Cross-Scene Classification of Remote Sensing Images Based on General-Specific Prototype Contrastive Learning

Puhua Chen, *Senior Member, IEEE*, Yu Qiu, Lei Guo, Xiangrong Zhang, *Senior Member, IEEE*, Fang Liu, *Senior Member, IEEE*, Licheng Jiao, *Fellow, IEEE*, and Lingling Li, *Senior Member, IEEE*

***Abstract*—In recent years, constrained by the challenges associated with expensive data annotation and poor generalization ability in supervised models, domain adaptation has been proposed to effectively mitigate domain gaps, which has gained significant attention in the field of remote sensing. However, most existing methods do not consider the inherent characteristics of remote sensing images when formulating adaptive strategies. In addition, there has been limited research on addressing class-imbalanced data situations, leading to undesirable performance in domain adaptation tasks involving long-tailed datasets. To overcome the aforementioned limitations, based on the analysis of intraclass diversity and intradomain style differences of remote sensing images, we propose a novel prototype contrastive learning framework called general-specific prototype contrastive learning (GSPCL). Due to the unreliability of clustering samples in conventional prototype-based clustering methods, the bidirectional weighted prototype strategy is proposed to optimize this loophole. Consequently, more robust prototypes are constructed in both domains, serving as mediators to reduce domain discrepancies bidirectionally. Particularly, often overlooked in most methods, we incorporate low-confidence sample features into the contrastive learning process alongside these prototypes to further guide the model to address feature alignment and long-tail issues effectively. Finally, in order to verify the superiority of our proposed method, we adhere to two existing experimental settings and construct an extra optical remote sensing domain adaptation dataset with class-imbalanced scenarios. In the first two experimental settings, GSPCL outperforms the second-ranked approach by 5.0% and 4.0% in average accuracy. Furthermore, our approach exhibits highly competitive results in handling long-tailed data scenarios.**

***Index Terms*—Contrastive learning, cross-scene classification, domain adaptation, prototype learning, remote sensing image.**

I. Introduction

REMOTE sensing scene classification, as an essential tool for interpreting remote sensing images, aims to categorize images into different scene classes by analyzing the features of objects in the images. In recent years, the rapid development of deep learning techniques, particularly the application of convolutional neural networks, has significantly advanced remote sensing scene classification and garnered widespread attention in the academic community [1], [2], [3], [4], [5]. However, the success of deep learning models relies heavily on time-consuming and expensive data annotation. Moreover, these supervised models often exhibit poor generalization when facing different distributed data [6]. Therefore, domain adaptation has been proposed to address the challenge of improving model generalization in scenarios where only labeled images from the source-domain and unlabeled target-domain images are available, allowing the model to perform more accurately on various downstream tasks in the target domain, including the cross-domain classification task addressed in this study. Existing domain adaptation methods mainly focus on learning domain-invariant representations, whether applied to conventional images or remote sensing images, which can be categorized into metric-based approaches [7], [8], adversarial-based approaches [9], [10], and reconstruction-based approaches [11], [12].

For optical remote sensing images, the inconsistency in feature distribution among datasets arises from differences in conditions, such as illumination, reflectance, and geographical location, during data collection. Therefore, a model trained on one dataset often struggles to achieve satisfactory performance on another dataset. Domain adaptation methods have been introduced to the field of remote sensing, and researchers have developed various domain adaptation methods for downstream tasks, such as cross-scene classification in remote sensing images. However, existing domain adaptation methods [13], [14], [15] for remote sensing images face several challenges. First,

Manuscript received 2 January 2024; revised 13 February 2024; accepted 9 March 2024. Date of publication 20 March 2024; date of current version 16 April 2024. This work was supported in part by the Fundamental Research Funds for the Central Universities (No. ZYTS23060), in part by the National Natural Science Foundation of China under Grant 62076192 and Grant 62276197, in part by the Key Research and Development Program in Shaanxi Province of China (No. 2019ZDLGY03-06), in part by the State Key Program of National Natural Science of China under Grant 61836009, in part by the Program for Cheung Kong Scholars and Innovative Research Team in University (No. IRT_15R53), in part by The Fund for Foreign Scholars in University Research and Teaching Programs (the 111 Project) under Grant B07048, in part by the Key Scientific Technological Innovation Research Project by Ministry of Education, and in part by the National Key Research and Development Program of China. *(Corresponding author: Puhua Chen.)*

Puhua Chen, Yu Qiu, Xiangrong Zhang, Fang Liu, Licheng Jiao, and Lingling Li are with the Key Laboratory of Intelligent Perception and Image Understanding, Ministry of Education, Xidian University, Xi'an 710071, China, also with the Joint International Research Laboratory of Intelligent Perception and Computation, International Research Center for Intelligent Perception and Computation, Xidian University, Xi'an 710071, China, and also with the School of Artificial Intelligence, Xidian University, Xi'an 710071, China (e-mail: phchen@xidian.edu.cn).

Lei Guo is with the School of Artificial Intelligence, Xidian University, Xi'an 710071, China, and also with General Design Department No. 5, Xi'an Electronic Engineering Research Institute, Xi'an 710071, China.

Digital Object Identifier 10.1109/JSTARS.2024.3379437

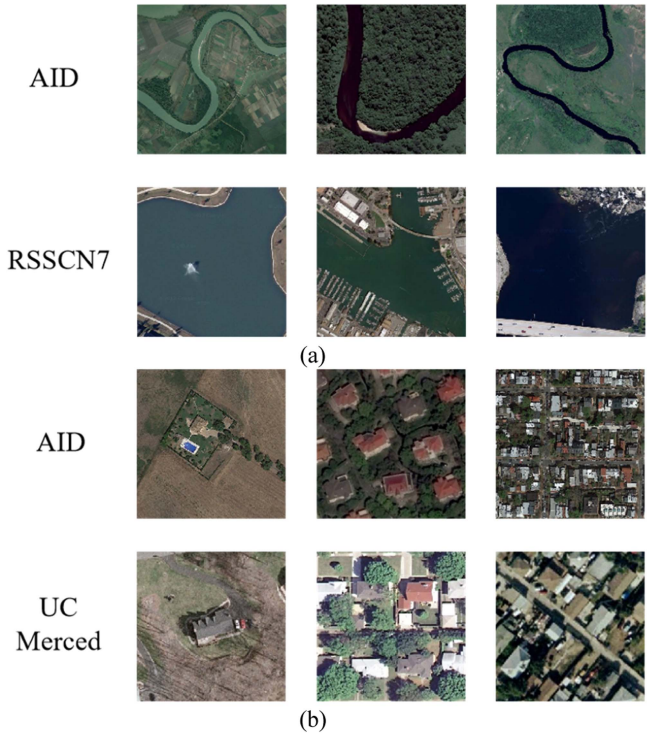


Fig. 1. Samples from various optical remote sensing scene classification datasets. (a) Water bodies from the AID dataset and the RSSCN7 dataset. (b) Residential areas from the AID dataset and the UCM dataset. The sequence from left to right comprises sparse residential, medium residential, and dense residential.

they have not achieved optimal results, indicating insufficient robustness. Second, many methods have not analyzed the specific characteristics of optical remote sensing image datasets to select suitable methods and operations for these images. Instead, they are generally applicable to both general and remote sensing images. In addition, the collection of remote sensing image data is not as straightforward as regular images, often leading to the issue of imbalanced dataset categories. Research in this specific domain is noticeably lacking, and most domain adaptation methods exhibit suboptimal performance in such circumstances.

To address the aforementioned challenges, we first analyze the features of optical remote sensing images. As shown in Fig. 1(a), water bodies in the aerial image dataset (AID) [16] dataset appear as slender stripes, while in the RSSCN7 [17] dataset, most water bodies cover large irregular areas or are located around ports, reflecting the inconsistent feature distribution between datasets caused by differences in image capture locations. Additionally, some datasets contain similar or finer categories. As shown in Fig. 1(b), the UC Merced (UCM) [18] and AID datasets further subdivide residences into dense residences, medium residences, and sparse residences, highlighting the intraclass diversity within a single category. Furthermore, compared with common optical image-domain adaptation scenarios between real and virtual scenes or commercial products and clip art, the domain gap in remote sensing images is less significant. Considering these factors, we believe prototype learning serves as an excellent tool to address domain adaptation challenges in optical remote sensing images, which just utilizes feature prototypes to determine sample categorization, demonstrating a low probability of misclassification in remote sensing images. Meanwhile, the intraclass multiprototype strategy is employed to address intraclass diversity as one of the fundamental principles guiding our approach.

In deep neural networks, the shallow layers extract low-level generic features from images, such as color and texture, while the deeper layers capture high-level semantic information relevant to classification [19]. In the subsequent discussion, we refer to these two types of features as category-general features and category-specific features, respectively. This fact has found broad applications in domain adaptation [8] and style transfer tasks [20], but some approaches only operate directly on these two types of features. In our method, we construct prototypes for both types of features in the source domain, while prototypes for the target domain are generated using high-confidence target-domain sample features through an improved prototype generation strategy. Regarding the application of contrastive learning in domain adaptation, most methods opt for high-confidence samples due to their reliability. However, inspired by Zhang et al. [21], utilizing high-confidence samples for contrastive learning often introduces semantic conflicts due to erroneous negative sample construction. It is highly likely that samples from the same class are treated as negative samples without considering the consistency of semantic information. On the other hand, low-confidence samples exhibit smaller within-class similarities and larger between-class similarities, making them suitable for contrastive learning to mitigate semantic conflicts without relying on categories. In general-specific prototype contrastive learning (GSPCL), we achieve this by employing category-general prototypes from the source domain to construct positive and negative samples for contrastive learning.

To assess the success and robustness of our model in adaptive classification, we construct three domain adaptation dataset settings using seven optical remote sensing datasets, each progressively increasing in difficulty. The results demonstrate a significant improvement in GSPCL compared with the existing approaches. Furthermore, the success of our model highlights the superiority of prototype learning in addressing domain adaptation challenges in remote sensing images. The specific contributions of this study can be outlined as follows.

1) A novel prototype contrastive learning framework is developed, leading to construct more diverse and robust prototypes for both the source and target domains, along with the formulation of new prototype losses. Besides, we integrate these prototypes with the inherent characteristics of low-confidence sample features to design novel contrastive losses, leading to improved performance compared with the utilization of high-confidence samples.
2) A new prototype generation strategy is proposed, named the bidirectional weighted prototype (BWP) strategy. For the source domain, we develop category-general prototypes and category-specific prototypes to better leverage self-supervised learning and contrastive learning methods.

3) Besides, a new class-imbalanced remote sensing dataset setting is designed to investigate the effectiveness of our approach under such conditions. Furthermore, we judiciously introduce consistency learning and entropy loss, which, combined with our method, achieved excellent results in addressing domain adaptation challenges in optical remote sensing images.

The rest of this article is organized as follows. Section II gives a brief introduction to related works. The detailed statements of the proposed method are shown in Section III. In Section IV, abundant experiments' results and analysis are shown to demonstrate the performance of the proposed method. Finally, Section V concludes this article.

II. Related Works

A. Domain Adaptation

Domain adaptation focuses on transferring knowledge learned from one or multiple source domains to one or multiple target domains in order to enhance the model's generalization capability. Mainstream domain adaptation methods can be categorized into three major categories. The first category involves metric-based methods, which entail selecting or introducing specific layers as adaptation layers. These methods incorporate metrics, such as MMD [22], and its enhancements as regularization terms to align the feature distributions between the source and target domains [23], [24], [25]. Recent metric-based adaptation approaches, such as GDCAN [8], take into consideration that the transferability of knowledge may vary with changes in convolutional layers. Consequently, they utilize attention mechanisms to extract statistical information and determine whether different layers require adaptation. The second category comprises adversarial-based methods, which draw inspiration from the principles of adversarial generative networks [26], [27], [28]. These methods introduce domain discriminators to distinguish sample features between two domains. However, the features extracted by the feature discriminator aim to confound the domain discriminator, making it indistinguishable whether the features originate from the source or target domain. The third category of methods pertains to reconstruction-based approaches. DSN [29] posits that domain features are composed of both shared and private components, which can be reconstructed. Loss functions are employed to encourage similarity between the shared features of the two domains while promoting dissimilarity between their respective shared and private components. Furthermore, there are other methods, such as MCC [30] and Leco [31], which primarily focus on the label space, while ECACL [32] and DaC [33] incorporate consistency loss to address the domain adaptation problem.

In the context of adaptive research pertaining to optical remote sensing image classification, both attention-based multiscale residual adaptation network (AMRAN) [13] and ADA-DDA [14] primarily employ attention mechanisms CBAM [34] and CMMD [35] loss to tackle classification adaptation. AMRAN places greater emphasis on addressing the issue of varying target object scales within images, introducing a multiscale adaptation module to mitigate this concern, while ADA-DDA further explores the automatic balancing of the relative importance between marginal distribution and conditional distribution alignment. However, these methods have not yet achieved satisfactory results.

B. Prototype Learning

Prototype learning aims to acquire category-specific feature prototypes. Given that prototypes can serve as representative features for classes, they can be employed in similarity or distance computations with unlabeled data, facilitating the assignment of pseudolabels [36]. This aligns well with the self-training concept, and therefore, prototype learning and self-training often co-occur. In the context of domain adaptation, prototype learning can be utilized to supplement missing label information in the target domain [37]. Chen et al. [38] propose a stepwise feature alignment network that gradually selects reliable pseudolabels based on cosine similarity, achieving domain alignment by aligning prototypes between the source and target domains. Similarly, methods, such as [39] and [40], also utilize averaging of latent features to construct class prototypes. On the other hand, PCT [41] avoids the computational cost associated with prototype construction by parameterizing prototypes using linear classifier weights. Some approaches rely on clustering methods to construct prototypes, such as SHOT [42] and SHOT++ [43], which draw inspiration from the weighted clustering approach in DeepCluster [44] to generate feature prototypes and assign pseudolabels for the target domain. BMD [45], however, recognized that directly clustering pseudolabels assigned to the target-domain by the source-domain model results in category-biased prototypes. To address this, it introduced class-balanced sampling strategies and intraclass multicenter prototype strategies to obtain more reliable prototypes.

C. Contrastive Learning

Contrastive learning focuses on capturing shared characteristics among similar instances and discerning differences among dissimilar instances, achieving remarkable performance in self-supervised learning. Wu et al. [46] introduced the concept of instance discrimination tasks and a memory bank. This method emphasizes that the images of objects with high visual similarity tend to receive similar classification results, regardless of their semantic labels. In contrast to [46], MOCO [47] introduced a queue as an additional array structure to replace the memory bank for storing negative samples, thus framing previous contrastive learning methods as dictionary lookup problems. On the other hand, SwAV [48] combined contrastive learning with clustering and introduced a multicrop strategy to increase the number of views, thereby enhancing the consistency of cluster assignments for different views of the same image. However, Tschannen et al. [49] pointed out that excessively strong instance discrimination capabilities can negatively impact downstream tasks, especially when too many negative pairs include semantically similar samples that should not be pushed apart. Therefore, PCL [50] introduced a new theoretical framework based on the expectation-maximization algorithm, improving the common

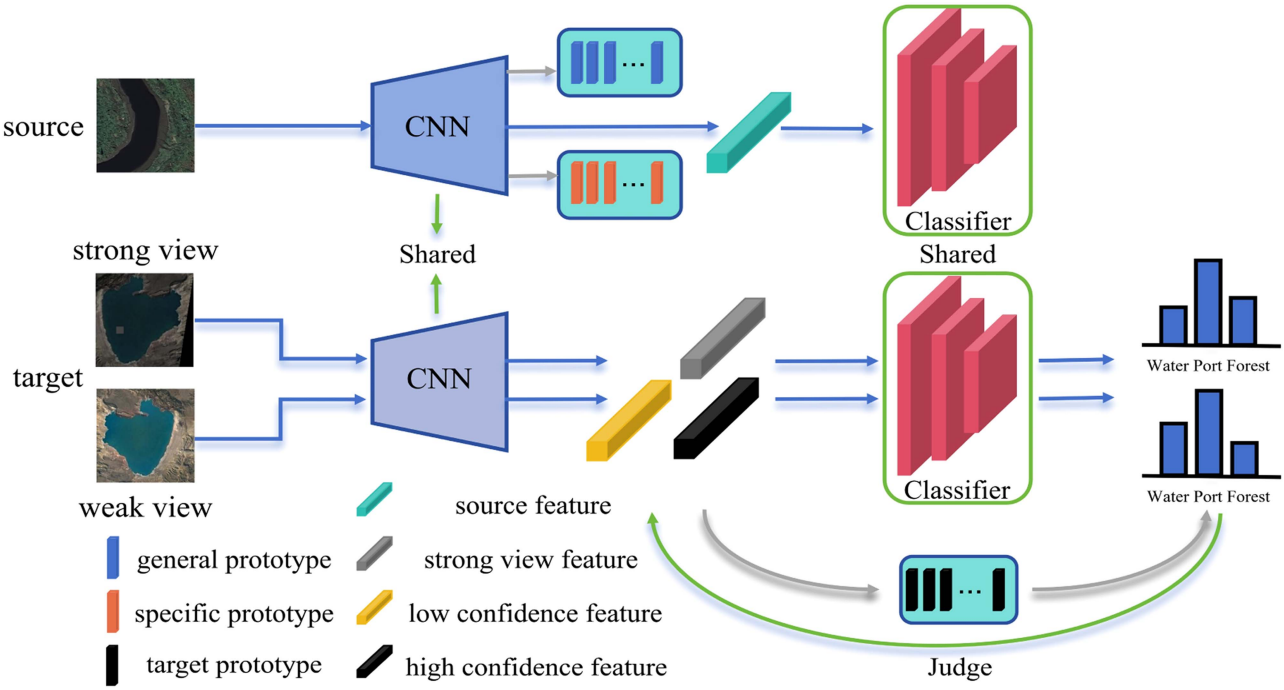


Fig. 2. Fundamental architecture of GSPCL. Blue lines represent the forward computation process of the model, while green lines indicate aspects related to parameters, and gray lines pertain to prototype generation. Three types of features and three types of prototypes are constructed and effectively utilized in the model.

InfoNCE [51] loss in contrastive learning to ProtoNCE. Specifically for domain adaptation methods that incorporate contrastive learning, Li et al. [52] considered the deviation between features and class weights. It proposed a new self-supervised paradigm for contrastive learning in domain adaptation. MixLRCo discovered that low-confidence samples have significantly lower intraclass similarity compared with high-confidence samples, which were contrary to interclass similarity. It explored the possibility of applying contrastive learning to low-confidence samples. Our approach similarly does not discard low-confidence samples, unlike most domain adaptation methods that combine self-training.

III. Proposed Method

This section primarily presents the specific methods we proposed for the task of domain adaptive classification in optical remote sensing images. It mainly includes the application of techniques, such as prototype learning, contrastive learning, and consistency learning in this task, along with a summary of the final losses. To provide a clearer exposition of our work, the architectural diagram of the method is illustrated in Fig. 2. Our approach first constructs prototypes through the WBP strategy, and then formulates prototype loss and contrastive loss based on the prototypes and features, respectively. In addition, since our method involves consistency learning, the "strong view" and "weak view" in Fig. 2 represent strong and weak augmentations applied to the target-domain images, ultimately promoting model robustness. Below, our proposed approach will be discussed in detail.

A. Preliminary

In this work, our primary focus is on the problem of unsupervised domain adaptation in optical remote sensing image classification tasks. In the conventional unsupervised domain adaptation task setting, we are provided with a source-domain dataset comprising n_s samples and a target-domain dataset comprising n_t samples, denoted as $D_s = \{(x_s^i, y_s^i)\}_{i=1}^{n_s}$ and $D_t = \{(x_t^i, y_t^i)\}_{i=1}^{n_t}$, respectively. Here, y^i represents the label for the corresponding image sample x^i. While the label spaces for the source and target domains are the same, their data spaces differ. During training, we can only predict labels for the target domain using labeled data from the source domain and unlabeled data from the target domain. Access to target-domain labels is unavailable while training and is only used for evaluation during the testing phase. In our approach, we commence by training a source-domain model using the source-domain data. This model comprises two main components: a feature extractor $f_s = X_s \rightarrow R^d$ and a classifier $g_s = R^d \rightarrow R^c$. The model's output is represented as $h_s(x^i) = g_s(f_s(x^i))$, where the feature extractor consists of commonly used classification backbone networks ResNet [53] and a domain adaptation-specific component, often referred to as an adaptation layer, which is commonly used for dimensionality reduction. Here, d represents the dimensionality of the extracted features, while c denotes the number of classes. Therefore, the objective of unsupervised domain adaptation is to learn a model $h_t = X_t \rightarrow Y_t$ that enhances the model's generalization capability in the target domain, achieving good classification results.

It is worth noting that inspired by the success of consistency learning in semi-supervised learning and domain adaptation,

we also introduce consistency learning to address the domain adaptation classification challenge in optical remote sensing images. For any given sample x_s^i in the target domain, we apply both weak augmentation and strong augmentation. Hence, in fact, for target-domain images, two distinct views of each sample are employed for training: $T_w(x_t^i)$ and $T_s(x_t^i)$, where T_w and T_s represent the weak transform and strong transform, respectively. T_w includes common operations, such as random cropping and random horizontal flipping, while T_s is executed using RandAugment.

B. Prototype Generation for Source Domain

As mentioned earlier, in our approach, we consider shallow-level features and deep-level features in deep neural networks as category-general features and category-specific features, respectively. Since GSPCL employs ResNet as the backbone, here, the shallow level refers to the first stage of ResNet, while the deep level refers to the fourth stage of ResNet or the dimension reduction layer within the entire feature extractor. Through experimentation, we ultimately select the dimension reduction layer, which not only facilitates dimensionality reduction but also extracts deeper and more relevant classification-related features, resulting in improved performance.

To obtain optimal category-general prototypes and category-specific prototypes, we adapt a two-step training approach. Initially, we pretrain a classification model on the source domain. During the adaptation phase on the target domain, we leverage the pretrained source-domain model to generate prototypes using source-domain images, which will later serve for the contrastive learning task. Similar to SHOT and BMD, we start by employing label smoothing and the following cross-entropy loss to train the source-domain model

$$L_s\left(f_s; X_s, Y_s\right) = -\mathbb{E}_{(x_s,y_s)\in X_s\times Y_s} \times \sum_{k=1}^{K} q_k \log \delta_k\left(h_s\left(x_s\right)\right) \tag{1}$$

where $\delta_k(\cdot)$ denotes the kth element in the softmax output, and q_k is "1" for the correct class, otherwise "0."

Taking into consideration that shallow features to some extent represent the stylistic characteristics within the source domain, and individual classes in the data still exhibit differences in fine-grained category features in order to avoid the inadequacy of coarse-grained single-source-domain prototypes in effectively representing the diversity and ambiguity present in the domain, for each class in the source domain, we only construct a single-source-domain prototype for the category-general features of the source domain, with the number of category-specific prototypes set to m for each class. More specifically, given the label information available in the source domain, we obtain prototypes for each class through k-means clustering. We formalize the category-general prototypes and category-specific prototypes as Pa and Ps, respectively. Their computational formulae are given as follows:

$$P\,a_i = K\underset{n=i}{\text{means}}\ \left(f_s^1\left(x_s^n\right)\right) \tag{2}$$

$$\left\{Ps_i^j\right\}_{j=1}^{m} = K\underset{n=i}{\text{means}}\ \left(f_s\left(x_s^n\right)\right) \tag{3}$$

where Pa_i is the category-general feature prototype of class i, Ps_i^j is the jth category-specific prototype of class i, and f_s^1 is the first stage of ResNet.

C. Prototype Generation for Target Domain

Our BWP strategy for the target domain is derived from a comprehensive analysis of various methods related to prototype construction through clustering. The prototype acquisition strategy in SHOT and SHOT++ [43] is inspired by DeepCluster, in essence, it involves obtaining softmax probabilities for target-domain instances x_t belonging to class i through the source-domain model and, subsequently, performing weighted k-means clustering to obtain prototype c_i^0 for each class in the target domain

$$c_i^0 = \frac{\sum_{x_t\in X_t} \delta_i\left(\hat{h}_t\left(x_t\right)\hat{f}_t\left(x_t\right)\right)}{\sum_{x_t\in X_t} \delta_i\left(\hat{h}_t\left(x_t\right)\right)} \tag{4}$$

where $\hat{h}_t = \hat{f}_t \circ \hat{g}_t$ denotes the previously learned target model.

Considering prototypes as classifiers, we compute the cosine distances $D_f(\cdot)$ between a specific sample feature in the target domain and various prototypes, thereby reassigning pseudolabel $\hat{y}_t$ to that sample

$$\hat{y}_t = \arg\min_i D_f\left(\hat{f}_t\left(x_t\right), c_i^0\right). \tag{5}$$

This process can be iteratively continued, meaning that new pseudolabel $\hat{y}_t'$ can be obtained by weighted clustering based on the newly acquired prototype c_i^1, as illustrated in the following equation:

$$c_i^1 = \frac{\sum_{x_t\in X_t} 1\left(\hat{y}_t = i\right)\hat{f}_t\left(x_t\right)}{\sum_{x_t\in X_t} 1\left(\hat{y}_t = i\right)} \tag{6}$$

$$\hat{y}\prime_t = \arg\min_i D_f\left(\hat{f}_t\left(x_t\right), c_i^1\right) \tag{7}$$

where $1(\cdot)$ is an indicator function.

However, we find that only one iteration is sufficient to obtain sufficiently good prototypes in these methods. Therefore, in the subsequent sections, we only describe the case of a single iteration and avoid unnecessary redundancy.

The class-balanced prototype (BP) strategy in BMD argues that the horizontal argmax approach used in SHOT tends to favor classes that are easily transferable to the extent that, in extreme cases, the number of target-domain samples used to construct clustering prototypes for a particular class is reduced to 0, as shown in Fig. 3(a). The class-balanced sampling strategy can be considered as a vertical sampling approach, as illustrated in Fig. 3(b). It involves considering the softmax values for all samples in the target domain. Consequently, the top-N $\delta_i(h_t(x_t))$ scores for all instances in class i on the target domain D_t are used for weighted clustering. The formula for assigning pseudolabels

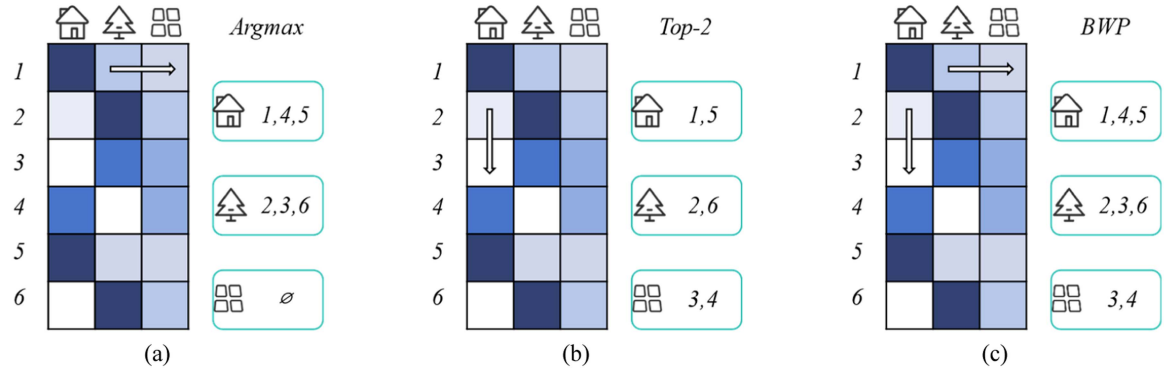


Fig. 3. Example of prototype generation strategies in remote sensing scene classification. From left to right, the strategies include the class-biased strategy (left), the BMD class-balanced strategy (middle), and the BWP strategy proposed by us (right).

$\hat{y}_t$ in BMDs prototype clustering is given as follows:

$$M_i = \underset{\substack{x_t \in X_t \\ |M_i|=N}}{\arg\max} \left(\hat{h}_t \left(x_t \right) \right)$$
$$c_i = \frac{1}{N} \sum_{i \in M_i} \hat{f}_t \left(x_t^i \right)$$
$$\hat{y}_t = \underset{i}{\arg\min}\, D_f \left(\hat{f}_t \left(x_t \right), c_i \right) \tag{8}$$

where M_i represents the set of samples used to create prototypes for the ith class.

Our strategy further addresses potential issues in the class-balanced sampling strategy. As shown in Fig. 3(b), while BMD strategy ensures that residential, forest, and agriculture classes all have samples available for prototype clustering, the reliability of the samples used for clustering remains questionable. Specifically, for the agriculture class, it utilizes samples 3 and 4 for clustering. However, based on the softmax outputs in Fig. 3(b), samples 3 and 4 are more likely to belong to the residential and forest classes, respectively. If the ground truth corresponds to residential and forest, then clustering for the agriculture class introduces incorrect samples, while the residential and forest class lose their originally assigned clustering samples, leading to prototype biases.

Therefore, our BWP strategy builds upon the horizontal argmax strategy by introducing a threshold and combining it with the BP strategy. As shown in Fig. 3(c), we extract sample index sets corresponding to both strategies and perform clustering. Since both strategies select samples that are most likely to belong to a particular class, there is an overlap in the sample indices. Thus, we take the union of the two index sets. To avoid potential issues where the introduction of the horizontal argmax strategy leads to significant differences in the number of clustering samples for each class, we impose a limit on the clustering quantity. Specifically, when the size of the union exceeds α times that of the BP strategy, we restrict the total number of clusters to α times that of the BP strategy. In a formal context, we represent the feature sets under the argmax strategy with introduced thresholds as S_h, the feature sets under the BP strategy as S_v, and the feature sets under the strategy we propose as S'_{bi}

$$S_h = \left\{ f\left(x_t^i\right) \left| \text{argmax} \delta_k \left(h_t \left(x_t^i \right) \right) \right\rangle \sigma \right\}$$
$$S_v = \left\{ f\left(x_t^i\right) \left| \text{top } N \delta_k \left(h_t \left(x_t \right) \right) \right) \right\}$$
$$S_{bi} = S_h \cup S_\nu \tag{9}$$
$$S\prime_{bi} = \begin{cases} \text{top } \alpha \cdot |S_v| \text{ in } S_{bi} & \text{if } |S_{bi}| \geq \alpha \cdot |S_v| \\ S_{bi} & \text{else} \end{cases} \tag{10}$$

where $|\cdot|$ represents the cardinality of the feature set, and σ is the threshold used to divide high- and low-confidence levels.

Simultaneously, since intraclass data in the target domain also exhibits fine-grained category differences, similar to the specific class prototypes in the source domain, we also set the number of intraclass prototypes in the target domain to m. This is done to obtain more robust and diverse target-domain prototypes, which subsequently aid in assigning pseudolabels to the samples. The formula for constructing prototypes in the target domain is given as follows:

$$\left\{ Pt_i^j \right\}_{j=1}^{m} = K \underset{n=i}{\text{ means }} \left(f_t \left(x_t^n \right) \right), x_t^i \in S\prime_{bi} \tag{11}$$

where Pt_i^j is the jth prototype of class i in the target domain.

D. Prototype Contrastive Learning Strategy

The fundamental task of domain adaptation is to reduce the distribution discrepancy between the source and target domains or to bring the feature mappings of the source and target domains closer together in high-dimensional space. Prototype learning, characterized by its interpretability and generalization capabilities, leverages class representatives. Therefore, after constructing category-specific prototypes in the source domain and target domain, we utilize prototypes as intermediaries to bidirectionally align the source- and target-domain features, rather than solely employing prototype learning for pseudolabeling the target domain. The losses related to this part are shown in Fig. 4.

On the one hand, we bring the source-domain features closer to the target-domain prototypes. Given input features obtained from the feature extractor for both the source and target domains as $F_s = f(x_s^i)$ and $F_t = f(T_w(x_t^i))$, where a minibatch of source- and target-domain features has dimensions of $b \times d$, and

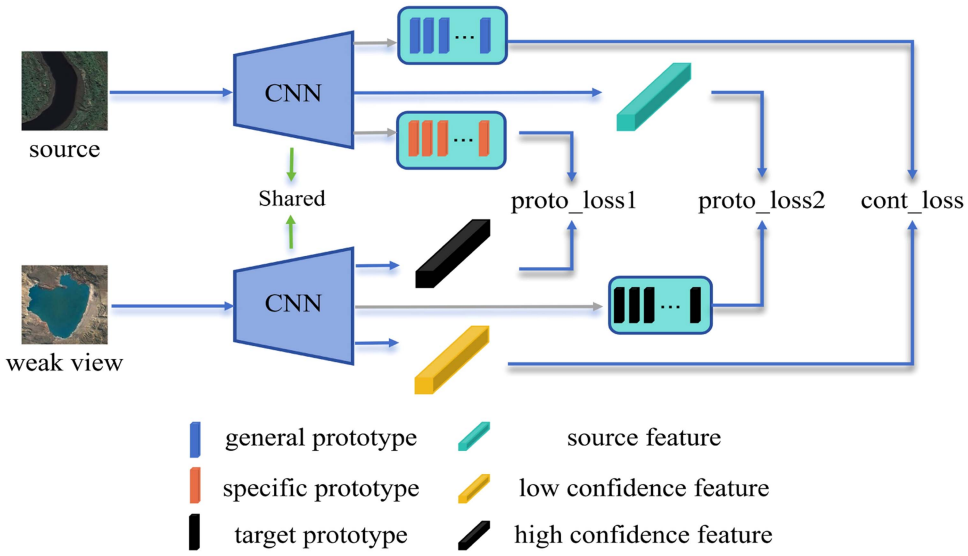


Fig. 4. Primary losses in GSPCL. Proto_loss1 utilizes classification-specific prototypes in the source domain and features of high-confidence target-domain samples for calculation, while proto_loss2 calculates based on source-domain features and target-domain prototypes; their sum aims to reduce the domain gap. Positive and negative samples are constructed using source-domain category-agnostic prototypes and features of low-confidence samples to compute the contrastive loss (cont_loss).

the dimensions of source-domain prototypes and target-domain prototypes are $c \times m \times d$. Here, b, d, c, and k represent the mini-batch size, feature dimension of feature extractor, the number of classes, and the number of intraclass prototypes, respectively. It is worth noting that our collection of category-general features and category-specific features is performed at different stages: the first stage of ResNet and the dimension reduction layer bottleneck in the feature extractor. Therefore, in terms of dimensions, the prototype features are identical to the image features during the forward computation process. Given that we have access to the labels of source-domain samples, it is straightforward to identify the prototypes corresponding to each class and compute the loss accordingly in the target domain.

On the other hand, we bring the features of high-confidence target-domain samples closer to the source-domain prototypes. Using high-confidence samples is crucial due to their high reliability, preventing negative transfer. The prototype loss is given as follows:

$$\begin{aligned}\text{Loss}_{\text{proto}} = &\sum_{i=0}^{c}\sum_{j=1}^{m}\left\|f\left(x_s^i\right) - Pt_i^j\right\|_2 \\ &+ \sum_{i=0}^{c}\sum_{j=1}^{m}\left\|f\left(x_t^i\right) - Ps_i^j\right\|_2 \cdot 1\left(\delta_i\left(h_t\left(x_t\right)\right) \geq \sigma\right)\end{aligned} \tag{12}$$

where $1(\cdot)$ is an indicator function, only samples with high-confidence levels are eligible for computation.

Contrastive learning, similar to prototype learning, can enhance the generalization capability of models, especially when combined with data augmentation techniques [54]. An important and diverse aspect of contrastive learning is the construction of positive and negative samples. Due to the presence of pseudolabels with a certain degree of reliability for high-confidence samples in the target domain, the majority of methods incorporate these high-confidence sample features as anchors, introducing contrastive learning, while neglecting the utilization of features from low-confidence samples. For instance, in our experiments, we can perturb sample features based on the pseudolabels of high-confidence samples in the target domain, utilizing prototypes that are unrelated to target-domain classification as well as specific classification prototypes from the source domain, thereby constructing positive and negative samples. However, employing this approach causes the learned feature representations to be biased toward samples in the target domain that are similar to the source domain, and in certain transfer tasks, it may introduce negative transfer compared with contrastive learning methods that do not involve high-confidence samples, leading to a less optimistic overall performance.

Inspired by MixLRCo, we also make use of low-confidence samples. However, unlike MixLRCo and similar to the contrastive learning applied to high-confidence samples from the target domain, the positive and negative samples for low-confidence samples are constructed using source-domain prototypes. As shown in Fig. 5, for the feature of a low-confidence sample from the target domain during the training process, we disturb it by using category-general source-domain prototypes of a certain class as an anchor, and we disturb it by using category-general source-domain prototypes of other classes as positive samples. We employ feature weighting as a form of perturbation, assigning higher weights to target-domain features to ensure their dominant influence. We set all pseudolabels with a confidence lower than the threshold to -1. To store the features of low-confidence samples, we introduce a memory bank while limiting its size to avoid excessive computational overhead.

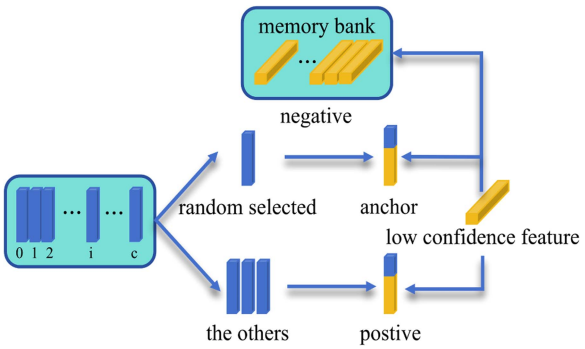


Fig. 5. Construction of anchor, positive samples, and negative samples required by contrastive learning in GSPCL. A memory bank is introduced to store a fixed number of features with low similarity to a low-confidence sample.

In addition, considering that the excessive construction of negative samples, as proposed in PCL, may include semantically similar samples, when constructing negative samples for low-confidence anchors, we first compute the distance between the anchor and low-confidence samples, and then store the features of the top memory size samples in the memory bank

$$\begin{aligned} z &\sim U\left(0.9,1\right) \\ \mathbf{r}_m^i &= zf\left(x_t^i\right)+\left(1-z\right)Pa_i \\ \mathbf{r}_+ &= zf\left(x_t^i\right)+\left(1-z\right)Pa_j \\ \mathbf{r}_- &= \text{top } M\ d\left(f\left(x_t^i\right),\mathbf{r}\right) \end{aligned} \tag{13}$$

where $\mathbf{r}_m^i$, $\mathbf{r}_+$, $\mathbf{r}_-$, and $\mathbf{r}$ denote the anchor, positive sample, negative sample, and all target samples, respectively. The variable *j* represents the classes without *i*, *M* signifies the size of the memory bank, and we use Euclidean distance as $d(\cdot)$.

In summarizing Section II-C, we construct positive and negative samples for low-confidence anchor samples. We formulate the contrastive loss as follows based on similarity measured using inner product metrics:

$$\text{Loss}_{\text{cont}} = -\log\frac{\sum_{r_+\in N}h\left(\mathbf{r}_m^i,\mathbf{r}_+\right)}{\sum_{r_+\in N}h\left(\mathbf{r}_m^i,\mathbf{r}_+\right)+\sum_{r_-\in M}h\left(\mathbf{r}_m^i,\mathbf{r}_-\right)} \tag{14}$$

where N is the number of categories minus one.

E. Loss Functions

Within the overarching framework of unsupervised domain adaptation, given that both the samples from the source domain and their corresponding labels are available, even though GSPCL directly utilizes a pretrained source-domain model, we still employ the cross-entropy loss in (1) for the source domain to prevent a decrease in classification performance during the domain adaptation process.

On the other hand, solely applying a classification loss to the source-domain data would lead to the model making overly confident predictions in the source domain, thereby reducing its generalization capability in the target domain. To encourage confident outputs and output diversity in the target domain, we introduce the Shannon entropy

$$\begin{aligned} L_e = &-\mathbb{E}_{\mathcal{B}_T}\left[\sum_c h_w^i\left[c\right]\log\left(h_w^i\left[c\right]\right)\right] \\ &+\sum_{c=1}^{C}\text{KL}(\mathbb{E}_{\mathcal{B}_T}\left[h_w^i\left[c\right]\right]||\frac{1}{C}) \end{aligned} \tag{15}$$

where h_w and h_s, respectively, denote the classification probabilities of $T_w(x_t^i)$ and $T_s(x_t^i)$, and $\mathcal{B}_T$ is a batch of data.

Moreover, as mentioned in our preliminary discussions, we employ a consistency loss to further make the model robust to image perturbations and enhance its generalization capability. Since low-confidence samples may be assigned incorrect pseudolabels, applying consistency learning to them might result in a negative transfer. Therefore, this loss is exclusively applied to high-confidence samples. In simple terms, we use the pseudolabels from the weakly augmented view as labels for the strongly augmented view, thereby constructing a cross-entropy loss

$$L_{\text{cons}} = \sum_{x_i\sim X_t}\left[1\left(\max\left(h_w\right)\geq\sigma\right)H\left(\tilde{h}_w,h_s\right)\right] \tag{16}$$

where $\tilde{h}_w = \text{argmax}(h_w)$ is the one-hot vector of prediction, and $1(\cdot)$ is an indicator function signifying that this term of loss is also exclusively applicable to high-confidence samples. $H(\cdot)$ is the cross-entropy loss function.

Combining (1) and (11)–(15), the loss function in GSPCL can be summarized as follows:

$$L_{\text{total}} = L_s\ +\lambda_1 L_{\text{proto}}+\lambda_2 L_{\text{cont}}+L_e+L_{\text{cons}} \tag{17}$$

where λ_1 and λ_2 are the weights of the two losses.

IV. Experiments

In this section, we thoroughly evaluate the proposed methodology based on three remote sensing domain adaptation dataset settings. Simultaneously, we compare our approach with competitive domain adaptation methods. The datasets utilized in the experiments encompass the AID, the UCM land use dataset, the Wuhan University (WHU) RS dataset, the RSD46-WHU dataset, the RSSCN7 dataset, the NWPU-RESISC45 dataset, and the PatternNet dataset.

A. Cross-Domain Optical Remote Sensing Datasets

Due to the absence of dedicated domain adaptation datasets tailored for remote sensing classification tasks, to evaluate the effectiveness of domain adaptation methods in the context of optical remote sensing datasets, similar to [13], [14], and [55], we extract common class images from multiple optical remote sensing image classification datasets to serve as the source domain in the transfer tasks. The categories of samples involved in each dataset in the aforementioned experimental settings are illustrated in Fig. 6. Prior to training, the images are resampled to a uniform size of 224 × 224. The differences between these datasets are briefly described as follows.

The AID dataset encompasses scene categories with pixel resolutions ranging from approximately 8 to about 0.2 m. Each

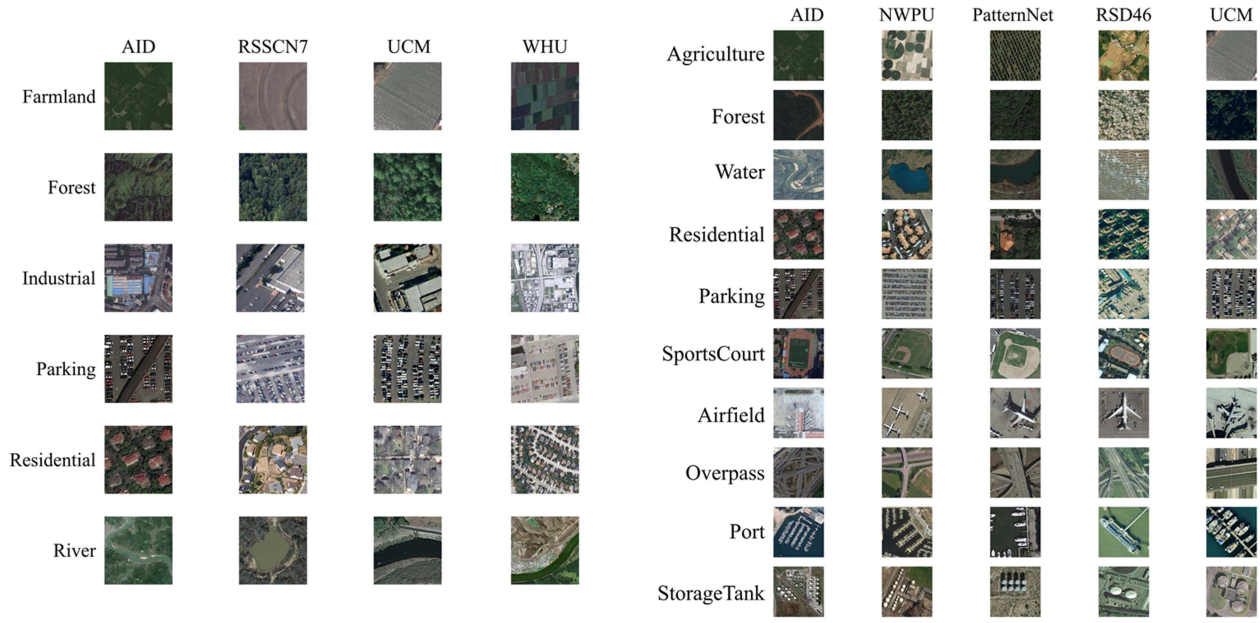


Fig. 6. Exhibition of dataset samples. (a) Contains image examples in dataset setting 1 and (b) shows image examples in dataset settings 2 and 3.

category comprises 220–420 images sized at 600 × 600 pixels. The UCM dataset consists of 100 images per category, each measuring 256 × 256 pixels at a resolution of one foot. While in the WHU dataset, there are around 50 images per category, with a resolution of 0.5 m and a size of 600 × 600 pixels [56]. The RSSCN7 dataset comprises 400 images per category collected at four different scales, with each scale containing 100 images. And the PatternNet dataset includes 800 images per category, each measuring 256 × 256 pixels [57]. Both the NWPU [58] and RSD46 [59] datasets are extensive datasets, the former consists of 700 images per category ranging in resolution from 20 cm/px to over 30 m/px, the latter comprises image data varying from 500 to 3000 per category, with resolutions ranging from 0.5 to 2 m. Each image in both datasets is sized at 256 × 256 pixels.

In order to enhance the evaluation of GSPCL and facilitate comparisons with the existing methods, we established three distinct dataset settings for the aforementioned datasets. The first two settings adhere to the principles outlined in [13] and [55], and the specific details are provided as follows.

1) Dataset Setting 1: We select six common or similar categories of images from the seven datasets mentioned above, including residential, farmland, forest, industrial, parking lot, and river. It is worth noting that, in the UCM dataset, dense residential and medium residential areas are merged into the residential category, while the other datasets classify dense residential areas as residential. The details are shown in Table I.

2) Dataset Setting 2: Considering the limited sample size in each domain and the insufficient variation among domains in Setting 1, to further validate the effectiveness of the proposed method, we follow the settings outlined in [55] to construct a dataset with higher transfer difficulty. Specifically, we select ten common or similar classes from UCM, AID, PatternNet, NWPU, and RSD46. These classes encompass agriculture, forest, water, residential, parking lot, sports field, airport, flyover, port, and storage tank, serving as the domains for the transfer task. Fine-grained or similar categories from the original datasets are merged; for instance, dense residential, medium residential, and sparse residential are consolidated into the residential category, and aircraft and airports are merged into the airport category. Table II presents the specific details.

TABLE I
NUMBER OF SAMPLES IN DATASET SETTING 1

Class	AID	RSSCN7	UCM	WHU
Farmland	370	400	100	50
Forest	250	400	100	53
Industrial	390	400	100	53
Parking	390	400	100	50
Residential	410	400	200	54
River	410	400	100	56
Total	2220	2400	700	316

3) Dataset Setting 3: Although the datasets in Setting 2 inherently exhibit class imbalance, to verify the robustness of GSPCL in domain adaptive tasks on long-tailed datasets, we follow the reversely unbalanced source and unbalanced target protocol to subsample the datasets in Setting 2. Our objective is to create unbalanced label distributions in both the source and target domains, with the source-domain's label distribution being the reverse of the target-domain's label distribution. Since each domain in Setting 2 comprises ten classes with inherent class imbalances, we first calculate the minimum number

TABLE II
NUMBER OF SAMPLES IN DATASET SETTING 2

Class	AID	NWPU	PatternNet	RSD46	UCM
Agriculture	370	1400	800	5394	100
Forest	250	700	800	5081	100
Water	830	1400	800	3679	100
Residential	1000	2100	800	2534	300
Parking	390	700	800	2598	100
SportsCourt	590	1400	1600	2728	100
Airfield	360	1400	800	3980	100
Overpass	420	700	800	2128	100
Port	380	700	800	2675	100
StorageTank	360	700	800	1368	100
total	4950	11200	8800	32165	1200

TABLE III
NUMBER OF SAMPLES IN DATASET SETTING 3

Class	A+	N+	P+	R+	U+
Agriculture	25	70	80	136	10
Forest	50	140	160	273	20
Water	75	210	240	410	30
Residential	100	280	320	547	40
Parking	125	350	400	684	50
SportsCourt	150	420	480	820	60
Airfield	175	490	560	957	70
Overpass	200	560	640	1094	80
Port	225	630	720	1231	90
StorageTank	250	700	800	1368	100
total	1375	3850	4400	7520	550

(CMN) of samples in all classes in the domain. Then, with an interval of 0.1, subsampling of samples with 0.1–1 times and 1–0.1 times the number of CMN is performed for 0–9 classes of source domain and target domain, respectively. Table III displays the scenario where the number of categories increases one by one. The situation where the number decreases one by one is the reverse, with the quantities flipped.

B. Experimental Settings

For dataset Setting 1, we designate the UCM dataset, WHU-RS dataset, AID dataset, and RSSCN7 dataset as domains U, W, A, and R, respectively. This results in 12 cross-scene classification tasks: U→W, U→A, U→R, W→U, W→A, W→R, A→U, A→W, A→R, R→U, R→W, and R→A. In this experimental setup, we select ResNet, DDC [60], DAN, joint adaptation network (JAN), RevGrad [61], and multirepresentation adaptation network (MRAN) [35] alongside the multiscale residual adaptation network (AMRAN) [13] as methods for comparison.

For dataset Setting 2, we designate the AID dataset, NWPU dataset, PatternNet dataset, RSD46 dataset, and UCM dataset as domains A, N, P, R, and U, respectively. Pairwise-domain transfer experiments are conducted between these domains, resulting in a total of 20 tasks. In addition, in order to better compare with the TSAN that proposed the dataset settings, individual domains are considered as source domains, while the other four domains are combined to form the target domain, without knowledge of the specific origin of the samples. This design introduced five additional tasks: A→{N, P, R, U}, N→{A, P, R, U}, P→{A, N, R, U}, R→{A, N, P, U}, and U→{A, N, P, R}. In this experimental setup, apart from the majority of methods proposed in Setting 1, we also select CDAN, TSAN, GDCAN, and PCT for comparison.

For dataset Setting 3, we designate domains with increasing class numbers from 0 to 9 as "+" domains and those with decreasing class numbers as "−" domains. Consequently, there are ten domains labeled as A+, A−, N+, N−, P+, P−, R+, R−, U+, and U−. It is important to note that not all ten domains are involved in pairwise transfer tasks. Specifically, transfers occur only from domains with "+" to "−" labels, and intradomain transfers between "+" and "−" within the same domain are not considered. For instance, transfers include A+→N−, but not N−→A+ or A+→A−. Hence, there are a total of 20 transfer tasks. In this experimental setup, we employ ResNet50 direct transfer, CDAN [27] and PCT [41], as comparative methods to validate the effectiveness of the proposed approach on long-tailed datasets.

Our proposed method and the compared methods are all based on the ResNet50 backbone and trained using the PyTorch-1.9 framework and RTX 3090 GPU. As discussed in Section III, we extract class-general prototypes and class-specific prototypes based on the first stage and the dimension reduction layer of ResNet50, ensuring that the dimensions of the prototypes and features are both 256. Similar to most domain adaptation experimental setups, we set the learning rates of the newly introduced layers, apart from the pretrained model, to be ten times higher than the other layers. We employ SGD with a momentum of 0.9 and weight decay of 1e-3, and the learning rate adjustment followed the formula, $\mu_p = \mu_0 / ((1 + \alpha p)^{\beta})$, where $p =$ epoch/total_epoch, $\mu_0 = 0.001$, $\alpha = 10$, and $\beta = 10$. The initial learning rate for all experiments is set to 1e-3, and the batch size is set to 16. In dataset Setting 1, the number of epochs is set to 20. For most tasks in Setting 3, the number of epochs is set to 10. Due to the large number of samples in the combined multitarget domain, often achieving satisfactory results within a single epoch, the single-source domain combined with multiple target domains in Setting 2 is set to 1 epoch. For the remaining transfer tasks in Setting 2, the number of epochs is adjusted based on the size of the target-domain samples, set to 8, 8, 8, 1, and 10 for tasks transferring to AID, NWPU, PatternNet, RSD46, and UCM, respectively.

TABLE IV
ACCURACY (%) ON CROSS-SCENE DATASET SETTING 1 FOR UNSUPERVISED DOMAIN ADAPTATION (RESNET-50)

Method	A→R	A→U	A→W	R→A	R→U	R→W	U→A	U→R	U→W	W→A	W→R	W→U	Avg
ResNet	76.88	74.86	98.42	85.18	80.71	86.71	66.44	56.50	71.20	93.11	71.25	75.14	78.03
DDC	75.63	78.29	99.21	84.68	79.71	89.24	66.26	57.71	71.84	90.09	70.13	78.00	78.40
DAN	76.88	80.29	99.37	85.32	82.86	88.92	67.07	57.50	74.68	92.79	72.79	80.14	79.88
JAN	76.83	80.71	99.05	85.50	83.86	90.19	67.48	55.83	72.47	92.30	72.75	80.86	79.82
RevGrad	78.13	82.86	98.10	84.28	81.57	88.61	69.19	61.89	76.58	92.29	73.58	81.86	80.75
MRAN	78.96	84.71	99.73	93.87	83.06	93.35	82.99	64.78	78.52	94.06	79.16	81.71	84.47
AMRAN	81.00	88.16	**99.68**	94.48	87.56	94.59	84.80	69.02	89.56	94.82	79.50	85.20	87.36
GSPCL	**81.79**	**92.14**	99.37	**98.38**	**91.86**	**98.42**	**98.51**	**80.38**	**97.78**	**96.49**	**82.21**	**89.29**	**92.36**

The bold entities represent the best classification accuracy of all methods.

Furthermore, the size of the memory bank used in contrastive learning is set to 1024 for all transfer tasks except for the single-source domain combined with multiple target domains, where it is set to 256. In the weighting process, features of low-confidence samples in the target domain are assigned weights ranging from 0.9 to 1, preventing the loss of specific classification information from the target-domain features during perturbation. In our proposed WBP strategy, α is typically set to 1.1. The confidence threshold setting for all experiments is set to 0.9. In the overall loss, the weights (λ_1 and λ_2) assigned to the contrastive loss and prototype loss were consistently set to 0.5 across all experiments. We update the source domain, pseudolabels, and stored sample features in the memory bank at each epoch.

C. Comparison and Analysis

The tasks involved in dataset Setting 1 are generally considered the most easily transferable among the three settings. The comparative results of GSPCL with other contrastive algorithms are presented in Table IV. Upon examining the average accuracy across all classification tasks within this dataset setting, our approach demonstrates significant improvements over other algorithms. Specifically, compared with ResNet, DDC, DAN, JAN, RevGrad, MRAN, and AMRAN, GSPCL achieves higher accuracy rates by 14.33%, 13.96%, 12.48%, 12.54%, 11.61%, 7.89%, and 5.00%, respectively. In this experimental setup, reference is made to AMRAN, establishing it as the primary comparative method. Our approach exhibits only minor discrepancies from AMRAN in the A→W transfer task, where the majority of methods achieve close to 100% accuracy. However, in other transfer tasks, GSPCL outperforms AMRAN significantly. Particularly in the U→A and U→R transfer tasks, our approach surpasses AMRAN by more than ten percentage points. This indicates that our method performs remarkably well in the majority of relatively straightforward domain adaptation tasks involving optical remote sensing images.

Due to the larger number of categories and images in the classification scenarios of dataset Setting 2 compared with Setting 1, the intraclass diversity and interclass similarity are substantially greater, rendering the transfer tasks considerably more challenging. The experimental results are presented in Table V. In the task of single-source-domain adaptation to a single-target domain, GSPCL exhibits an average accuracy that surpasses DAN, DANN [62], JAN, CDAN, GDCAN, and PCT by 15%, 7.4%, 9.2%, 4%, 11.8%, and 5.2%, respectively. Among the 20 transfer tasks, GSPCL achieves the best results in 13 tasks. Particularly noteworthy is the remarkable improvement in accuracy from 81.2% to 99.9% in the R→P task. It is worth mentioning that GDCAN and PCT, both employed as comparative methods, are relatively new techniques. While both our method and PCT utilize prototype learning, PCT achieves an average accuracy superior to GDCAN by 6.6% across all tasks, providing substantial evidence for the effectiveness of prototype learning in domain adaptation tasks involving optical remote sensing images. For the five tasks involving single-source-domain adaptation to a combination of multiple target domains, our method outperforms all other comparative methods, except for the R→{A, N, P, U} task. Particularly noteworthy is the exceptional performance of GSPCL in the U→{A, N, P, R} task, surpassing the most effective method, CDAN, by 12%. When considering the average accuracy across these five tasks, our approach exhibits superior performance compared with DAN, DANN, JAN, CDAN, and TSAN by 13.3%, 8.9%, 6.6%, 4.2%, and 6.5%, respectively. This observation underscores GSPCLs enhanced classification capabilities and robustness, even in the face of more challenging domain adaptation tasks. Upon comprehensive analysis of dataset Settings 1 and 2, it is evident that the integration of various modules and operations in our method has facilitated a closer alignment of feature mappings between the source and target domains within the feature extractor. In addition, these components have significantly enhanced the model's generalization abilities.

Dataset Setting 3 is specifically designed to assess the performance of methods when facing domain adaptation tasks in long-tail datasets, which are more challenging due to the contrasting distributions of labels in the source and target domains. As shown in Table VI, GSPCL achieves an average accuracy that surpasses ResNet50 directly transferred to the target domain, CDAN, and PCT by 15.2%, 12.3%, and 0.3%, respectively, across the 20 tasks. Moreover, GSPCL outperforms other methods in 12 tasks, demonstrating its capability to handle domain adaptation tasks in long-tail datasets. Here, we introduce CDAN as a comparison

TABLE V
ACCURACY (%) ON CROSS-SCENE DATASET SETTING 2 FOR UNSUPERVISED DOMAIN ADAPTATION (RESNET-50)

Setting	Source	A				N				P				R				U				Avg
	Target	N	P	R	U	A	P	R	U	A	N	R	U	A	N	P	U	A	N	P	R	(%)
S-T	DAN	84.2	82.6	68.5	77.6	89.0	92.2	65.9	82.9	66.5	69.3	52.9	83.4	69.2	64.9	67.7	63.8	68.3	62.5	94.0	53.4	72.9
	DANN	89.4	90.1	77.9	92.1	92.7	91.0	69.2	88.4	77.9	75.6	55.7	94.5	80.9	78.9	80.5	72.9	81.0	73.7	**93.9**	53.9	80.5
	JAN	88.1	91.0	72.2	77.8	89.4	92.0	66.8	86.0	77.0	**79.8**	59.3	90.9	73.3	76.0	75.9	70.3	75.6	78.5	92.5	62.3	78.7
	CDAN	95.0	97.4	81.2	92.3	97.1	93.3	84.1	96.2	77.9	78.5	58.3	**98.1**	83.4	79.7	81.2	73.3	85.5	75.1	93.1	57.0	83.9
	GDCAN	88.8	84.8	74.3	84.3	91.7	84.5	68.4	90.6	71.0	66.4	49.9	82.1	75.3	73.3	76.4	69.8	77.5	68.2	87.2	58.4	76.1
	PCT	95.5	89.6	**89.2**	97.8	95.6	81.8	79.3	98.4	76.6	70.9	**61.5**	95.0	**91.2**	70.4	59.6	77.5	90.4	71.9	90.8	**69.9**	82.7
	GSPCL	**96.3**	**99.9**	88.1	**97.9**	**97.2**	**99.9**	**91.1**	**98.5**	**82.4**	78.0	55.5	92.2	90.0	**82.2**	**99.9**	**82.3**	**94.1**	**80.9**	91.8	60.3	**87.9**
S-C	DAN	76.3				74.0				55.1				68.0				60.6				66.8
	DANN	79.7				76.5				61.7				74.8				63.3				71.2
	JAN	78.5				73.0				75.3				69.4				71.6				73.5
	CDAN	84.1				79.1				67.9				**76.6**				71.7				75.9
	TSAN	84.3				81.9				60.5				73.4				68.1				73.6
	GSPCL	**84.8**				**85.7**				**76.9**				69.2				**83.8**				**80.1**

The bold entities represent the best classification accuracy of all methods.

TABLE VI
ACCURACY (%) ON CROSS-SCENE DATASET SETTING 3 FOR UNSUPERVISED DOMAIN ADAPTATION (RESNET-50)

Source	A+				N+				P+				R+				U+				Avg
Target	N-	P-	R-	U-	A-	P-	R-	U-	A-	N-	R-	U-	A-	N-	P-	U-	A-	N-	P-	R-	(%)
ResNet	67.8	56.2	73.7	60.2	78.8	63.6	58.7	66.0	45.8	57.1	45.9	62.9	60.4	60.0	59.6	60.9	53.7	55.2	66.7	45.6	59.9
CDAN	76.1	66.3	71.8	56.9	85.0	76.2	65.9	69.6	50.8	65.3	47.8	62.9	62.3	61.4	38.5	54.0	54.8	65.1	82.6	42.7	62.8
PCT	**82.8**	**67.9**	94.7	60.9	86.2	**80.6**	**76.7**	75.5	62.0	**76.2**	57.3	72.4	81.7	64.9	**85.7**	70.2	**72.2**	70.0	**98.8**	59.4	74.8
GSPCL	77.8	55.8	**97.2**	**72.1**	**86.3**	69.1	66.9	**80.6**	**64.8**	75.0	**64.4**	**79.3**	**83.6**	**76.1**	70.6	**78.0**	67.8	**74.8**	96.1	**66.0**	**75.1**

The bold entities represent the best classification accuracy of all methods.

method because it performs better on the first two settings than other classical methods. Surprisingly, CDAN does not yield satisfactory results and, in some tasks, introduces negative transfer. For instance, in the A+→P− task, CDANs accuracy drop from 59.6% achieved by ResNet50 to 38.5%, illustrating its inability to handle domain adaptation tasks in long-tail datasets. In contrast, PCT, due to the incorporation of prototype learning, achieves highly competitive results comparable with GSPCL, especially outperforming our approach in certain tasks. However, PCT differs from our approach both in the aspect of prototype learning itself and in the additional operations and methods apart from prototype learning. This discrepancy might be a reason for the variations in tasks where our method and PCT perform best.

To present a clearer demonstration of the alignment and discriminative capabilities of our method, we employ t-SNE [63] to visualize the features extracted by different methods in the R→P task within dataset Setting 2; the data are randomly sampled from ten classes in the PatternNet domain, comprising 5000 samples. As shown in Fig. 7(a)–(f), in comparison with other methods, our approach exhibits remarkable intraclass cohesion and clearer boundaries between classes in the target domain. Particularly noteworthy is the fact that features extracted by other methods form two clusters for the "Sports court" class, whereas GSPCL yields only one cluster. In addition, we introduce gradient-weighted class activation mapping (GradCAM) [64] to generate attention maps. These maps utilize the specific class gradients from the last convolutional layer to provide an approximate localization of important regions in the images. We employ GradCAM to highlight the successful classification of our domain adaptation method in the target domain, as illustrated in Fig. 8(a)–(c); GSPCL accurately localizes the position of the oil tank and airplane in the image.

In Table VII, parameter numbers of comparing methods and the proposed GSPCL are given. The parameter number of the proposed GSPCL is relatively less than most methods, which demonstrates that the complexity of the designed network is not too high. During different training strategies and experimental details, the real computation costs of the above methods have some differences. In the proposed method, the additional computation cost of training processing is mainly caused by the design of the WBP prototype strategy and the construction of multiple losses, but the difference in this partial computation cost is slight to other methods that also involve exceptional strategies to improve the domain adaptation performance. However, when large-scale datasets are involved in the training process, the computation cost of domain adaptation methods increases seriously. In the following research, finding

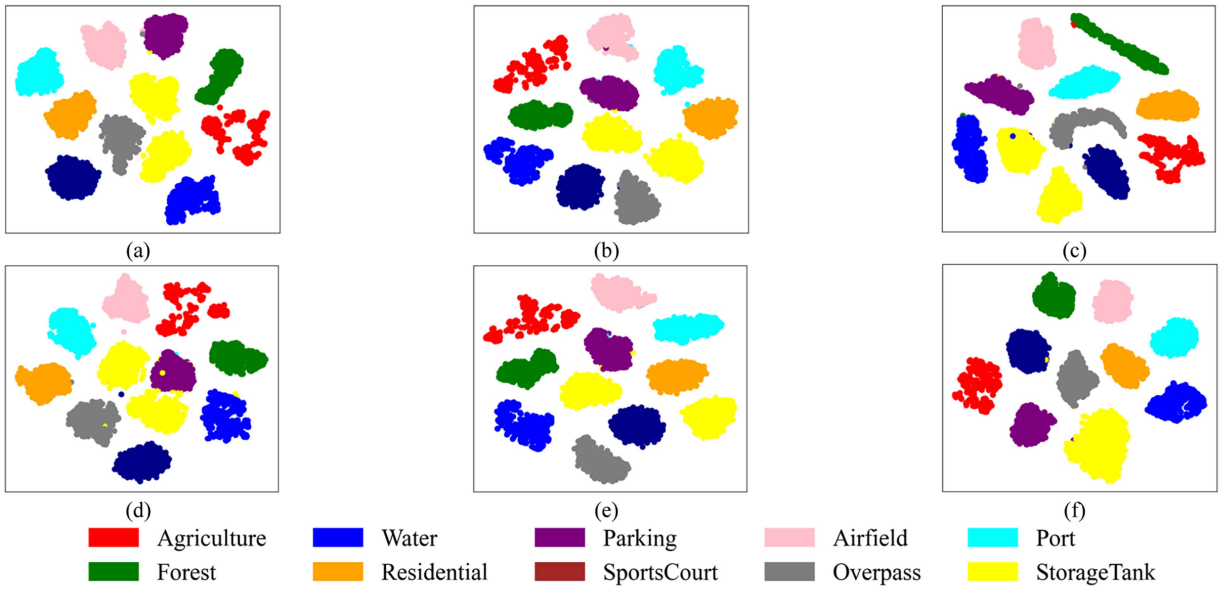


Fig. 7. t-SNE of representations on target domain for task R→P. (a) DANN. (b) JAN. (c) CDAN. (d) GDCAN. (e) PCT. (f) GSPCL.

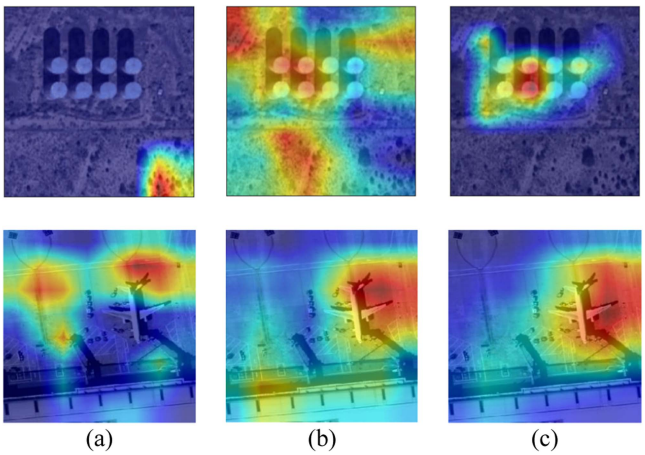


Fig. 8. Attention visualizations of the last convolutional layer learned by (a) GDCAN, (b) PCT, and (c) GSPCL.

TABLE VII
PARAMETER NUMBERS OF DIFFERENT METHODS

Methods	Parameter Number
GDCAN	104.1MB
PCT	99.5MB
CDAN	99.5MB
AMRAN	102.3MB
DAN	91.7MB
DANN	92.2MB
JAN	99.5MB
TSAN	96.7 MB
GSPCL	91.8MB

an appropriate learning strategy for domain adaptation to reduce the computation cost on large-scale datasets is worthy of note.

D. Ablation Study

As mentioned earlier, the essence of our proposed method lies in the integration of prototype learning and contrastive learning, as well as the utilization of pseudolabel generation based on BWPs. To investigate the effectiveness of each component in GSPCL, we conduct ablation studies using dataset Setting 2. In addition, to validate the potential negative impact of applying contrastive learning to high-confidence samples in the target domain, we specifically design a contrastive learning loss for high-confidence sample features. In this approach, high-confidence sample features are used as anchors. Positive samples are generated by perturbing anchors with source-domain category-general features from a randomly chosen class. Negative samples are constructed by perturbing high-confidence sample features with source-domain-specific class prototypes corresponding to the pseudolabels of other classes. To achieve optimal performance for this part of the loss, we conduct experiments and ultimately choose a multilinear mapping method inspired by CDAN as the interference technique. The formulation for constructing positive and negative samples is given as follows:

$$\begin{aligned} \mathbf{r}'_{+} &= T_{\odot}\ (r, Ps_i) = (\mathbf{R_f} r) \odot (\mathbf{R_g} Ps_i) \\ \mathbf{r}'_{-} &= T_{\odot}\ (\mathbf{r}, Ps_j) = (\mathbf{R_f} r) \odot (\mathbf{R_g} Ps_j) \end{aligned} \tag{18}$$

where Ps_i and Ps_j represent the source-domain-specific prototypes corresponding to the same class and other classes as anchor point $\mathbf{r}$, and $\mathbf{R_f}$ and $\mathbf{R_g}$ are the random matrices sampled only once and fixed in training.

TABLE VIII
ABLATION EXPERIMENTS ON DATASET SETTING 2

	A				N				P				R				U				Avg (%)
	N	P	R	U	A	P	R	U	A	N	R	U	A	N	P	U	A	N	P	R	
base	95.3	**99.9**	**92.6**	97.5	91.7	91.7	87.8	97.0	78.5	74.6	**64.2**	92.0	81.8	69.3	83.3	74.1	86.7	80.6	91.8	60.0	84.5
w p1	95.4	**99.9**	87.4	97.7	96.6	**99.9**	90.1	98.2	79.9	77.2	56.3	91.8	89.8	**82.8**	**99.9**	75.1	94.0	80.0	91.1	61.0	87.2
w p1+p2	95.3	**99.9**	87.5	97.7	97.6	**99.9**	90.3	98.1	80.2	77.1	56.4	93.3	**90.0**	82.1	**99.9**	76.3	94.2	80.0	**93.3**	60.2	87.5
w p+c1	95.2	**99.9**	91.0	97.6	97.0	91.0	85.8	97.3	78.4	74.4	62.5	92.2	81.7	60.1	81.5	74.2	87.5	80.7	91.2	60.2	84.0
w p+ c2	**96.3**	**99.9**	88.1	**97.9**	97.2	**99.9**	**91.1**	**98.5**	**82.4**	**78.0**	55.5	92.2	**90.0**	82.2	**99.9**	**82.3**	94.1	80.9	91.8	60.3	**87.9**
w p+c1+c2	96.2	**99.9**	87.5	**97.9**	**98.6**	93.5	82.6	97.2	81.7	75.7	61.5	**94.3**	82.8	81.5	83.8	73.6	**94.3**	**81.9**	92.0	**68.6**	86.2
w BMD	95.5	**99.9**	88.4	**97.9**	97.6	**99.9**	**91.1**	97.8	80.8	75.4	55.3	91.0	87.3	81.4	99.7	75.3	93.4	80.2	92.2	60.6	87.0

The bold entities represent the best classification accuracy of all methods.

The contrastive loss constructed from high-confidence sample features is given as follows:

$$\text{Loss}'_{\text{cont}} = -\log \frac{\sum_{r_+ \in C_-} h\left(\mathbf{r}, \mathbf{r}'_+\right)}{\sum_{r_+ \in C_-} h\left(\mathbf{r}, \mathbf{r}'_+\right) + \sum_{r_- \in M} h\left(\mathbf{r}, \mathbf{r}'_-\right)} \quad (19)$$

where $C_- = (c-1) \times k$, c is the number of classes, and m is the number of prototypes in each class.

We set the baseline method as consisting solely of classification loss, consistency loss, and entropy loss. A summary of the ablation experimental results regarding prototype loss, low-confidence contrastive loss, BWP strategy, and high-confidence contrastive loss is presented in Table VIII. Here, $c1$, $c2$, $p1$, and $p2$ represent the high-confidence contrastive loss, low-confidence contrastive loss, and two prototype losses, respectively. p represents the sum of $p1$ and $p2$. $w\,c1 + c2 + p$ indicates the addition of these three losses to the baseline method. Our proposed method is denoted as $w\,c2{+}p$, while w BMD signifies the replacement of the BWP strategy proposed by us with the BMD strategy.

1) Prototype Loss and Contrastive Loss: As shown in Table VIII, the baseline method we construct demonstrates impressive performance. This achievement can be attributed to the consistency loss enforced on two levels of transformation in the target domain, namely, $T_w(x_t^i)$ and $T_s(x_t^i)$, which enhances the robustness of the adaptive model. In addition, the entropy loss ensures that the model's classification outputs in the target domain do not exhibit excessively high-confidence scores for any particular class, thus enhancing the model's generalization abilities. Building upon this foundation, our introduced prototype loss significantly improves classification accuracy, elevating the average accuracy across the 20 adaptive tasks from 84.5% to 87.5%. This improvement indicates the success of our construction of classification prototypes in both the source and target domains. The prototype loss encourages the features of source-domain samples to converge toward the target-domain prototype features, while the features of high-confidence samples in the target domain converge toward the source-domain class prototype features. Consequently, the data from both domains exhibit similar feature mappings on the feature extractor. Indeed, optical remote sensing image datasets exhibit a certain degree of intraclass diversity and interclass similarity. However, this degree is not as substantial as the differences observed in conventional datasets, such as the significant distinction between the clipart domain and product domain in OfficeHome dataset. Optical remote sensing images share inherent similarities due to their common nature, making them suitable for problem solving through prototype learning. The incorporation of contrastive loss further improves the overall average accuracy by 0.4%. It is noteworthy that the contrastive loss specifically utilizes low-confidence samples, which are characterized by their low similarity with other samples within the same class, especially those from the target domain that resemble the source-domain samples. These samples tend to exhibit low confidence because of their dissimilarity with other samples within the same class and their resemblance to source-domain samples. The construction of positive and negative samples for low-confidence samples, along with the design of the contrastive loss, directs the model's attention toward cross-class samples in the target domain, mitigating the ambiguity associated with classifying cross-class samples.

2) Low-Confidence and High-Confidence Contrastive Loss: From Table VIII, it is evident that, even though incorporating high-confidence contrastive loss in addition to GSPCL leads to improved results in certain individual tasks, the average accuracy decreases by 1.7%. Particularly, significant drops in accuracy are observed in the N→P, N→R, and R→P tasks, namely, 6.4%, 8.5%, and 16.1%, respectively. When we exclude low-confidence contrastive loss and instead replace it with high-confidence contrastive loss, the average accuracy decreases by 0.5% compared with the baseline method and by 3.9% compared with our proposed approach. Overall, incorporating high-confidence samples for contrastive learning harms the adaptability performance. Comparing the lines $w\,c1 + p$, $w\,c2 + p$, and $w\,c1 + c2 + p$, it is evident that $w\,c2 + p$ achieves the best or comparable results in 12 out of 20 tasks. Thus, solely incorporating low-confidence contrastive loss proves to be the optimal choice. In addition, it can be observed that GSPCLs performance is not as competitive as $w\,c1 + c2 + p$ when transferring from the source domain U to other domains. We hypothesize that this outcome is attributed to the fact that domain U has the smallest sample size among the five domains and

exhibits substantial dissimilarity compared with other domains. In situations where the target-domain data are abundant, relying solely on low-confidence contrastive loss may leave a significant portion of high-confidence sample features unaddressed. At such times, incorporating high-confidence contrastive loss can enhance the model's generalization abilities.

3) BMD and BWP Strategy: It can be observed that when the loss functions remain constant and only our prototype strategy is replaced by the BWP strategy to the BMD strategy, the average accuracy decreases by 0.9%. This indicates the superiority of our strategy. Our prototype generation strategy is designed to facilitate pseudolabel assignment for the target domain and contrastive learning. It addresses potential shortcomings in the BMD strategy, generating more robust prototypes. Across the 20 tasks, our strategy achieves superior or comparable results to the BMD strategy in 16 tasks. However, due to the necessity of limiting the number of clustered samples to prevent prototype dominance, a constraint dictated by parameters, the final adaptation performance is influenced by these parameters. Consequently, our method might not achieve optimal results across all tasks.

V. Conclusion

In this work, our primary focus lies in the intrinsic characteristics of datasets and the mitigation of performance degradation issues often overlooked by most domain adaptation methods, particularly in the context of long-tailed datasets. The proposed GSPCL is a novel prototype contrastive learning framework, employed for addressing cross-scene classification challenges in optical remote sensing images. Through the proposed prototype generation strategy and prototype loss function, the feature mappings between the source and target domains within the feature extractor are brought closer, facilitating the learning of domain-invariant features. Besides, based on the conducting experiments and analysis on utilizing high- and low-confidence samples for contrastive learning, a novel contrastive loss is proposed by integrating features from low-confidence samples with prototypes, aiming to further mitigate the bias issues introduced by long-tail distributions. However, there are some imperfections, such as an excessive number of parameters should be determined before training and the construction of prototypes could be more adaptive and robust. In the future, we will explore more novel strategies for remote sensing domain adaptation problem.

References

[1] J. Shi, W. Liu, H. Shan, E. Li, X. Li, and L. Zhang, "Remote sensing scene classification based on multibranch fusion attention network," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, pp. 1–5, Mar., 2023, Art. no. 3001505.

[2] G. Xu, X. Jiang, and X. Liu, "Color-aware self-supervised learning for scene classification and segmentation of remote sensing images," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2023, pp. 5049–5052.

[3] L. Fang, Y. Kuang, Q. Liu, Y. Yang, and J. Yue, "Rethinking remote sensing pretrained model: Instance-aware visual prompting for remote sensing scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, pp. 1–13, Nov. 2023, Art. no. 5626713 .

[4] T. Liu, Y. Gu, W. Yu, X. Jia, and J. Chanussot, "Separable coupled dictionary learning for large-scene precise classification of multispectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–14, Oct. 2022, Art. no. 5542214.

[5] C. Xu, G. Zhu, and J. Shu, "A lightweight and robust lie group-convolutional neural networks joint representation for remote sensing scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–15, Jan. 2022, Art. no. 5501415.

[6] A. Gretton, A. Smola, J. Huang, M. Schmittfull, K. Borgwardt, and B. Scholkopf, "Covariate shift by kernel mean matching," in *Dataset Shift in Machine Learning*, Q. Candela, M. Sugiyama, A. Schwaighofer, and N. Lawrence, Eds. Cambridge, MA, USA: MIT Press, 2009, pp. 131–160.

[7] X. Huang, X. Wang, Q. Yu, and Y. Cheng, "Minimum adversarial distribution discrepancy for domain adaptation," *IEEE Trans. Cogn. Develop. Syst.*, vol. 14, no. 4, pp. 1440–1448, Dec. 2022.

[8] S. Li, B. Xie, Q. Lin, C. H. Liu, G. Huang, and G. Wang, "Generalized domain conditioned adaptation network," *IEEE Trans. Pattern. Anal. Mach. Intell.*, vol. 44, no. 8, pp. 4093–4109, Aug. 2022.

[9] X. Zhang, Y. Li, Q. Pan, and C. Yu, "Triple loss adversarial domain adaptation network for cross-domain sea–land clutter classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, pp. 1–18, Oct. 2023, Art. no. 5110718.

[10] Y. Huang, T. Li, S. Liu, and W. Mei, "Adversarial self-training unsupervised domain adaptation for remote sensing scene classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2022, pp. 1572–1575.

[11] A. Wu, Y. Han, L. Zhu, and Y. Yang, "Instance-invariant domain adaptive object detection via progressive disentanglement," *IEEE Trans. Pattern. Anal. Mach. Intell.*, vol. 44, no. 8, pp. 4178–4193, Aug. 2022.

[12] Z. Wang, Y. Luo, P.-F. Zhang, S. Wang, and Z. Huang, "Discovering domain disentanglement for generalized multi-source domain adaptation," in *Proc. IEEE Int. Conf. Multimedia Expo.*, 2022, pp. 1–6.

[13] S. Zhu, B. Du, L. Zhang, and X. Li, "Attention-based multiscale residual adaptation network for cross-scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–15, Mar. 2021, Art. no. 5400715.

[14] C. Yang, Y. Dong, B. Du, and L. Zhang, "Attention-based dynamic alignment and dynamic distribution adaptation for remote sensing cross-domain scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–13, Nov. 2022, Art. no. 5634713.

[15] B. Zhang, T. Chen, and B. Wang, "Curriculum-style local-to-global adaptation for cross-domain remote sensing image segmentation," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–12, Oct. 2021, Art. no. 5611412.

[16] G.-S. 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.

[17] Q. Zou, L. Ni, T. Zhang, and Q. Wang, "Deep learning based feature selection for remote sensing scene classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 11, pp. 2321–2325, Nov. 2015.

[18] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *Proc. 18th SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, 2010, pp. 270–279.

[19] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?," in *Proc. 27th Int. Conf. Neural Inf. Process. Syst.*, 2014, pp. 3320–3328.

[20] S. Lee, S. Cho, and S. Im, "DRANet: Disentangling representation and adaptation networks for unsupervised cross-domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 15247–15256.

[21] Y. Zhang, J. Li, and Z. Wang, "Low-confidence samples matter for domain adaptation," 2022, *arXiv:2202.02802*.

[22] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," *J. Mach. Learn. Res.*, vol. 13, pp. 723–773, Mar. 2012.

[23] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Learning transferable features with deep adaptation networks," in *Proc. 32nd Int. Conf. Mach. Learn.*, 2015, pp. 97–105.

[24] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Unsupervised domain adaptation with residual transfer networks," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst.*, 2016, pp. 136–144.

[25] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Deep transfer learning with joint adaptation networks," in *Proc. 34th Int. Conf. Mach. Learn.*, 2017, pp. 2208–2217.

[26] I. J. Goodfellow et al., "Generative adversarial nets," in *Proc. 27th Int. Conf. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[27] M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Conditional adversarial domain adaptation," in *Proc. 32nd Int. Conf. Neural Inf. Process. Syst.*, 2018, pp. 1647–1657.

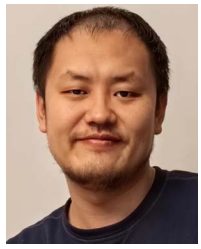
[28] M. Chen, S. Zhao, H. Liu, and D. Cai, "Adversarial-learned loss for domain adaptation," in *Proc. 34th AAAI Conf. Artif. Intell.*, 2020, pp. 3521–3528.
[29] K. Bousmalis, G. Trigeorgis, N. Silberman, D. Krishnan, and D. Erhan, "Domain separation networks," in *Proc. 30th Conf. Neural Inf. Process. Syst.*, 2016, pp. 343–351.
[30] Y. Jin, X. Wang, M. Long, and J. Wang, "Minimum class confusion for versatile domain adaptation," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 464–480.
[31] X. Wang, J. Zhuo, M. Zhang, S. Wang, and Y. Fang, "Revisiting unsupervised domain adaptation models: A smoothness perspective," in *Proc. Asian Conf. Comput. Vis.*, 2022, pp. 1504–1521.
[32] K. Li, C. Liu, H. Zhao, Y. Zhang, and Y. Fu, "ECACL: A holistic framework for semi-supervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 8558–8567.
[33] D. Huang, J. Li, W. Chen, J. Huang, Z. Chai, and G. Li, "Divide and adapt: Active domain adaptation via customized learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 7651–7660.
[34] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 3–19.
[35] Y. Zhu et al., "Multi-representation adaptation network for cross-domain image classification," *Neural Netw.*, vol. 119, pp. 214–221, Nov. 2019.
[36] Z. Liu, Z. Miao, X. Zhan, J. Wang, B. Gong, and S. X. Yu, "Large-scale long-tailed recognition in an open world," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2537–2546.
[37] S. H. Tan, X. C. Peng, and K. Saenko, "Class-imbalanced domain adaptation: An empirical odyssey," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 585–602.
[38] C. Chen et al., "Progressive feature alignment for unsupervised domain adaptation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 627–636.
[39] Y. Pan, T. Yao, Y. Li, Y. Wang, C.-W. Ngo, and T. Mei, "Transferrable prototypical networks for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2234–2242.
[40] G. Kang, L. Jiang, Y. Yang, and A. G. Hauptmann, "Contrastive adaptation network for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4888–4897.
[41] K. Tanwisuth et al., "A prototype-oriented framework for unsupervised domain adaptation," in *Proc. 35th Conf. Neural Inf. Process. Syst.*, 2021, vol. 34, pp. 17194–17208.
[42] J. Liang, D. Hu, and J. Feng, "Do we really need to access the source data? Source hypothesis transfer for unsupervised domain adaptation," in *Proc. 37th Int. Conf. Mach. Learn.*, 2020, pp. 6028–6039.
[43] J. Liang, D. Hu, Y. Wang, R. He, and J. Feng, "Source data-absent unsupervised domain adaptation through hypothesis transfer and labeling transfer," *IEEE Trans. Pattern. Anal. Mach. Intell.*, vol. 44, no. 11, pp. 8602–8617, Nov. 2022.
[44] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep clustering for unsupervised learning of visual features," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 139–156.
[45] S. Qu, G. Chen, J. Zhang, Z. Li, W. He, and D. Tao, "BMD: A general class-balanced multicentric dynamic prototype strategy for source-free domain adaptation," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 165–182.
[46] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3733–3742.
[47] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 9726–9735.
[48] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2020, pp. 9912–9924.
[49] M. Tschannen, J. Djolonga, P. K. Rubenstein, S. Gelly, and M. Lucic, "On mutual information maximization for representation learning," in *Proc. Int. Conf. Learn. Representation*, 2020, pp. 1–16.
[50] J. Li, P. Zhou, C. Xiong, R. Socher, and S. C. H. Hoi, "Prototypical contrastive learning of unsupervised representations," 2020, *arXiv:2005.04966*.
[51] A. V. D. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018, *arXiv:1807.03748*.
[52] J. Li, Y. Zhang, Z. Wang, and K. Tu, "Probabilistic contrastive learning for domain adaptation," 2021, *arXiv:2111.06021*.
[53] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[54] Z. Dong, T. Liu, and Y. Gu, "Spatial and semantic consistency contrastive learning for self-supervised semantic segmentation of remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, pp. 1–12, Sep. 2023, Art. no. 5621112.
[55] J. Zheng et al., "A two-stage adaptation network (TSAN) for remote sensing scene classification in single-source-mixed-multiple-target domain adaptation (S^2M^2T DA) scenarios," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–13, Aug. 2021, Art. no. 5609213.
[56] G.-S. Xia, W. Yang, J. Delon, Y. Gousseau, H. Sun, and H. Maître, "Structural high-resolution satellite image indexing," in *Proc. ISPRS TC VII Symp.*, 2010, vol. 38, pp. 298–303.
[57] X. Zhou and S. Prasad, "Deep feature alignment neural networks for domain adaptation of hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 10, pp. 5863–5872, Oct. 2018.
[58] G. Cheng, J. Han, and X. Lu, "Remote sensing image scene classification: Benchmark and state of the art," *Proc. IEEE*, vol. 105, no. 10, pp. 1865–1883, Oct. 2017.
[59] Z. Xiao, Y. Long, D. Li, C. Wei, G. Tang, and J. Liu, "High-resolution remote sensing image retrieval based on CNNs from a dimensional perspective," *Remote Sens.*, vol. 9, no. 7, Jul. 2017, Art. no. 725.
[60] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell, "Deep domain confusion: Maximizing for domain invariance," 2014, *arXiv:1412.3474*.
[61] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *Proc. 32nd Int. Conf. Mach. Learn.*, 2015, vol. 37, pp. 1180–1189.
[62] Y. Ganin et al., "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 2030–2096, May 2015.
[63] L. V. D. Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, 2008.
[64] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 618–626.

Puhua Chen (Senior Member, IEEE) received the B.S. degree in environmental engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2009, and the Ph.D. degree in circuit and system from Xidian University, Xi'an, China, in 2016.

She is currently an Associate Professor with the School of Artificial Intelligence, Xidian University. Her current research interests include machine learning, pattern recognition, and remote sensing image interpretation.

Yu Qiu received the B.S. degree in computer science from Huazhong Agricultural University, Wuhan, China, in 2022. He is currently working toward the M.S. degree in artificial intelligence with the School of Artificial Intelligence, Xidian University, Xi'an, China.

His research interests include computer vision and deep learning, especially remote sensing image analysis and domain adaptation.



Lei Guo received the B.S. degree in intelligent science and technology from Xidian University, Xi'an, China, in 2009. He is currently working toward the Ph.D. degree in electronic information with the School of Artificial Intelligence, Xidian University, Xi'an, China.

His research interests include pattern recognition, deep learning, machine vision, and intelligent signal processing.

Xiangrong Zhang (Senior Member, IEEE) received the B.S. and M.S. degrees in computer application technology from the School of Computer Science, Xidian University, Xi'an, China, in 1999 and 2003, respectively, and the Ph.D. degree in pattern recognition and intelligent system from the School of Electronic Engineering, Xidian University, Xi'an, China, in 2006.

From 2015 to 2016, she was a Visiting Scientist with Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA, USA. She is currently a Professor with the Key Laboratory of Intelligent Perception and Image Understanding, Ministry of Education, Xidian University. Her research interests include pattern recognition, machine learning, and remote sensing image analysis and understanding.

Fang Liu (Senior Member, IEEE) received the B.S. degree from Xi'an Jiaotong University, Xi'an, China, and the M.S. degree from Xidian University, Xi'an, China, in 1984 and 1995, respectively, both in computer science and technology.

She is currently a Professor with the School of Computer Science, Xidian University. Her research interests include signal and image processing, synthetic aperture radar image processing, multiscale geometry analysis, learning theory and algorithms, optimization problems, and data mining.

Prof. Liu was a recipient of the Second Prize of the National Natural Science Award in 2013.

Licheng Jiao (Fellow, IEEE) received the B.S. degree in electrical engineering and computer science from Shanghai Jiao Tong University, Shanghai, China, in 1982, and the M.S. degree in electric engineering and Ph.D. degree in signals, circuits and systems from Xi'an Jiaotong University, Xi'an, China, in 1984 and 1990, respectively.

Since 1992, he has been a Professor with the School of Electronic Engineering, Xidian University, Xi'an, China, where he is currently the Director of the Key Laboratory of Intelligent Perception and Image Understanding, Ministry of Education, Xidian University, Xi'an, China. His research interests include image processing, natural computation, machine learning, and intelligent information processing.

Dr. Jiao is a Foreign Member of the Academia Europaea and the Russian Academy of Natural Sciences. He is a Fellow of IET, CAAI, CIE, CCF, and CAA. He is also a Councilor of the Chinese Institute of Electronics, a Committee Member of the Chinese Committee of Neural Networks, an Expert of the Academic Degrees Committee of the State Council, the Chairman of the Awards and Recognition Committee, and the Vice Board Chairperson of the Chinese Association of Artificial Intelligence.

Lingling Li (Senior Member, IEEE) received the B.S. and Ph.D. degrees in circuit and system from Xidian University, Xi'an, China, in 2011 and 2017, respectively.

From 2013 to 2014, she was an exchange Ph.D. student with Intelligent Systems Group, Department of Computer Science and Artificial Intelligence, University of the Basque Country UPV/EHU, Spain. She is an Associate Professor with the Key Laboratory of Intelligent Perception and Image Understanding, Ministry of Education, School of Artificial Intelligence, Xidian University. Her research interests include image processing, deep learning, and pattern recognition.