations on the XvsS task are performed on the following configurations: 0.2, 0.4, 0.6, 0.8, 1.0, 1.2, 1.4, 1.6, 1.8. As shown in Fig. 4, with the value of λ increasing from 0.2 to 1.0, the average accuracy gradually increases. The average accuracy reaches the peak when λ equals to 1.0. When the value of λ is larger than 1.0, the average accuracy decreases slowly. This means that when conducting the feature alignment in the CA module, the classification loss and the discriminator loss are equally important. Therefore, we set the hyper-parameter λ to be 1.0 in the experiments.

F. Computational Complexity

In the experiments, we implemented the ASCA-Net in Tensorflow using Keras API, and executed it on a HP workstation with 8 Intel cores of 2.1-GHz, 128 GB memory, and a P100 graphics card. To analyze the time complexity of our proposed ASCA-Net, we recorded the running time of each module. During the training phase, the backbone network, AS module, and CA module take an average of 21.3, 1.5, and 4.2 min, respectively. During the testing phase, the ASCA-Net takes only 8.2 ms to classify an object. Moreover, we also recorded the running time of other methods for comparison. Specifically, the training phase of PointDAN and MCD methods takes 86.7 and 64.2 min, respectively. In addition, for classifying an object, the PointDAN and MCD methods take 10.4 and 10 ms, respectively. This demonstrates that our proposed method requires less time in the training and testing phases.

IV. Conclusion

This letter has proposed the ASCA-Net, which is an efficient network architecture to solve 3-D object classification in the heterogeneous point clouds. To minimize the data discrepancy of intraclass objects in different domains, we design an AS module to implement the intraclass feature alignment. To preserve the discriminative information for guaranteeing the dissimilarities of interclass objects in different domains, a CA module is integrated into our network to consider the classification information conveyed from classifier. The proposed ASCA-Net has been evaluated on six tasks built on three different point cloud datasets. The experimental evaluations have exhibited that the proposed ASCA-Net can perform well on object classification in heterogeneous point clouds.

References

[1] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–10.

[2] R. Huang, Y. Xu, and U. Stilla, "GraNet: Global relation-aware attentional network for semantic segmentation of ALS point clouds," *ISPRS J. Photogramm. Remote Sens.*, vol. 177, pp. 1–20, Jul. 2021.

[3] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell, "Deep domain confusion: Maximizing for domain invariance," 2014, *arXiv:1412.3474*.

[4] H. Luo, C. Wang, Y. Wen, and W. Guo, "3-D object classification in heterogeneous point clouds via bag-of-words and joint distribution adaption," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 12, pp. 1909–1913, Dec. 2019.

[5] M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Conditional adversarial domain adaptation," in *Proc. NeurIPS*, 2018, pp. 1–11.

[6] C. Qin, H. You, L. Wang, C.-C. J. Kuo, and Y. Fu, "PointDAN: A multiscale 3D domain adaption network for point cloud representation," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 7192–7203.

[7] I. Achituve, H. Maron, and G. Chechik, "Self-supervised learning for domain adaptation on point clouds," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 123–133.

[8] L. Yi, B. Gong, and T. Funkhouser, "Complete & label: A domain adaptation approach to semantic segmentation of LiDAR point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 15363–15373.

[9] Q. Hu *et al.*, "RandLA-Net: Efficient semantic segmentation of large-scale point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11108–11117.

[10] Q. Guo, W. Feng, C. Zhou, R. Huang, L. Wan, and S. Wang, "Learning dynamic Siamese network for visual object tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1763–1771.

[11] T. Hackel, N. Savinov, L. Ladicky, J. Wegner, K. Schindler, and M. Pollefeys, "semantic3D.net: A new large-scale point cloud classification benchmark," *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. 4, pp. 91–98, May 2017.

[12] H. Luo *et al.*, "Semantic labeling of mobile LiDAR point clouds via active learning and higher order MRF," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 7, pp. 3631–3644, Jul. 2018.

[13] 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.*, Jun. 2018, pp. 1–10.

[14] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph CNN for learning on point clouds," *ACM Trans. Graph.*, vol. 38, no. 5, pp. 1–12, Nov. 2019.