A Hybrid Multitask Learning Network for Hyperspectral Image Classification With Few Labels

Hao Liu, Mingyang Zhang, *Member, IEEE*, Ziqi Di, Maoguo Gong, *Fellow, IEEE*, Tianqi Gao, *Student Member, IEEE*, and A. K. Qin, *Senior Member, IEEE*

***Abstract*— Recently, the field of hyperspectral image (HSI) classification has witnessed advancements with the emergence of deep learning models. Promising approaches, such as self-supervised strategies and domain adaptation, have effectively tackled the overfitting challenges posed by limited labeled samples in HSI classification. To extract comprehensive semantic information from different types of auxiliary tasks, which view the problem from multiple perspectives, and efficiently integrate multiple tasks into a single network, this article proposes a hybrid multitask learning (MTL) framework (HyMuT) by sharing representations across multiple tasks. Based on the similarity between the data and the target classification task, we construct three auxiliary tasks that are similar, related, and weakly correlated to the target task, while three corresponding MTL methods are integrated. The framework establishes a backbone network with a hard parameter sharing mechanism, which handles the main task and a similar spatial mask classification task. Subsequently, a hierarchical transfer MTL approach is introduced to transfer the knowledge of a spatial-spectral joint mask reconstruction task from the autoencoder to the backbone network. Furthermore, a new source domain HSI dataset is introduced as an auxiliary task weakly correlated. To solve the source domain classification task and assist the hard parameter sharing mechanism, a dual adversarial classifier based on adversarial learning is employed. This classifier effectively extracts domain and task invariance. Extensive experiments are conducted on four benchmark HSI datasets to evaluate the performance. The results demonstrate that HyMuT outperforms state-of-the-art methods. This code will be available from the website: https://github.com/HaoLiu-XDU/HyMuT.**

***Index Terms*— Domain adaptation, hyperspectral image (HSI) classification, multitask learning (MTL), self-supervised strategy.**

I. Introduction

HYPERSPECTRAL image classification technology utilizes the rich spectral and spatial information of hyperspectral images (HSI) to classify each pixel and finds crucial applications in agriculture, forestry, mining, and other fields [1], [2], [3].

Traditional methods for HSI classification often rely on shallow feature engineering [4], [5] or simple classifiers [6], [7]. But these approaches tend to focus solely on pixel-level features, overlooking the spatial–spectral joint relationships between adjacent pixels. In recent years, the development of deep learning methods [8], [9] is rapid and they can directly learn feature representations based on terminal tasks, thereby improving the shortcomings of feature engineering and classifiers. Zhang et al. [10] proposed a spatial–logical aggregation network (SLA-NET) with morphological transformation for tree species classification and Fang et al. [11] employed instance segmentation into HSI interpretation. The application and advancement of deep learning techniques in remote sensing field have led to the development of deep and wide networks [12], [13], [14]. These networks have the capability to capture deep spatial–spectral features based on classification task, enabling more comprehensive and accurate classification. However, the strong generalization ability of these large-scale networks often relies on a large number of high-quality labeled samples. Acquiring valuable labels for HSI datasets is time-consuming, laborious, and, in some cases, even infeasible due to inaccessible imaging areas. The growing gap between the increasing amount of HSI data and the scarcity of valuable labels necessitates techniques to address the problem of limited labels in datasets. Designing strategies to enhance the generalization ability of models based on limited data remains an ongoing challenge.

The conventional mainstream approach to tackle this challenge is to leverage semisupervised learning methods, which extract information from unlabeled samples to enhance classification performance. These methods include active learning [15], self-training models [16], and cotraining models [17], [18]. However, a single wrong pseudo label sample with high confidence can significantly degrade the overall model performance, which is particularly problematic for HSI classification with only a few labeled samples.

In recent years, two emerging approaches, domain adaptation and few-shot learning (FSL) techniques, have gained attention in addressing the problem of sample insufficiency. Domain adaptation [19], [20], [21], [22] utilizes a large amount of source domain data and prior knowledge to support

Manuscript received 31 July 2023; revised 3 October 2023 and 23 November 2023; accepted 16 January 2024. Date of publication 19 January 2024; date of current version 1 February 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62376205 and Grant 62036006, in part by the China Postdoctoral Science Foundation under Grant 2021T140528, and in part by the Key Research and Development Project of Shaanxi Province under Grant 2022QCY-LL-35. *(Corresponding author: Mingyang Zhang.)*

Hao Liu, Mingyang Zhang, Ziqi Di, Maoguo Gong, and Tianqi Gao are with the Key Laboratory of Collaborative Intelligent Systems, Ministry of Education, Xidian University, Xi'an 710071, China (e-mail: hliu_7@stu.xidian.edu.cn; omegazhangmy@gmail.com; qzdi@stu.xidian.edu.cn; gong@ieee.org; tqgao@stu.xidian.edu.cn).

A. K. Qin is with the Department of Computing Technologies, Swinburne University of Technology, Melbourne, VIC 3122, Australia (e-mail: kqin@swin.edu.au).

Digital Object Identifier 10.1109/TGRS.2024.3356008

target domain tasks. However, public HSI datasets are often targeted at specific areas, and the common categories across datasets are limited. To enhance the rapid learning capability of the HSI data through meta-learning, and make effective use of the existing labeled samples, several typical few-shot learning methods, including siamese networks [23], prototypical networks [24], [25], and relation networks [26], were continuously proposed for HSI classification. Besides, some advanced multisource classification methods [27], [28], [29], [30] were also proposed to leverage data from other source domains. These methods rely on similarity measures to achieve efficient classification with limited samples.

In order to further solve the classification problem of few samples, numerous studies have focused on combining few-shot learning and domain adaptation for HSI, which have made significant progress. Li et al. [31] introduced the deep cross-domain few-shot learning method (DCFSL), Zhang et al. [20] proposed a topological structure and semantic information transfer network, and Xi et al. [32] proposed an FSL framework with a class-covariance metric (CMFSL). Additionally, Zhang et al. [33] presented the graph information aggregation cross-domain few-shot learning (Gia-CFSL) framework, and Zhang et al. [34] proposed the cross-domain self-taught network (CDSTN). These works have achieved significant progress in HSI classification. However, the aforementioned methods paid attention to take use of advanced semantic information related to classification, ignoring the possibility to extract more information from the redundant data of HSI.

To extract features from multiple perspectives of multiple auxiliary tasks, we employ multitask learning (MTL) to build a unified framework. MTL prompts the model to generalize better in terms of the initial task through the sharing representations between various related or similar tasks, which may mutually reinforce each other. MTL excels at transferring the knowledge between tasks, resulting in an improved diversity of knowledge extracted from the network. There are two main implementations of MTL: feature-based and parameter-based approaches [35]. In the context of neural networks, deep relationship networks [36], fully-adaptive feature sharing [37], cross-stitch networks [38], weight losses with uncertainty [39], and tensor factorization [40] received significant attention. However, these methods typically emphasize algorithmic optimization processes for MTL or seek other datasets to construct tasks.

Self-supervised learning gives us inspiration for auxiliary tasks construction. Self-supervised learning is a data-driven methodology by learning data representations without human annotation. Existing self-supervised methods can be broadly categorized into three groups: generative, predictive, and contrastive methods. Generative self-supervised methods learn representations by reconstructing or generating input data. For instance, Mou et al. [41] proposed a fully conv–deconv network for spectral–spatial feature learning, while He et al. [42] reconstructed randomly masked patches using a vision transformer. Predictive methods introduce new label prediction tasks based on spatial, spectral, or other characteristics. Singh et al. [43] utilized image inpainting as a pretext task, and Vincenzi et al. [44] provided an initial attempt to leverage spectral context for self-supervised learning. Contrastive methods train models by contrasting semantically similar inputs and pushing them to be close in the representation space. Zhao et al. [45] employed a self-supervised model based on siamese networks to extract features, and Guan and Lam [46] proposed cross-domain contrastive learning for unsupervised representation learning of HSI. Inspired by self-supervised strategy, numerous auxiliary tasks were designed to enhance image representation. While there have been numerous studies on two-dimensional (2-D) remote sensing images due to their similarity to red, green, blue (RGB) images [47], [48], [49], the scenario changes when dealing with three-dimensional (3-D) HSI. In the case of HSI, many conventional and well-established auxiliary tasks fail due to the unique spectral continuity of the data.

In the field of HSI processing, several MTL methods based on the self-supervised strategy were proposed to construct various auxiliary tasks tailored to the primary task of HSI processing, such as reconstruction task [50], [51], [52], superpixel-based feature cubes [53], RGB images super-resolution [54], and graph information task [55]. However, these methods typically focus on building a single pretext task, often at the pixel-level, patch-level, or image-level, resulting in a lack of capturing multiple types of information. Additionally, some multiple auxiliary tasks often lack efficient knowledge transfer methods, and cannot efficiently integrate tasks into a single network. In general, although the aforementioned methods improved HSI classification, they still have certain limitations.

1) The self-supervised strategy-based method constructs a limited number of auxiliary tasks, resulting in a lack of diversity in effective information and tasks.
2) There is a lack of inappropriate auxiliary tasks that cater to the unique characteristics of HSI, making it difficult to utilize features extracted from auxiliary tasks.
3) Traditional MTL frameworks rely on a single MTL approach, limiting the flexibility to set different methods for various tasks.

To meet these challenges, this article proposes a novel approach that combines three MTL methods capable of handling tasks with varying degrees of similarity between the data. Suitable tasks are constructed for each method. First, the target classification and a spatial mask prediction tasks are constructed. Two task-specific layers follow behind the backbone network, enabling an MTL approach with hard parameter sharing. Additionally, a spatial–spectral joint mask reconstruction task is designed to feed a symmetric autoencoder for the target domain data. This task facilitates the transfer of knowledge adaptively from the weakly shared encoder to the backbone network through hierarchical transfer MTL. Moreover, the source domain data is treated as a separate task, and the shared backbone network is trained through adversarial-based MTL. Finally, a dual adversarial classifier is introduced after the backbone network to extract invariant features across different domains and tasks using gradient reversal layer [56]. The contributions of this article are as follows.

1) We present a hybrid MTL approach that combines three distinct methods to effectively leverage the diverse characteristics of pretext tasks in parallel.
2) We propose a novel end-to-end framework that integrates self-supervised strategy-based and domain adaptation auxiliary tasks, enabling the extraction of high-level semantic information from various auxiliary tasks, each offering different perspectives.
3) Based on the concept of self-supervised strategy, we construct two mask auxiliary tasks tailored to the characteristics of HSI data, facilitating the extraction of highly redundant information from HSI.

The remainder of this article is organized as follows. In Section II, we provide background knowledge and discuss the motivation behind HyMuT. Section III presents a detailed description of the proposed method. In Section IV, we validate the effectiveness of the proposed method on four real datasets and analyze the hyperparameters involved. Finally, in Section V, we conclude this article with final remarks.

II. Preliminaries and Motivation

A. Multitask Learning

Focusing solely on a single task may lead to overlooking valuable information that could enhance the performance of desired metrics. This valuable information can be derived from training related or similar tasks. By sharing representations across multiple tasks, a model can achieve better generalization on the original task, known as MTL. In MTL, we consider a set of m related learning tasks $\{T_i\}_{i=1}^{m}$, with the objective of jointly learning these tasks to improve the performance of each task by leveraging the knowledge contained in the other tasks. Each task T_i typically has its own dataset D_i consisting of n_i training samples $\boldsymbol{X^i} = (\boldsymbol{x_1^i}, \boldsymbol{x_2^i}, \ldots, \boldsymbol{x_{n_i}^i})$ along with their corresponding labels $\boldsymbol{Y^i} = (y_1^i, y_2^i, \ldots, y_{n_i}^i)$.

Previous research on MTL for HSI interpretation has predominantly focused on sparse representation methods [57], [58], [59]. However, more recent studies have emerged that combine MTL with auxiliary tasks for HSI processing. For instance, Liu et al. [50] simultaneously performed classification and reconstruction tasks to aid in classifying unknown classes, Hang et al. [51] designed a generator network to handle both reconstruction and classification tasks, Tu et al. [53] introduced a superpixel-based auxiliary MTL approach using auxiliary feature cubes, Li et al. [54] improved HSI super-resolution using RGB images as an auxiliary task, and Li et al. [55] proposed incorporating graph information to learn intrinsic relationships among samples. Song et al. [52] proposed perturbations, masked feature reconstruction, and spectral clip order prediction tasks to perform MTL. These methods contributed to the application of MTL in the field of HSI.

However, these methods typically rely on a single auxiliary task, such as data reconstruction or the utilization of unlabeled data. While these auxiliary tasks are undoubtedly beneficial for the main task, they have limitations when dealing with real classification scenarios with very few labeled samples. Furthermore, existing multitask knowledge transfer frameworks often adjust the losses of multiple tasks simultaneously using the backpropagation algorithm. This structure resembles a multi-objective optimization problem and introduces additional challenges in parameter settings and adjustments. To address these limitations, we propose three auxiliary tasks within a hybrid MTL framework (HyMuT) that employs adaptive hierarchical transfer MTL for a data reconstruction task, a hard parameter sharing mechanism for prediction tasks, and an adversarial learning-based classifier for domain adaptation task.

B. Self-Supervised Strategy Task

Using a related task as an auxiliary task in MTL is a classical approach. While leveraging related or similar tasks can enhance the performance of the main task, constructing suitable auxiliary tasks remains a significant challenge when dealing with the complex data of HSI.

Self-supervised strategy offers a solution by providing a predefined pretext task to train the model. The learned visual features can then be transferred to downstream tasks. Generally, shallow layers capture general low-level features such as edges, corners, and textures, while deep layers capture task-specific high-level features. This mechanism can improve the model in terms of data and feature representation. Inspired by self-supervised learning, we aim to generate additional data from the existing data itself, rather than relying on unlabeled samples, human annotations, or other datasets, in order to construct new auxiliary tasks.

However, commonly used self-supervised strategy-based tasks are typically designed based on RGB images. Remote sensing images, on the other hand, have different shooting angles, and their fine-grained nature implies that each pixel represents a distinct object. Feature extraction becomes more complex and challenging. Tasks like rotation angle prediction, jigsaw puzzles, and relative position prediction are not as effective in this context. Additionally, due to the complex spectral dimension of HSI, many tasks based on patch cutting fail to preserve the spectral information. Effectively defining self-supervised strategy-based tasks and coordinating their relationships to efficiently transfer knowledge pose further challenges.

Inspired by masked autoencoders (MAE) [42], we randomly apply some masks to the data during each iteration, following the training method of multiple iterations on few-shot learning. This approach enables the network model to capture features from all pixel positions. The mask mechanism filters out redundant information while accelerating the learning of image details. Considering the structural characteristics of HSI, we introduce a spatial–spectral joint mask reconstruction task and a spatial mask prediction task, which collectively establish an MTL framework based on self-supervised strategy.

C. Domain Adaptation Auxiliary Task

Domain adaptation is a widely used approach in transfer learning. It involves utilizing source data with abundant labeled samples and target data with limited labeled samples. In this scenario, we have a source domain D_s and a target domain D_t with respective joint distributions P_s and P_t,

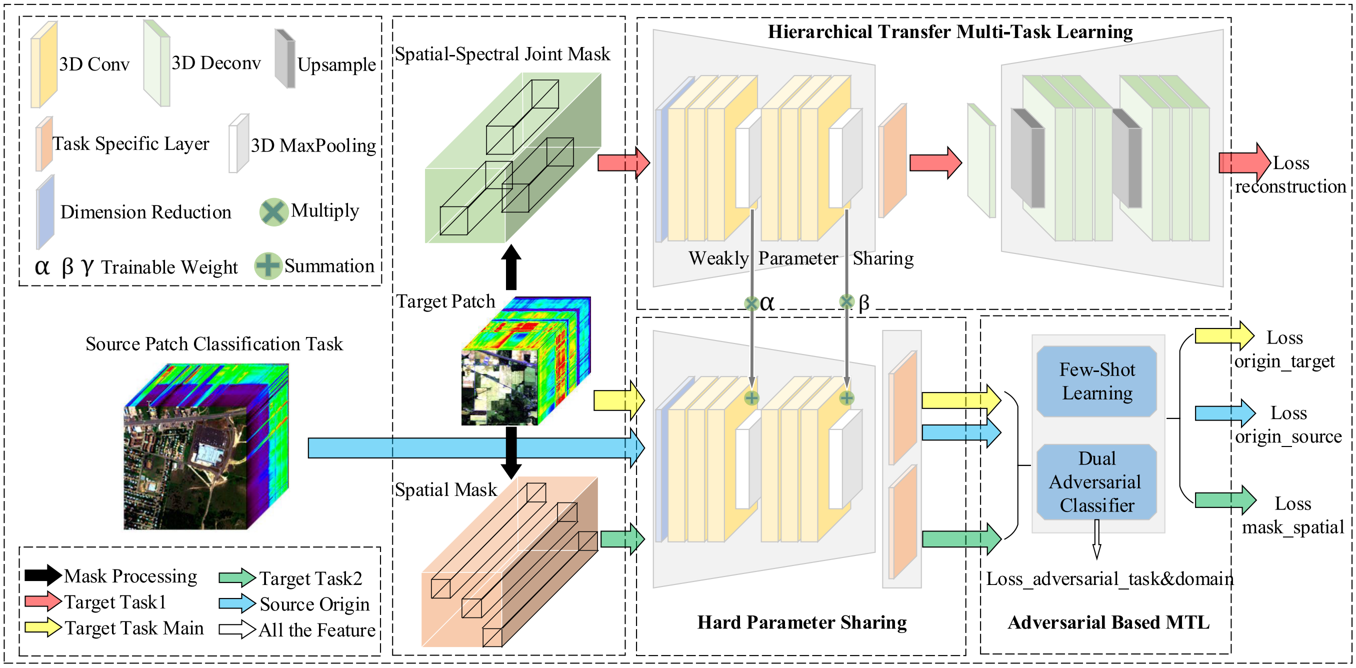


Fig. 1. Flowchart of the proposed HyMuT approach: HyMuT is composed of a similar mask prediction task, a related spatial–spectral joint mask reconstruction task, and a weakly correlated domain adaptation task. Correspondingly, it includes three MTL strategies: hierarchical transfer MTL for similar tasks, hard parameter sharing for related tasks, and dual adversarial-based MTL for weakly correlated tasks.

where $P_s \neq P_t$, and the labeled categories Y_s are different from Y_t. The source domain D_s contains more categories C_s and labeled samples compared to the target domain C_t [60], allowing knowledge transfer from the source to the target domain.

From an MTL perspective, domain adaptation plays a significant role. Adversarial learning is employed, alternating between source domain task T_s and target domain task T_t, to train the feature backbone network in mapping D_s and D_t to a shared feature space. This adversarial multitask model typically comprises three networks: a feature network, a classification network, and a domain network. The classification network minimizes the training loss for all tasks based on the feature network, while the domain network aims to discern the task of each data instance.

In response to the lack of diversity in auxiliary tasks in our MTL framework and enable the shared feature backbone network to acquire additional knowledge, we introduce an extra HSI dataset as the source domain. By employing adversarial learning methods, this model effectively incorporates a domain adaptation task, resulting in improved and stable performance.

III. Methodology

The HyMuT framework is designed to empower the few-shot learning network with a holistic comprehension of HSI data. Fig. 1 illustrates the components of HyMuT, which consist of a similar mask prediction task, a related spatial–spectral joint mask reconstruction task, and a weakly correlated domain adaptation task. These task designs are guided by the similarity of the tasks. Furthermore, it comprises three corresponding MTL strategies: hierarchical transfer MTL for similar tasks, hard parameter sharing for related tasks, and dual adversarial-based MTL for weakly correlated tasks. Different colored arrows indicate data flows corresponding to various tasks.

In Sections III-B–III-D, each MTL strategy of the HyMuT framework is described in detail, providing insights into their functionalities and interactions.

A. Auxiliary Tasks Construction

Unlike conventional RGB images, HSI contains rich redundant information that can potentially enhance performance. Motivated by the concept of self-supervised strategy, we propose three auxiliary tasks specifically designed for HSI images without requiring additional human annotations.

To prepare the HSI data, we perform data preprocessing to divide it into small patches, forming a dataset for network training. Inspired by MAE [42], we randomly incorporate masks of a specific size into each patch, thereby introducing new tasks based on the original data. Considering the HSI data, we propose a mask processing with two operations.

1) Adding a single spatial mask block to the original data to obstruct a specific spatial pixel block along with its spectral information.
2) Adding a spatial–spectral joint mask block of a certain size, where a single mask block only covers part of the spatial–spectral data. For example, in the case of the Indian Pines (IP) dataset with a size of $145 \times 145 \times 200$, it is processed into patch blocks of size $9 \times 9 \times 200$. The individual mask sizes $M1$ and $M2$ corresponding to the two tasks are $1 \times 1 \times 200$ and $2 \times 2 \times 50$, respectively.

Due to the small size of the data patch, a single pixel already contains a significant amount of HSI information. Hence, the size of the mask block should not be excessively large, and a large number of small mask blocks are beneficial for extracting redundant information. Additionally, the spatial–spectral relationship is preserved. Therefore, we adopt small mask sizes and high mask ratios. Furthermore, different tasks employ different mask ratios $R1$ and $R2$. For the similar task involving mask task prediction, a higher mask ratio of 75% is necessary to extract valuable information from redundant data. For the related reconstruction task, a mask ratio of 25% is sufficient to extract the classification effect while preserving the feasibility of classification tasks. Excessively high mask ratios could increase task difficulty. Further details on mask ratio parameter experimentation are described in Section IV-E.

For the target classification task T_t and the dataset D_t consisting of n_t training samples $\boldsymbol{X}^t = (\boldsymbol{x}_1^t, \ldots, \boldsymbol{x}_{n_t}^t)$ along with their corresponding labels $\boldsymbol{Y}^t = (y_1^t, \ldots, y_{n_t}^t)$, we construct two mask tasks T_{spa} and T_{ss}. The labels of the original data are retained as the labels for the new tasks ($\boldsymbol{Y}^t = \boldsymbol{Y}^{\text{spa}}$). The data augmented by the mask is denoted as $\boldsymbol{X}^{\text{spa}}$ and $\boldsymbol{X}^{\text{ss}}$

$$\begin{aligned}\boldsymbol{X}^{\text{spa}} &= \text{Random}\left(\boldsymbol{X}^t, R1, M1\right)\\ \boldsymbol{X}^{\text{ss}} &= \text{Random}\left(\boldsymbol{X}^t, R2, M2\right)\end{aligned} \tag{1}$$

where Random$(\cdot)$ indicates masking operation based on random sampling strategy. Ultimately, new self-supervised strategy auxiliary datasets D_{spa} and D_{ss} are formed.

To augment the task diversity, we incorporate a weakly correlated domain adaptation approach. Following the principle of domain adaptation [60], we select the Chikusei dataset as the source domain D_s, which contains a larger number of classes compared to other datasets. The source domain task T_s is a classification task similar to the target domain.

Based on the above auxiliary tasks T_{spa}, T_{ss}, and T_s, we adopt distinct network structures and training strategies for each task, as detailed in the subsequent sections.

B. Hard Parameter Sharing

The auxiliary task T_{spa} shares similar classification tasks with the target task T_t, and only a small portion of information is lost without significantly damaging the data. To achieve MTL for these similar tasks, we employ the hard parameter sharing mechanism, which is a widely used approach in neural networks for MTL [61]. This approach involves sharing the hidden layers among all tasks and maintaining task-specific output layers. In our approach, we construct a shared backbone network for the three prediction tasks and assign task-specific layers to each task. The backbone network consists of a 2-D convolution layer for dimension reduction and two groups of 3-D residual convolutions and 3-D maxpooling operations to extract spatial–spectral joint features. The task-specific layer for the three tasks is a 3-D convolution. The network structure is illustrated in Fig. 2.

During a training episode of hard parameter sharing, the patch data from T_t and T_{spa} is simultaneously fed into the network structure, generating the classification loss through the few-shot learning module.

In the few-shot learning stage, support sets and query sets are formed by sampling K and N samples for each of the C categories in the target domain, resulting in support sets of size $C \times K$ and query sets of size $N \times K$. The support set data undergoes network feature extraction, resulting in C prototypes. The query set is then classified by comparing the distance between the features extracted from the query data and the prototypes. Taking the target task T_t as an example, the classification probability of $\boldsymbol{x}_i^t$ is calculated as follows:

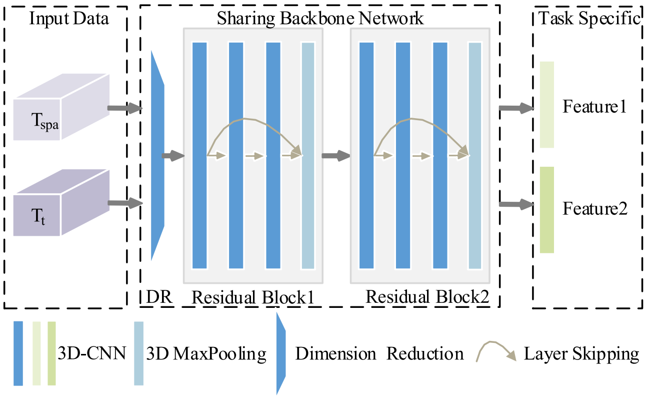


Fig. 2. Illustration framework of the proposed embedded feature extractor.

$$\boldsymbol{P}\left(\hat{y}_i^t|\boldsymbol{x}_i^t\right) = \text{Softmax}\left(-\text{ED}\left(F^t\left(\boldsymbol{x}_i^t\right), \boldsymbol{c}_k^t\right)\right) \tag{2}$$

where $\hat{y}_i^t$ is the predicted label for the sample. Softmax$(\cdot)$ is the softmax function, ED$(\cdot)$ is the Euclidean distance, $F^t(\cdot)$ is the backbone network for the target task, and $\boldsymbol{c}_k^t$ is the prototype of the kth category.

Using the probabilities obtained from the query set in the target task, we can compute the loss of the target classification by cross-entropy loss, which is expressed as follows:

$$L_{\text{CE}}^t = -\frac{1}{n_t}\sum_{i=1}^{n_t} y_i^t \log \boldsymbol{P}\left(\hat{y}_i^t|\boldsymbol{x}_i^t\right) \tag{3}$$

where n_t is the number of samples in the query set from the target task, y_i^t and $\hat{y}_i^t$ are the true label and predicted label, respectively, for the sample $\boldsymbol{x}_i^t$.

Therefore, the features vector can be inputted into the few-shot learning module, and the backbone network can be trained using the labeled samples. Similarly, the losses for the other auxiliary tasks T_s and T_{spa} are denoted as L_{CE}^s and $L_{\text{CE}}^{\text{spa}}$, respectively.

C. Hierarchical Transfer MTL

When dealing with spatial–spectral joint mask data, where numerous small mask blocks obstruct the original data, the spatial–spectral relationship is damaged seriously. In such cases, employing the aforementioned hard parameter sharing mechanism may introduce more negative impacts than benefits, while also posing a challenge to the network architecture. Specifically, as the tasks become more difficult, achieving a proper balance between shared layers and task-specific layers becomes increasingly challenging. Motivated by Misra et al. [38], we propose a hierarchical transfer method to achieve the efficient knowledge transfer between the auxiliary tasks and the main task. This MTL method is specifically designed for transferring knowledge from auxiliary tasks, particularly for complex but related tasks in deep neural networks.

To tackle the data damage in task T_{ss} while paying more attention to redundant and secondary information, we employ a symmetric autoencoder for data reconstruction. By reconstructing the patch data from the masked data, the network can infer the covered spatial pixels and spectral curves from the surrounding pixels and spectra. Through iterative training, the autoencoder ensures that all pixels and spectra of the few labeled samples are adequately covered. Consequently, the reconstruction task provides the network with a better understanding of the HSI data.

The autoencoder F_{auto} consists of a weakly parameter sharing encoder F_{en} and a symmetrical decoder F_{de}. The encoder employs a weak parameter sharing mechanism, sharing the same network structure but with different parameters and objective functions. This mechanism enhances the fusion of features from related tasks, facilitating effective knowledge transfer. On the other hand, the decoder has a symmetrical network structure and operation positions to that of the encoder. The 3-D maxpooling operation corresponds to upsample, while the 3-D convolution operation corresponds to 3-D transposed convolution operation. For the task T_{ss} with data $\boldsymbol{X}^{\text{ss}} = (\boldsymbol{x}_{\mathbf{1}}^{\text{ss}}, \ldots, \boldsymbol{x}_{\boldsymbol{n_t}}^{\text{ss}})$, the output of F_{auto} with parameter $\boldsymbol{\theta}$ is as follows:

$$\hat{\boldsymbol{X}}^{\text{ss}} = F_{\text{auto}}\left(\boldsymbol{X}^{\text{ss}}, \boldsymbol{\theta}\right) \tag{4}$$

and the loss of reconstruction task is expressed as follows:

$$L_{\text{recon}}^{\text{ss}} = \frac{1}{n_t}\sum_{i=1}^{n_t}\left(\hat{\boldsymbol{x}}^{\text{ss}} - \boldsymbol{x}^{\text{ss}}\right)^2. \tag{5}$$

The loss of the reconstruction task adopts the mean square error, and the loss is minimized to train the autoencoder to achieve data reconstruction efficiently.

To transfer knowledge from the autoencoder hierarchically, we establish two trainable connections to the backbone from different levels. Simultaneously, the features from the autoencoder are multiplied by the trainable parameters and forwarded to the backbone network with weakly shared parameters. Specifically, F_1^{ss} and F_2^{ss} represent the feature extraction networks from two residual blocks of F_{en}, respectively. Correspondingly, the network structures in the backbone network for the target classification are denoted as F_1^t and F_2^t. With the hierarchical transfer method, the output features of F_1^t and F_2^t for the target classification task are obtained as follows:

$$\begin{aligned} \boldsymbol{f_1^t} &= F_1^t\left(\boldsymbol{x^t}\right) + \alpha F_1^{\text{ss}}\left(\boldsymbol{x}^{\text{ss}}\right) \\ \boldsymbol{f_2^t} &= F_2^t\left(\boldsymbol{f_1^t}\right) + \beta F_2^{\text{ss}}\left(F_1^{\text{ss}}\left(\boldsymbol{x}^{\text{ss}}\right)\right) \end{aligned} \tag{6}$$

where $\boldsymbol{x}^{\text{ss}}$ and $\boldsymbol{x^t}$ represent the inputs from tasks T_{ss} and T_t, respectively, and α and β are the trainable parameters mentioned above.

D. Dual Adversarial Classifier

Adversarial-based MTL is a commonly used method for handling auxiliary tasks with significant differences. It effectively extracts the invariance between tasks, facilitating knowledge transfer. In the case of HSI data, there exists a substantial domain gap due to variations in sensors, weather conditions, and target categories. Among the auxiliary tasks, the source domain task T_s poses the greatest challenge. Furthermore, the prediction tasks with hard parameter sharing still require effective extraction of task-invariant features. To address these challenges, we propose the integration of a dual adversarial classifier. This classifier aims to handle the more difficult source domain task and assists the hard parameter sharing mechanism in extracting features across prediction tasks.

The purpose of a dual adversarial classifier is to extract domain invariant features from the source and target domains, as well as task-invariant features from the target domain classification task and spatial mask classification task. The source domain classification task is processed by the same backbone network used for the target domain classification task, and the resulting features are marked with tags 0 and 1, respectively. Similarly, the features generated by the auxiliary classification task and the target domain classification task receive different labels. These features and their corresponding labels are then fed into the dual adversarial classifier, which employs adversarial strategies to make it indistinguishable from which domain or task the features originate, thus achieving the extraction of invariant features. Unlike conventional gradient propagation algorithms, we introduce a gradient inversion layer [56] to update parameters. In order to speed up the confusion of domain discriminators, we add the posterior probability information of the classifier to the feature vector.

Considering that both prediction tasks and domain adaptive tasks share the same strategy, we use the domain task as an example. Let $\boldsymbol{\theta_{st}}$ represent the parameters of the backbone network and the target-specific layer F^{st}. $\boldsymbol{\theta_d}$ means the parameters of the classifier D. The objective function is defined as follows:

$$L_D^{\text{st}} = \max_{\boldsymbol{\theta_{st}}} \min_{\boldsymbol{\theta_d}} L_D^{\text{st}}(\boldsymbol{\theta_{st}}, \boldsymbol{\theta_d}). \tag{7}$$

Among them, $L_D^{\text{st}}(\boldsymbol{\theta_{st}}, \boldsymbol{\theta_d})$ can be expressed in detail as follows:

$$\begin{aligned} &L_D^{\text{st}}(\boldsymbol{\theta_{st}}, \boldsymbol{\theta_d}) \\ &\quad = \frac{1}{n_t}\sum_{i=1}^{n_t} L_D\left(D\left(\left(F\left(\boldsymbol{x_i}^{\text{st}}, \boldsymbol{\theta_{st}}\right), \boldsymbol{g_i}^{\text{st}}\right); \boldsymbol{\theta}_d\right), d_i^{\text{st}}\right) \end{aligned} \tag{8}$$

where L_D is the loss of domain classification and n_t is the total number of samples. $\boldsymbol{g_i}^{\text{st}}$ and d_i^{st} are the posterior probability and the label of task corresponding to the sample $\boldsymbol{x_i}^{\text{st}}$, respectively. Notably, $\boldsymbol{g_i}^{\text{st}}$ could accelerate the alignment of tasks.

Same as (7), we can define the objective function for the prediction task as follows:

$$L_D^p = \max_{\boldsymbol{\Theta}} \min_{\boldsymbol{\theta_d}} L_D^p(\boldsymbol{\Theta}, \boldsymbol{\theta_d}) \tag{9}$$

where Θ presents the parameters of backbone and task-specific layers for the three prediction tasks. By combining (7) and (9), we could formulate the objective of the dual adversarial classifier as follows:

$$\begin{aligned} &\max_{\boldsymbol{\Theta}} \min_{\boldsymbol{\theta_d}} L_D(\boldsymbol{\Theta}, \boldsymbol{\theta_d}) \\ &\quad = \max_{\boldsymbol{\Theta}} \min_{\boldsymbol{\theta_d}}\left(L_D^{\text{st}}(\boldsymbol{\theta_{st}}, \boldsymbol{\theta_d}) + L_D^p(\boldsymbol{\Theta}, \boldsymbol{\theta_d})\right). \end{aligned} \tag{10}$$

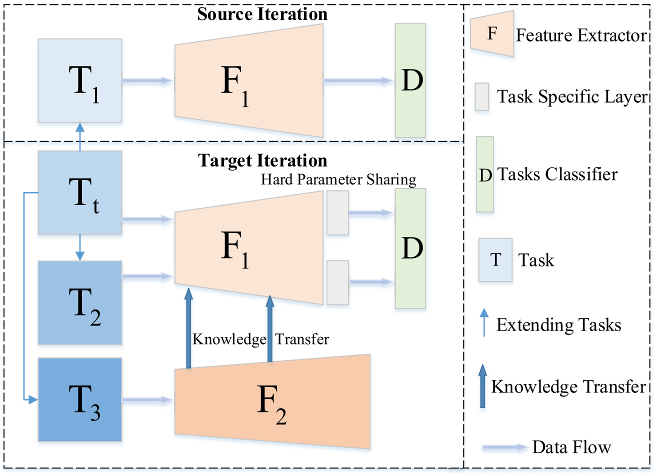


Fig. 3. Flowchart of the overall simple framework.

It is important to note that the parameter $\mathbf{\Theta}$ includes $\boldsymbol{\theta}_{st}$, representing the parameters of the backbone network with the target-specific layer.

E. Overall Objective

To summarize, the proposed HyMuT method involves four tasks and employs three MTL methods. Through alternating training of two domains, we train the backbone network using source classification task and then train the whole network with the rest target tasks simultaneously. Without loss of generality, the framework foundation we proposed can be expressed as Fig. 3.

On the one hand, the tasks in the figure can be freely set with appropriate auxiliary tasks, such as self-supervised-based tasks and other homogeneous image tasks. Besides, the feature extraction network can also be modified according to the characteristics of the data, and it is possible to employ attention mechanisms and potential network structure. Consequently, a possible task could choose different MTL methods for knowledge transfer.

On the other hand, our model could establish more task branches according to increasing workload. The network achieves adversarial learning through alternating iterations with task classifier, hard parameter is achieved through task-specific layers, and knowledge transfer from F2 to F1. Therefore, more alternating iterations, task-specific layers, and similar autoencoders could be added easily to handle more tasks.

It is important to note that appropriate experiments and methods need to be used to evaluate the similarity between tasks, in order to fill in appropriate MTL methods. The loss function for the source iteration is given by the sum of the source classification loss and the domain classification loss

$$L_{\mathrm{CE}}^{s} + L_{D}^{\mathrm{st}}. \tag{11}$$

In the target iteration, in addition to similar target classification loss and the domain classification loss, the function consists of reconstruction loss, mask prediction loss, and task classification loss

$$L_{\mathrm{CE}}^{t} + L_{D}^{\mathrm{st}} + \lambda L_{\mathrm{CE}}^{\mathrm{spa}} + \eta L_{D}^{p} + \omega L_{\mathrm{recon}}^{\mathrm{ss}}. \tag{12}$$

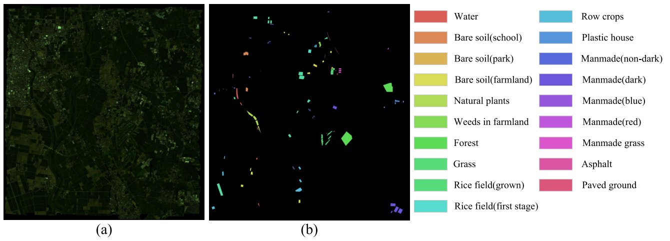


Fig. 4. Chikusei. (a) Pseudo color image (bands 12, 41, and 55). (b) Ground truth map.

Here, $L_{\mathrm{recon}}^{\mathrm{ss}}$ represents the loss from the spatial–spectral reconstruction task in (1). The target classification loss L_{CE}^{t} and spatial mask classification $L_{\mathrm{CE}}^{\mathrm{spa}}$ are from (3). Additionally, the domain classification loss L_{D}^{st} and the task classification loss L_{D}^{p} are included in the overall loss. Furthermore, we introduce trade-off factors λ, η, and ω to control the importance of these auxiliary tasks. The parameters' aspect is elaborated in Section IV-C.

IV. Experimental Result and Discussion

HyMuT is an MTL method designed for few-shot HSI classification. In terms of data requirements, HyMuT only requires two datasets: one serving as the source domain and the other as the target domain. Therefore, for our experiments, we selected four widely used hyperspectral remote sensing datasets as the target domain: IP, Salinas (SA), Houston, and Pavia Center (PC). As for the source domain, we selected the Chikusei dataset for its diversity.

We select several state-of-the-art few-shot HSI classification methods for comparison with our proposed HyMuT. These include three classical methods: support vector machine (SVM) [6], 3-D CNN [62], and self-supervised deep metric model (DMM) [63]. Additionally, we include five cross-domain methods: Two-CNN [64], relation network few-shot classification (RN-FSC) [26], DCFSL [31], Gia-CFSL [33], and CMFSL [32]. These benchmark methods provide a basis for performance comparison with our proposed approach.

A. Experimental Datasets Description

1) Source Domain: The Chikusei is an airborne HSI dataset collected using the Headwall Hyperspec-VNIR-C imaging sensor. It covers agricultural and urban areas in the Chikusei region. The dataset consists of 19 classes and has a spatial resolution of 2.5 m per pixel, with dimensions of 2517 by 2335 pixels. It comprises 128 spectral bands ranging from 363 to 1018 nm. In Fig. 4, we present a pseudo color image created by selecting the 12th, 41st, and 55th bands, along with the corresponding ground truth map. Table I provides an overview of the samples in the Chikusei dataset.

2) Target Domain: The IP dataset is a widely used HSI dataset obtained using the Airborne Visible and InfraRed Imaging Spectrometer (AVIRIS) sensor. It was collected over an agricultural area in Indiana. The dataset contains 145 × 145 pixels and consists of 220 spectral bands covering a range from 0.4 to 2.5 μm. To facilitate classification modeling, 20 bands affected by water vapor were removed, leaving 200 spectral bands. The IP dataset includes 16 landscape types

TABLE I
NUMBERS OF PIXELS AND LAND COVER CLASSES IN THE CHIKUSEI

ID	Class	Numbers
1	Water	2845
2	Bare soil(school)	2859
3	Bare soil(park)	286
4	Bare soil(farmland)	48525
5	Natural plants	4297
6	Weeds in farmland	1108
7	Forest	20516
8	Grass	6515
9	Rice field(grown)	13369
10	Rice field(first stage)	1268
11	Row crops	5961
12	Plastic house	2193
13	Manmade(non-dark)	1220
14	Manmade(dark)	7664
15	Manmade(blue)	431
16	Manmade(red)	222
17	Manmade grass	1040
18	Asphalt	801
19	Paved ground	145
	Total	77592

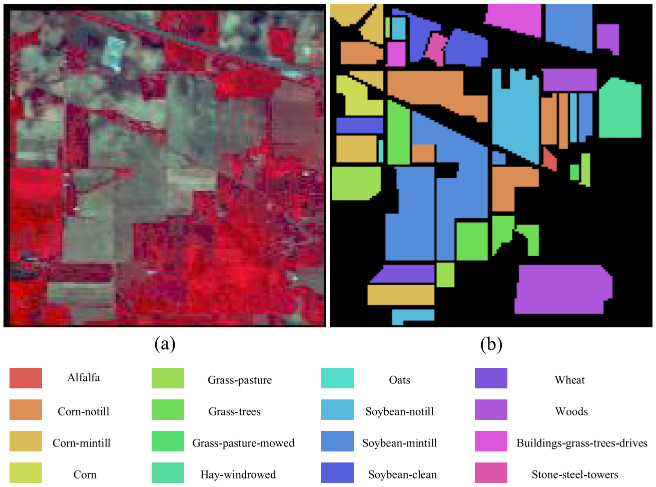


Fig. 5. IP. (a) Pseudo color image (bands 11, 21, and 43). (b) Ground truth map.

such as crops, trees, and bare soil. Table II provides the pixel counts for each landscape class. In Fig. 5, we showcase a pseudo color image created using the 11th, 21st, and 43rd bands, along with the corresponding ground truth map.

The PC dataset, also acquired using Reflective Optics Spectrographic Imaging System (ROSIS) over Pavia, comprises 1906 × 715 pixels and 102 spectral bands. After removing noisy bands, the wavelength range spans from 0.43 to 0.86 μm. Table III provides an overview of the nine categories present in the dataset and their quantities. Fig. 6 displays a pseudo color image created using the 12th, 41st, and 55th bands, along with the ground truth map.

The Houston dataset was acquired by the ITRES CASI-1500 sensor over the University of Houston and its surroundings in Texas, USA. The spatial resolution is 2.5 m. There are 15 labeled categories in the Houston dataset as shown in Table IV. There are 349 × 1905 pixels and 144 spectral bands after absorption bands were removed in the wavelength range from 0.36 to 1.05 μm. Fig. 7 is a pseudo color image using the 12th, 41st, and 55th bands, with the ground truth map.

TABLE II
NUMBERS OF PIXELS AND LAND COVER CLASSES IN THE IP

ID	Class	Numbers
1	Alfalfa	46
2	Corn-notill	1428
3	Corn-mintill	830
4	Corn	237
5	Grass-pasture	483
6	Grass-trees	730
7	Grass-pasture-mowed	28
8	Hay-windrowed	478
9	Oats	20
10	Soybean-notill	972
11	Soybean-mintill	2455
12	Soybean-clean	593
13	Wheat	205
14	Woods	1265
15	Buildings-grass-trees-drives	386
16	Stone-steel-towers	93
	Total	10249

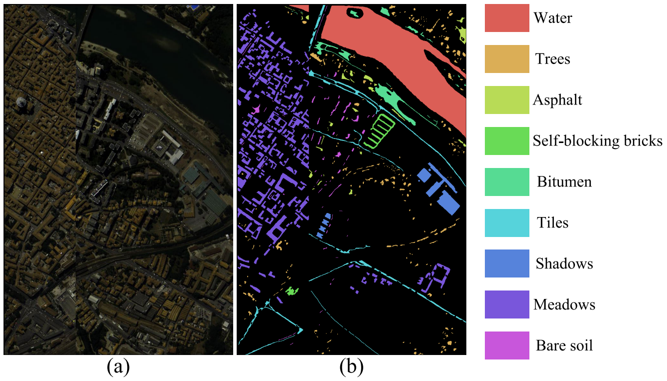


Fig. 6. PC. (a) Pseudo color image (bands 12, 41, and 55). (b) Ground truth map.

TABLE III
NUMBERS OF PIXELS AND LAND COVER CLASSES IN THE PC

ID	Class	Numbers
1	Water	65971
2	Trees	7598
3	Asphalt	3090
4	Self-blocking bricks	2685
5	Bitumen	6584
6	Tiles	9248
7	Shadows	7287
8	Meadows	42826
9	Bare soils	2863
	Total	148152

The SA dataset represents the Salinas Valley in California and was captured using the AVIRIS sensor. It consists of 512 × 127 pixels and encompasses 224 bands. Similar to the IP dataset, 20 bands affected by water vapor were removed, resulting in 204 remaining spectral bands ranging from 0.4 to 2.5 μm. The ground truth map and a pseudo color image created using the 11th, 21st, and 43rd bands are presented in Fig. 8. For more detailed information on pixel counts and landscape types, please refer to Table V.

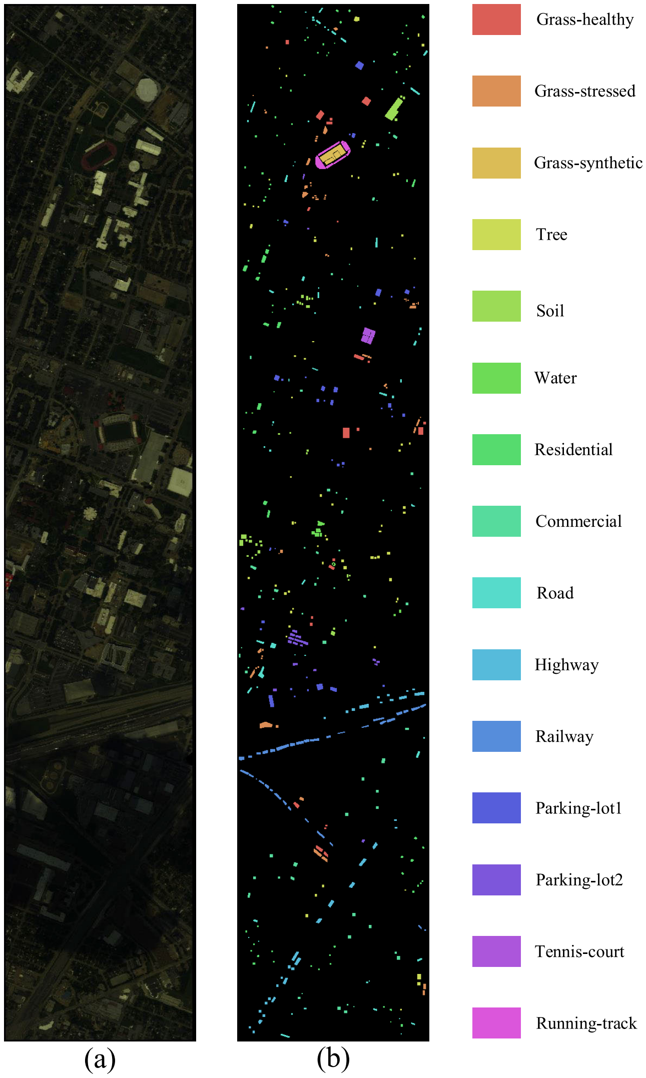


Fig. 7. Houston. (a) Pseudo color image (bands 12, 41, and 55). (b) Ground truth map.

TABLE IV

NUMBERS OF PIXELS AND LAND COVER CLASSES IN THE HOUSTON

ID	Class	Numbers
1	Grass-healthy	1251
2	Grass-stressed	1254
3	Grass-synthetic	697
4	Tree	1244
5	Soil	1242
6	Water	325
7	Residential	1268
8	Commercial	1244
9	Road	1252
10	Highway	1227
11	Railway	1235
12	Parking-lot1	1233
13	Parking-lot2	469
14	Tennis-court	428
15	Running-track	660
	Total	15029

In our experiment, we employ four metrics to quantitatively evaluate the classification performance: class-specific accuracy, overall accuracy (OA), average accuracy (AA), and kappa coefficient (Kappa). These metrics provide comprehensive measures of the classification accuracy.

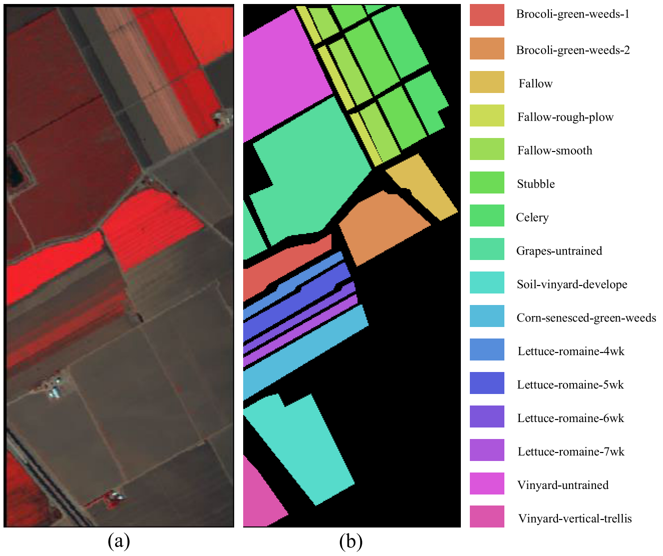


Fig. 8. SA. (a) Pseudo color image (bands 11, 21, and 43). (b) Ground truth map.

TABLE V

NUMBERS OF PIXELS AND LAND COVER CLASSES IN THE SA

ID	Class	Numbers
1	Brocoli-green-weeds-1	2009
2	Brocoli-green-weeds-2	3726
3	Fallow	1976
4	Fallow-rough-plow	1394
5	Fallow-smooth	2678
6	Stubble	3959
7	Celery	3579
8	Grapes-untrained	11271
9	Soil-vinyard-develope	6203
10	Corn-senesced-green-weeds	3278
11	Lettuce-romaine-4wk	1068
12	Lettuce-romaine-5wk	1927
13	Lettuce-romaine-6wk	916
14	Lettuce-romaine-7wk	1070
15	Vinyard-untrained	7268
16	Vinyard-vertical-trellis	1807
	Total	54129

B. Ablation Study

The design of the three auxiliary tasks is a crucial aspect of HyMuT, and the three MTL methods support these tasks. To assess the contribution of these auxiliary tasks and their corresponding learning strategies, ablation studies are conducted on each task using the four datasets in this section.

1) Hard Parameter Sharing: The primary target classification task and the auxiliary mask prediction task employ the hard parameter sharing mechanism to enhance performance. To analyze the contribution of these two auxiliary tasks, we design experiments by removing each task individually. Specifically, we refer to the experiments without the auxiliary spatial prediction task as "no T_{spa}."

2) Hierarchical Transfer MTL: The reconstruction task T_{ss} is closely related to the target classification task and plays a crucial role in hierarchical transfer MTL. In this module,

TABLE VI
ABLATION STUDY (%) ON THE OA INDEX FOR DATASETS

Dataset	no T_{ss}	no T_{spa}	no task	no domain	HyMuT
Salinas	91.01	89.59	90.49	90.44	**91.11**
Indian Pines	69.53	67.18	69.84	69.01	**70.57**
Houston	78.52	77.18	78.47	78.26	**78.61**
Pavia Center	97.53	97.36	97.48	97.17	**97.70**

TABLE VII
RESULTS (%) WITH DIFFERENT PARAMETERS FOR $R1$ AND $R2$

		$R1$		
		25%	50%	75%
$R2$	25%	90.70	90.75	**91.11**
	50%	90.76	90.81	90.64
	75%	90.81	90.98	90.87

we conduct experiments by removing the reconstruction task to assess the performance of their combination. We denote these experiments as “no T_{ss}.”

3) Dual Adversarial Classifier: The dual adversarial classifier is based on adversarial learning in MTL and is applied to both the domain adaptation task and the three prediction tasks. To demonstrate the effect of this classifier, we design experiments to remove both types of tasks separately. We refer to the experiments without any tasks as “no task” and the experiments without any domain-related tasks as “no domain.”

The results of these experiments are shown in Table VI. It can be seen that the spatial mask prediction task has the highest contribution to the target classification, while the dual adversarial classifier and the corresponding tasks have a moderate contribution. Due to the complexity of the autoencoder, spatial–spectral joint mask reconstruction task has the smallest effect, but when these modules are combined, HyMuT works best.

C. Parameter Tuning

HyMuT involves several hyperparameters that require adjustment. Therefore, we conducted a series of comparative experiments to explore the impact of these parameters on the model. Specifically, we focused on the mask ratios $R1$ and $R2$, as well as the trade-off parameters λ and ω. We selected two sets of hyperparameter values from the sets {0.05, 0.1, 0.2} and {25%, 50%, 75%}, respectively. Notably, η is used to assist in the hard parameter sharing mechanism, and {0.01, 0.05, 0.1} are suitable to control the weight.

The mask ratio indicates the degree of damage to the HSI. A higher mask ratio corresponds to a stronger focus of the network on a single small data block, making it more challenging to perform multiple tasks simultaneously. Balancing the difficulty of the auxiliary tasks and the level of assistance is a key aspect of MTL.

Table VII illustrates the influence of mask ratios on model performance. It is worth noting that the mask tasks operate within the overall network, and the parameters can impact the model’s overall effectiveness. In the parameter tuning experiments, we set 0.1 for λ, ω, and η, respectively. From Fig. 9, higher mask ratios for the prediction mask tasks lead to

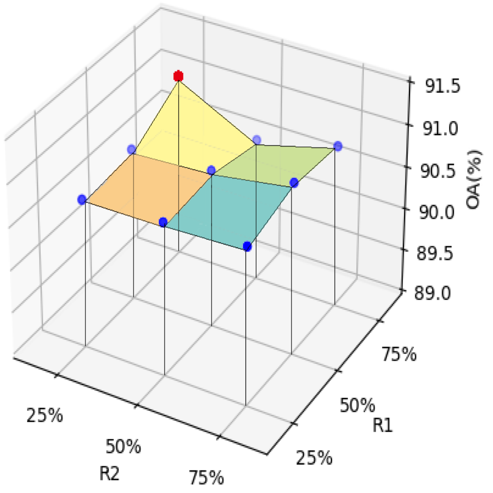


Fig. 9. Parameter tuning of $R1$ and $R2$.

TABLE VIII
RESULTS (%) WITH DIFFERENT PARAMETERS FOR λ AND ω WHEN $\eta = 0.01$

$\eta = 0.01$		λ		
		0.05	0.1	0.2
ω	0.05	90.55	90.20	90.89
	0.1	90.37	90.70	90.80
	0.2	91.00	90.86	**91.01**

TABLE IX
RESULTS (%) WITH DIFFERENT PARAMETERS FOR λ AND ω WHEN $\eta = 0.05$

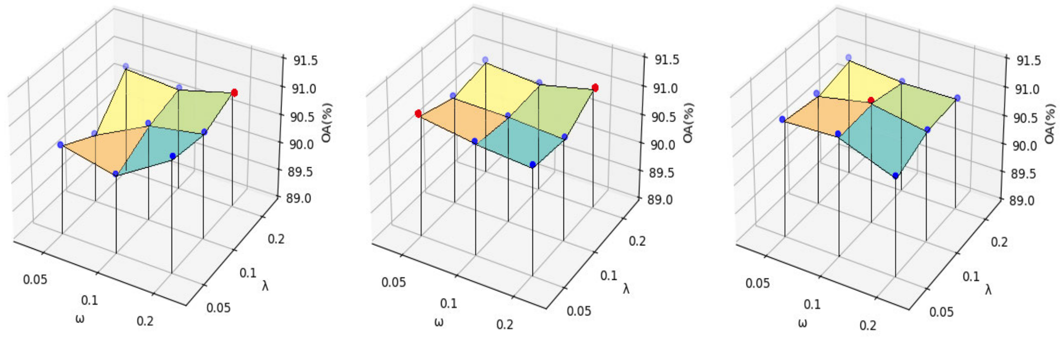
$\eta = 0.05$		λ		
		0.05	0.1	0.2
ω	0.05	**91.09**	90.87	90.99
	0.1	90.94	90.83	90.88
	0.2	90.87	90.78	**91.09**

TABLE X
RESULTS (%) WITH DIFFERENT PARAMETERS FOR λ AND ω WHEN $\eta = 0.1$

$\eta = 0.1$		λ		
		0.05	0.1	0.2
ω	0.05	91.01	90.93	91.05
	0.1	91.08	**91.11**	90.92
	0.2	90.70	90.90	90.93

improved model performance, while the reconstruction mask task shows worse performance at a 50% mask rate generally. Consequently, we select the mask rates of 75% for $R1$ and 25% for $R2$, which yields the most significant improvements in performance.

Furthermore, the trade-off parameters determine the weight of the auxiliary prediction tasks, which affect the learning efficiency and optimization direction of the target iteration MTL. The spatial mask task T_{spa} is determined by the value of λ since they involve similar auxiliary tasks. On the other hand, ω is used to control the reconstruction task T_{ss}. And η is to control the task classifier. We designed experiments with different combinations of these parameters assuming $R1$ is 75% and $R2$ is 25%, as shown in Tables VIII–X. From the tables, it can be seen that when the values of λ and ω are



(a) (b) (c)

Fig. 10. OA (%) with different λ, ω, and η parameters on SA. (a) $\eta = 0.01$. (b) $\eta = 0.05$. (c) $\eta = 0.1$.

TABLE XI

Classification Results (%) on IP Dataset With Five Labeled Sample Each Class

Class	SVM	3D-CNN	Two-CNN	RN-FSC	DMM	DCFSL	Gia-CFSL	CMFSL	HyMuT
1	84.86±10.33	76.30±15.72	62.17±22.41	22.12±7.86	83.48±14.59	92.20±8.64	91.60±7.69	88.54±1.62	**98.05±2.84**
2	24.93±15.20	35.64±12.29	26.19±15.40	42.26±0.65	39.03±11.84	46.20±8.34	50.46±7.85	44.33±10.05	**60.51±9.87**
3	22.86±18.16	24.80±12.26	16.95±9.82	26.84±4.23	45.54±6.62	52.73±8.47	44.88±9.45	56.27±4.88	**65.50±10.42**
4	22.45±14.31	37.48±9.65	29.09±15.97	48.67±12.00	51.31±7.63	**84.14±7.36**	81.61±10.76	81.13±4.94	78.19±15.15
5	14.68±21.44	53.79±16.79	33.82±18.79	66.49±6.21	68.36±21.73	77.15±7.76	70.76±8.08	**79.90±8.31**	73.93±7.51
6	39.88±18.98	73.96±12.54	51.67±15.74	66.05±5.38	84.85±4.24	83.50±11.36	84.25±6.89	84.48±7.80	**87.45±4.87**
7	92.98±1.12	88.69±10.76	87.50±8.52	49.13±23.48	**100.00±0.00**	99.13±1.74	97.10±5.75	**100.00±0.00**	**100.00±0.00**
8	36.32±16.42	72.13±15.71	61.09±17.29	88.82±2.24	**92.43±2.41**	83.34±16.11	91.12±4.23	84.93±13.49	81.61±15.57
9	78.00±8.62	97.67±4.42	87.50±10.31	21.45±14.10	**100.00±0.00**	92.00±0.00	99.26±2.22	**100.00±0.00**	**100.00±0.00**
10	37.14±17.78	40.97±12.67	38.40±17.90	43.60±13.18	63.44±3.34	67.18±5.98	62.68±7.84	60.36±4.46	**68.11±5.28**
11	38.58±21.36	36.93±15.24	43.98±19.57	**69.53±3.73**	46.22±9.29	50.94±18.05	66.54±5.99	65.42±7.79	61.44±11.39
12	16.47±9.47	28.01±11.61	11.75±5.82	37.76±3.43	37.64±9.39	37.04±5.08	42.06±10.39	41.04±11.43	**58.78±18.39**
13	94.65±2.68	87.38±9.98	84.76±9.43	46.31±18.43	99.32±0.96	99.40±0.80	97.11±4.58	**99.70±0.33**	98.90±1.11
14	81.00±16.81	71.37±16.58	69.81±14.87	**92.13±1.34**	91.57±1.69	78.32±8.27	87.10±6.11	86.14±5.39	84.54±16.45
15	12.72±7.85	22.88±9.60	33.24±14.63	36.38±5.04	56.94±6.71	**77.85±7.67**	68.74±10.56	79.66±8.07	74.02±16.86
16	85.13±1.82	69.43±21.94	79.09±14.22	66.44±2.12	98.71±1.72	**99.55±0.56**	97.47 ±4.01	88.41±2.05	98.86±1.02
OA	38.72±4.34	45.30±5.46	41.64±3.46	50.04±2.64	60.33±3.21	66.10±4.46	67.42±1.80	67.32±1.43	**70.57±3.47**
AA	48.92±1.81	56.95±5.53	51.06±2.92	51.50±3.19	72.43±1.04	76.29±1.97	77.05±1.59	77.52±0.90	**80.62±1.91**
Kappa	31.36±4.03	39.21±5.71	35.08±3.30	44.50±2.66	55.75±3.35	61.54±1.57	63.10±2.03	63.36±1.44	**66.88±3.63**

equal, the better the model's performance, which also means that auxiliary tasks provide more assistance for the main task. The visualization diagram of the parameter tuning experiments is shown in Fig. 10. The red dot represents the maximum value of this set of parameter experiments

All experiments are conducted using the PyTorch framework, and the evaluations are performed on an NVIDIA GeForce RTX 3090 24 GB graphics card. The input is set to a patch size of 9×9×Bands. The optimization scheme employed is adaptive moment estimation (Adam) with a learning rate of 1×10^{-3}. The convolutional kernel weights and linear layers are initialized using Xavier normalization.

Additionally, the trade-off factors λ, η, and ω are set to 0.2, 0.01, and 0.2 to balance the different loss terms. Each iteration of the process involves performing the N-way K-shot task. Following the principles of meta-learning, we set K to 1, indicating one labeled sample per category, and the number of query sets is set to 19 for each category.

D. Classification Results

To demonstrate the effectiveness of HyMuT in the few-shot scenario, we compare it with several well-established HSI classification models. These models encompass various domains, including classical machine learning SVM, deep learning 3-D CNN, self-supervised learning DMM, transfer learning Two-CNN, and four different cross-domain few-shot learning algorithms RN-FSC, DCFSL, Gia-CFSL, and CMFSL. By evaluating HyMuT against these diverse approaches, we aim to highlight its superiority in handling limited labeled samples.

Given the focus of HyMuT on few-shot HSI classification, we randomly select the five labeled samples from the target domain. It is worth mentioning that Two-CNN, being a semisupervised method, utilizes both labeled and unlabeled samples. Moreover, in the case of DCFSL, Gia-CFSL, and CMFSL, the Chikusei dataset is utilized as the source domain for the classification task. Additionally, RN-FSC additionally employs Botswana and Kennedy Space Center to form a larger source domain.

Specifically, SVM solves linear separable problems by mapping data to a high-dimensional space. 3-D CNN effectively extracts spatial–spectral features from HSI data using 3-D convolution kernels. Two-CNN learns joint spectral–spatial features through a two-branch architecture, while the source domain and target domain data are supposed to be from the same sensor; as a result, PC and Pavia University (PU) are the source domain and target domain of each other, which is the same as IP and SA. DMM is a supervised FSL method that maps the data to the metric space, and then selects pairs of samples for similarity learning. And RN-FSC introduces the cross-domain idea on the basis of FSL. DCFSL learns a common feature space for the source and target domains. Gia-CFSL introduces a domain adaptation strategy based on graph information. CMFSL transforms samples into

TABLE XII

CLASSIFICATION RESULTS (%) ON SA DATASET WITH FIVE LABELED SAMPLE EACH CLASS

Class	SVM	3D-CNN	Two-CNN	RN-FSC	DMM	DCFSL	Gia-CFSL	CMFSL	HyMuT
1	95.95±3.25	74.93±32.27	94.27±2.61	**99.95±0.10**	99.54±0.77	99.10±0.77	98.02±1.62	98.54±1.80	98.84±1.82
2	69.68±14.74	70.74±38.05	86.89±7.01	95.60±3.46	97.78±0.34	99.61±0.38	98.93±1.27	**99.75±0.33**	99.55±0.42
3	30.08±9.33	63.99±31.53	53.70±13.00	98.99±0.84	96.46±3.48	96.51±3.50	96.06±0.39	**99.23±0.52**	98.13±1.00
4	94.96±1.46	91.82±49.90	90.11±5.91	**99.55±0.97**	87.42±3.37	99.41±0.73	94.82±0.78	99.16±0.90	98.99±0.57
5	96.38±0.35	**96.59±6.10**	92.30±4.53	94.96±4.37	95.35±2.38	92.97±1.83	90.44±2.07	94.20±3.46	92.59±0.82
6	94.44±1.30	99.41±0.27	96.29±0.39	99.36±0.57	91.97±3.44	**99.64±0.44**	99.58±0.56	98.56±0.96	99.33±0.86
7	96.47±1.27	74.96±35.31	85.01±15.99	87.35±1.97	**99.80±0.17**	99.67±0.24	99.76±0.15	99.47±0.49	99.14±0.59
8	51.30±18.95	60.60±27.07	55.11±20.16	**86.94±3.26**	66.73±12.93	79.64±3.47	81.65±5.55	75.87±11.11	80.85±1.24
9	90.33±11.67	91.59±20.35	92.14±3.81	88.31±3.82	98.70±1.54	99.17±1.51	**99.65±0.36**	98.38±2.15	98.64±1.07
10	9.64±19.25	50.69±20.86	50.08±16.85	88.93±8.06	87.64±4.62	77.73±11.93	81.92±5.36	**89.73±4.50**	87.13±5.84
11	74.67±23.22	84.19±17.26	76.09±16.07	**99.13±1.41**	96.11±4.12	98.53±0.81	91.93±1.59	97.99±2.37	98.53±0.69
12	90.14±5.56	78.69±29.97	66.96±17.80	77.76±10.95	**99.77±0.45**	99.23±1.31	99.48±0.49	99.26±1.93	99.29±0.86
13	95.17±5.14	93.09±17.41	93.77±7.64	73.63±12.17	96.93±2.09	**99.45±0.33**	92.83±0.53	99.26±0.77	99.26±0.61
14	86.32±1.55	85.89±16.67	83.07±7.98	94.59±3.42	98.67±0.99	98.65±0.62	98.75±0.35	**98.98±0.85**	98.81±0.23
15	57.17±17.39	45.22±26.32	44.56±19.01	80.03±3.50	**81.34±10.20**	69.44±9.88	71.80±2.53	77.54±8.75	77.33±9.34
16	50.11±17.88	64.52±8.46	68.75±8.26	91.00±9.28	92.74±8.09	91.39±7.30	89.94±8.00	92.90±6.30	**94.36±3.32**
OA	68.92±3.08	69.73±13.59	71.45±3.55	87.84±1.81	89.02±2.75	89.23±1.16	89.66±0.53	90.26±2.08	**91.11±1.06**
AA	73.93±2.30	74.62±14.26	76.82±2.91	89.73±1.30	92.93±1.10	93.76±1.02	92.85±0.96	94.93±0.87	**95.05±0.72**
Kappa	65.80±3.55	66.65±13.87	68.65±3.79	87.43±2.00	87.83±3.03	87.26±2.50	88.84±0.58	89.18±2.30	**90.12±1.19**

TABLE XIII

CLASSIFICATION RESULTS (%) ON HOUSTON DATASET WITH FIVE LABELED SAMPLE EACH CLASS

Class	SVM	3D-CNN	Two-CNN	RN-FSC	DMM	DCFSL	Gia-CFSL	CMFSL	HyMuT
1	81.21±2.02	86.62±5.72	94.25±2.21	78.49±14.79	84.22±10.07	90.20±6.73	88.47±6.74	89.54±6.45	**96.30±3.07**
2	45.80±23.24	61.36±14.95	73.18±18.42	52.81±21.62	76.66±15.02	84.09±9.47	83.62±10.10	**86.02±9.91**	85.26±4.51
3	**98.62±0.28**	89.99±4.85	86.75±2.01	91.68±4.74	91.24±3.77	94.70±6.08	91.80±7.30	93.80±5.94	94.15±3.82
4	88.88±2.98	63.91±19.94	74.58±5.11	88.64±5.17	93.13±3.47	89.98±4.76	90.37±4.86	**94.83±2.47**	93.84±2.23
5	74.08±5.58	67.04±15.74	65.38±2.22	82.52±29.76	95.60±6.63	**97.62±1.73**	93.23±6.82	95.88±6.89	97.18±3.13
6	82.30±1.53	53.39±10.20	62.93±9.74	**85.46±5.29**	75.48±19.54	81.68±5.18	75.27±6.91	81.20±6.67	80.66±7.30
7	12.87±12.35	40.98±7.08	41.60±5.25	42.78±13.84	54.91±7.69	61.43±9.75	56.52±11.64	**68.12±11.57**	66.08±12.27
8	14.03±3.19	46.79±10.36	44.39±6.14	33.00±15.88	40.84±15.47	49.13±9.56	47.60±11.16	51.94±14.57	**55.17±10.11**
9	**75.92±6.41**	43.86±10.93	49.44±17.28	50.93±8.26	54.84±12.52	54.50±13.98	51.80±8.61	62.47±11.95	60.58±5.89
10	5.29±10.59	49.89±10.17	47.31±7.94	49.16±11.09	60.47±14.75	65.15±6.60	57.72±10.47	70.87±10.00	**72.39±12.24**
11	28.65±25.10	31.20±4.93	28.61±4.77	39.64±19.86	38.16±10.16	57.89±15.63	55.21±16.24	62.14±16.17	**63.50±19.09**
12	5.39±6.55	43.92±8.80	35.81±12.80	53.40±22.07	68.63±16.55	**78.10±18.08**	77.47±13.39	75.88±16.47	73.68±18.77
13	10.35±9.56	35.51±6.62	44.83±17.18	84.95±12.92	**90.21±5.68**	84.58±9.30	75.77±20.06	89.70±6.55	84.37±9.80
14	95.01±3.14	58.83±13.29	56.40±7.83	87.00±10.26	92.98±7.21	89.21±4.28	89.80±3.87	**95.51±4.37**	95.28±4.20
15	92.83±1.44	62.63±17.48	72.64±4.51	84.77±11.01	95.22±1.90	92.59±6.40	91.63±4.97	96.15±3.72	**97.93±2.08**
OA	48.55±3.11	54.60±7.82	56.88±2.94	61.77±3.12	70.24±2.25	75.24±1.25	72.47±2.18	78.25±2.12	**78.61±2.21**
AA	54.08±2.55	55.73±8.06	58.54±2.93	67.02±2.62	74.17±2.75	78.06±1.08	75.08±2.26	80.94±1.68	**81.09±2.19**
Kappa	44.80±3.27	51.03±8.42	53.46±3.20	58.82±3.33	67.92±2.43	73.26±1.35	70.26±2.36	76.51±2.29	**76.89±2.38**

TABLE XIV

CLASSIFICATION RESULTS (%) ON PC DATASET WITH FIVE LABELED SAMPLE EACH CLASS

Class	SVM	3D-CNN	Two-CNN	RN-FSC	DMM	DCFSL	Gia-CFSL	CMFSL	HyMuT
1	99.30±0.10	98.01±0.43	98.73±0.73	99.02±0.31	99.41±0.27	**99.76±0.14**	99.57±0.19	99.54±0.21	99.74±0.21
2	57.47±16.41	81.10±17.13	87.65±13.93	78.38±11.67	71.33±36.03	92.24±4.01	89.28±2.28	91.08±2.90	**92.52±4.49**
3	76.94±18.35	82.05±11.31	89.73±6.17	78.34±13.50	91.18±5.76	92.87±4.07	**95.57±1.88**	87.27±0.95	94.49±2.69
4	70.60±19.82	61.86±25.15	71.47±21.75	72.71±12.17	88.05±7.22	92.20±7.23	92.30±2.44	**96.19±3.44**	92.77±7.13
5	36.54±19.24	74.48±15.29	78.01±17.99	68.51±15.70	82.66±6.76	90.99±5.59	89.14±3.15	87.88±2.76	**92.41±5.33**
6	93.01±5.62	84.76±17.90	77.53±34.64	88.04±7.85	**99.24±0.17**	96.99±1.66	95.90±1.16	97.18±1.26	97.14±2.11
7	64.09±13.75	78.22±8.27	73.36±15.85	74.48±14.15	**87.29±3.62**	85.43±2.16	84.33±1.68	86.48±2.08	84.74±1.65
8	72.15±12.83	89.36±11.95	72.19±29.07	95.79±2.23	97.67±0.63	98.38±1.47	97.28 ±1.02	98.36±0.99	**99.04±0.37**
9	98.88±1.93	96.71±13.28	78.50±30.87	**99.73±0.24**	97.04±2.13	98.56±0.87	97.77 ±2.03	95.94±0.53	99.19±0.71
OA	83.40±3.37	90.78±4.77	91.13±13.72	92.88±1.09	95.69±2.03	97.41±0.60	96.69±0.37	97.07±0.40	**97.70±0.36**
AA	74.33±2.56	82.95±6.80	82.80±17.65	83.89±2.66	90.43±4.43	94.16±0.63	93.46±0.38	93.32±0.34	**94.67±0.54**
Kappa	77.10±4.35	87.20±6.43	85.10±18.17	89.98±1.54	93.92±2.87	96.34±0.84	95.43±0.52	96.03±0.56	**96.74±0.51**

a class-covariance metric embedded space. Parameter settings for DCFSL, Gia-CFSL, and CMFSL are consistent with HyMuT.

The classification performance of the aforementioned methods, measured by OA, AA, kappa coefficient, and class-specific accuracy, is summarized in Tables XI–XIV. To facilitate visual comparison, the corresponding classification maps are provided in Figs. 11–14. Based on these results, the following conclusions can be made.

1) In the case of few labeled samples, deep learning models exhibit stronger feature extraction capabilities compared to the classic machine learning model SVM. When measured by the OA index, these models outperform SVM by at least 2.92%, 0.81%, 6.05%, and 7.38% on the four datasets.
2) Few-shot learning models demonstrate excellent classification performance compared to deep learning methods. 3-D CNN and Two-CNN could extract spatial–spectral

TABLE XV

TIME (S), FLOPS (M, MILLION), AND NUMBER OF PARAMETERS (M, MILLION) OF DIFFERENT METHODS

		SVM	3D-CNN	Two-CNN	RN-FSC	DMM	DCFSL	Gia-CFSL	CMFSL	HyMuT
Salinas	TIME(s)	0.02	51.24	1217.85	472.52	187.01	2092.52	2490.59	1422.11	4595.25
	FLOPs(MMac)	-	45.21	15.13	6.25	10.33	47.01	67.82	28.51	98.48
	#params(M)	-	0.60	1.91	0.19	0.14	4.26	0.31	0.65	4.69
Indian Pines	TIME(s)	0.02	27.21	1359.4	309.11	84.24	1936.84	3916.49	1386.91	4548.17
	FLOPs(MMac)	-	44.51	15.13	6.25	10.29	46.98	67.79	28.47	98.42
	#params(M)	-	0.59	1.91	0.19	0.14	4.26	0.31	0.65	4.69
Houston	TIME(s)	0.02	20.95	1245.37	250.6	97.81	1692.37	2435.04	1199.89	4244.35
	FLOPs(MMac)	-	31.22	14.14	6.25	9.70	46.52	67.34	27.99	97.31
	#params(M)	-	0.31	1.81	0.19	0.13	4.26	0.31	0.64	4.68
Pavia	TIME(s)	0.05	20.64	1654.45	180.67	619.06	1469.07	1851.60	791.24	3082.65
	FLOPs(MMac)	-	21.44	13.47	6.25	9.27	46.18	67.01	27.68	96.83
	#params(M)	-	0.28	1.74	0.19	0.12	4.26	0.31	0.64	4.67

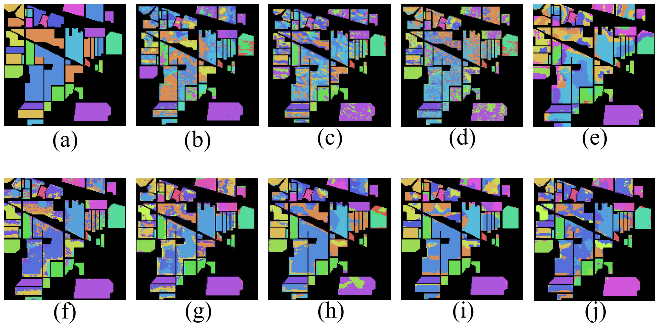


Fig. 11. IP. (a) Ground truth map. (b) SVM (38.72%). (c) 3-D CNN (45.30%). (d) Two-CNN (41.64%). (e) RN-FSC (50.04%). (f) DMM (60.33%). (g) DCFSL (66.10%). (h) Gia-CFSL (67.42%). (i) CMFSL (67.32%). (j) HyMuT (70.57%).

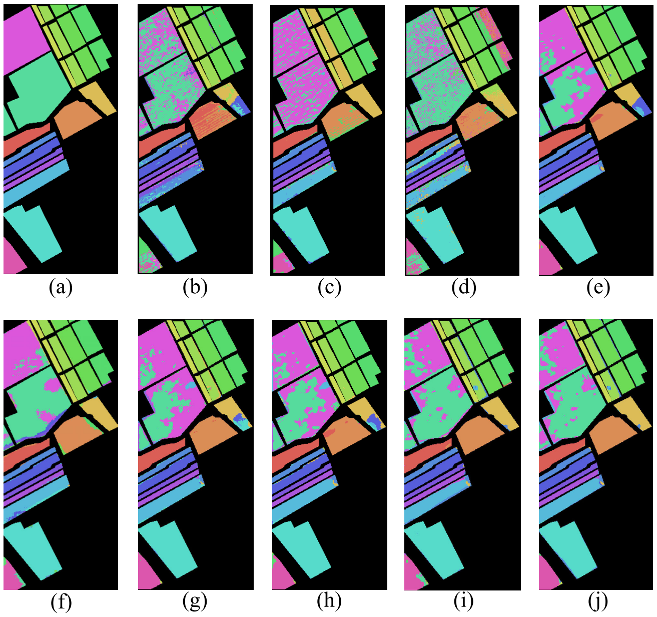


Fig. 12. SA. (a) Ground truth map. (b) SVM (68.92%). (c) 3-D CNN (69.73%). (d) Two-CNN (71.45%). (e) RN-FSC (87.84%). (f) DMM (89.02%). (g) DCFSL (89.23%). (h) Gia-CFSL (89.66%). (i) CMFSL (90.26%). (j) HyMuT (91.11%).

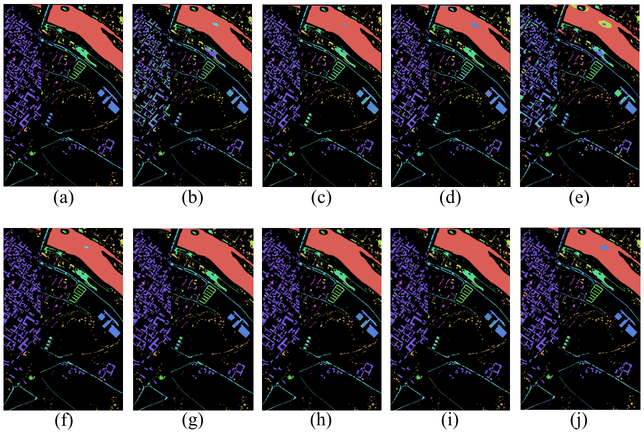


Fig. 13. PC. (a) Ground truth map. (b) SVM (83.40%). (c) 3-D CNN (90.78%). (d) Two-CNN (91.13%). (e) RN-FSC (92.88%). (f) DMM (95.69%). (g) DCFSL (97.41%). (h) Gia-CFSL (96.69%). (i) CMFSL (97.07%). (j) HyMuT (97.70%).

joint features, and on this basis, the few-shot learning methods further utilize the advanced semantic information of classification through relation network, contrastive learning, and prototype network. Few-shot learning-based models outperform previous deep learning models by at least 8.4%, 16.39%, 4.89%, and 1.75% in terms of the OA index.

3) Methods DCFSL, Gia-CFSL, and CMFSL, which fully combine few-shot learning and domain adaptation, exhibit superior performance compared to few-shot learning models alone. By leveraging source domain data, cross-domain few-shot learning methods demonstrate improved performance on target domain tasks. DCFSL achieves classification results that are 5.77%, 0.21%, 5%, and 1.72% higher than DMM in terms of OA. Building upon this, Gia-CFSL and CMFSL models prioritize intraclass similarity and interclass differences, incorporating superior cross-domain techniques and achieving optimal AA metrics for specific datasets.
4) Furthermore, HyMuT combines three auxiliary tasks and the corresponding MTL methods to fully mine redundant information from multiple perspectives. HyMuT achieves the highest OA, AA, and kappa scores, surpassing the previous best model by 3.25%, 0.85%, 0.36%, and 0.29% on the OA index.

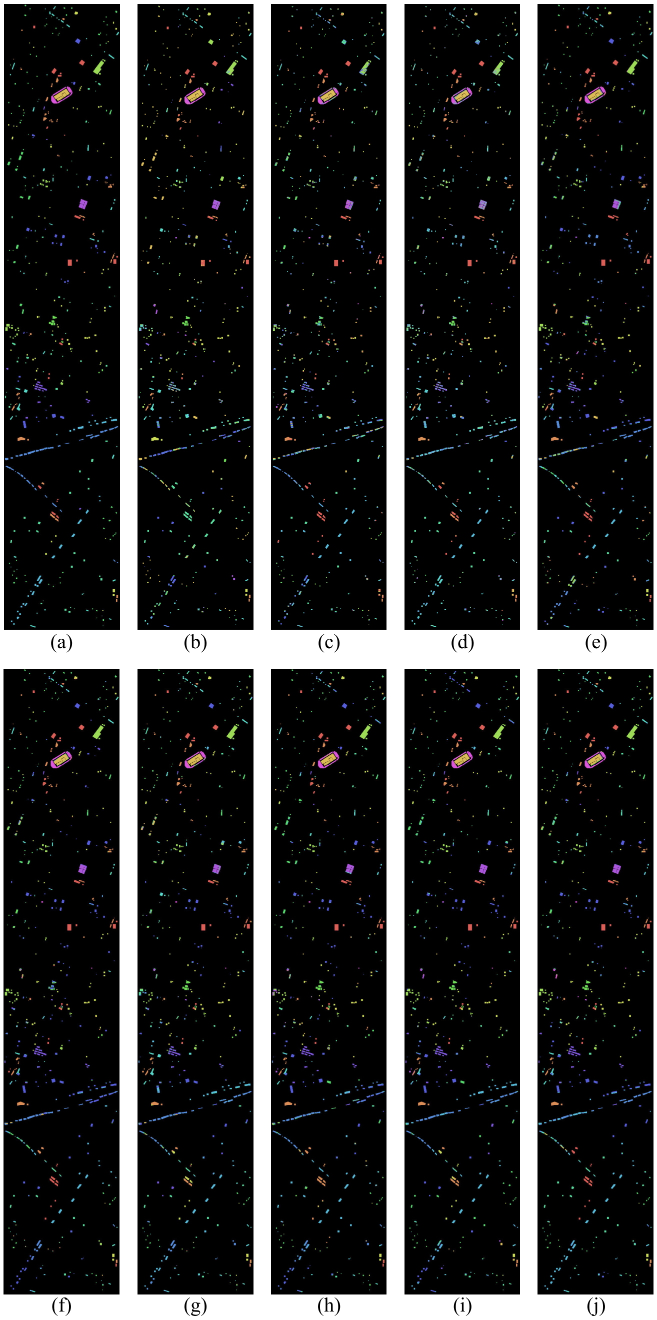


Fig. 14. Houston. (a) Ground truth map. (b) SVM (48.55%). (c) 3-D CNN (54.60%). (d) Two-CNN (56.88%). (e) RN-FSC (61.77%). (f) DMM (70.24%). (g) DCFSL (75.24%). (h) Gia-CFSL (72.47%). (i) CMFSL (78.25%). (j) HyMuT (78.61%).

E. Analysis of the Computational Complexity

Time complexity analysis is a fundamental approach for assessing the computational efficiency of algorithms. In this study, we evaluate the time complexity of our model using metrics such as floating-point operations per second (FLOPs), parameters, and the overall training time (TIME).

Table XV presents the computational complexity of HyMuT compared to other methods. Although HyMuT exhibits higher computational complexity, the indexes remains comparable to other methods. One of the major drawbacks is the high computational complexity of the model, and the increased complexity is primarily attributed to the inclusion of additional network modules in our model, resulting in a higher number of parameters and FLOPs. However, considering the superior performance achieved by HyMuT, the associated computational burden remains acceptable.

V. Conclusion

In this article, we propose HyMuT, a hybrid multitask learning framework, to address the challenge of limited labeled samples in HSI classification. HyMuT incorporates three auxiliary tasks and integrates three MTL methods to effectively transfer and leverage information. Specifically, a hard parameter sharing backbone network is employed to handle the primary target classification task along with a similar spatial mask prediction task. Furthermore, we introduce a weakly parameter sharing autoencoder to address the related spatial–spectral mask reconstruction task and facilitate adaptive feature transfer. Additionally, a weakly correlated domain adaptation task is introduced to enhance task diversity. A dual adversarial classifier is utilized to extract task invariance and domain invariance for improved performance. Extensive experiments conducted on four benchmark datasets demonstrate the superiority of HyMuT over other classical models. Ablation studies further validate the effectiveness of different components.

On the other hand, the model has some shortcomings that need to be noted. The main issue is how to select appropriate auxiliary tasks for the original target task without causing damage to the model. Besides, the autoencoder constructed through soft parameter sharing results in unsatisfactory model parameters and TIME, and further research is needed to optimize this deficiency. HyMuT serves as a fundamental framework for MTL without the need for excessive amounts of datasets. Potential research directions will focus on refining the self-supervised strategy-based auxiliary tasks and enhancing the optimization of MTL. Additionally, we will explore more concise and elegant approaches to multitask coupling.

References

[1] S. Meng, X. Wang, X. Hu, C. Luo, and Y. Zhong, "Deep learning-based crop mapping in the cloudy season using one-shot hyperspectral satellite imagery," *Comput. Electron. Agricult.*, vol. 186, Jul. 2021, Art. no. 106188.

[2] B. Lu, P. Dao, J. Liu, Y. He, and J. Shang, "Recent advances of hyperspectral imaging technology and applications in agriculture," *Remote Sens.*, vol. 12, no. 16, p. 2659, Aug. 2020.

[3] T. Adão et al., "Hyperspectral imaging: A review on UAV-based sensors, data processing and applications for agriculture and forestry," *Remote Sens.*, vol. 9, no. 11, p. 1110, Oct. 2017.

[4] F. Xue, F. Tan, Z. Ye, J. Chen, and Y. Wei, "Spectral–spatial classification of hyperspectral image using improved functional principal component analysis," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.

[5] J. Huang, K. Liu, M. Xu, M. Perc, and X. Li, "Background purification framework with extended morphological attribute profile for hyperspectral anomaly detection," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 8113–8124, 2021.

[6] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machines," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 8, pp. 1778–1790, Aug. 2004.

[7] L. Samaniego, A. Bardossy, and K. Schulz, "Supervised classification of remotely sensed imagery using a modified k-NN technique," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 7, pp. 2112–2125, Jul. 2008.

[8] L. Zhang and L. Zhang, "Artificial intelligence for remote sensing data analysis: A review of challenges and opportunities," *IEEE Geosci. Remote Sens. Mag.*, vol. 10, no. 2, pp. 270–294, Jun. 2022.

[9] J. Wang, W. Li, M. Zhang, R. Tao, and J. Chanussot, "Remote-sensing scene classification via multistage self-guided separation network," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5615312.

[10] M. Zhang, W. Li, X. Zhao, H. Liu, R. Tao, and Q. Du, "Morphological transformation and spatial-logical aggregation for tree species classification using hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5501212.

[11] L. Fang, Y. Jiang, Y. Yan, J. Yue, and Y. Deng, "Hyperspectral image instance segmentation using spectral–spatial feature pyramid network," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5502613.

[12] H. Lee and H. Kwon, "Going deeper with contextual CNN for hyperspectral image classification," *IEEE Trans. Image Process.*, vol. 26, no. 10, pp. 4843–4855, Oct. 2017.

[13] J. Xi et al., "Wide sliding window and subsampling network for hyperspectral image classification," *Remote Sens.*, vol. 13, no. 7, p. 1290, Mar. 2021.

[14] C. Yu, R. Han, M. Song, C. Liu, and C.-I. Chang, "Feedback attention-based dense CNN for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5501916.

[15] X. Cao, J. Yao, Z. Xu, and D. Meng, "Hyperspectral image classification with convolutional neural network and active learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 7, pp. 4604–4616, Jul. 2020.

[16] J. Li, J. M. Bioucas-Dias, and A. Plaza, "Semisupervised hyperspectral image classification using soft sparse multinomial logistic regression," *IEEE Geosci. Remote Sens. Lett.*, vol. 10, no. 2, pp. 318–322, Mar. 2013.

[17] X. Zhang, Q. Song, R. Liu, W. Wang, and L. Jiao, "Modified co-training with spectral and spatial views for semisupervised hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2044–2055, Jun. 2014.

[18] B. Liu, X. Yu, P. Zhang, X. Tan, A. Yu, and Z. Xue, "A semi-supervised convolutional neural network for hyperspectral image classification," *Remote Sens. Lett.*, vol. 8, no. 9, pp. 839–848, May 2017.

[19] J. Peng, Y. Huang, W. Sun, N. Chen, Y. Ning, and Q. Du, "Domain adaptation in remote sensing image classification: A survey," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 9842–9859, 2022.

[20] Y. Zhang, W. Li, M. Zhang, Y. Qu, R. Tao, and H. Qi, "Topological structure and semantic information transfer network for cross-scene hyperspectral image classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 6, pp. 2817–2830, Sep. 2021.

[21] Y. Zhang, W. Li, R. Tao, J. Peng, Q. Du, and Z. Cai, "Cross-scene hyperspectral image classification with discriminative cooperative alignment," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 11, pp. 9646–9660, Nov. 2021.

[22] L. Zhang, M. Lan, J. Zhang, and D. Tao, "Stagewise unsupervised domain adaptation with adversarial self-training for road segmentation of remote-sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5609413.

[23] Z. Cao, X. Li, J. Jianfeng, and L. Zhao, "3D convolutional Siamese network for few-shot hyperspectral classification," *J. Appl. Remote Sens.*, vol. 14, no. 4, Nov. 2020, Art. no. 048504.

[24] C. Zhang, J. Yue, and Q. Qin, "Global prototypical network for few-shot hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 4748–4759, 2020.

[25] D. Pal, V. Bundele, B. Banerjee, and Y. Jeppu, "SPN: Stable prototypical network for few-shot learning-based hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.

[26] K. Gao, B. Liu, X. Yu, J. Qin, P. Zhang, and X. Tan, "Deep relation network for hyperspectral image few-shot classification," *Remote Sens.*, vol. 12, no. 6, p. 923, Mar. 2020.

[27] M. Zhang, W. Li, Y. Zhang, R. Tao, and Q. Du, "Hyperspectral and LiDAR data classification based on structural optimization transmission," *IEEE Trans. Cybern.*, vol. 53, no. 5, pp. 3153–3164, May 2023.

[28] J. Wang, W. Li, Y. Wang, R. Tao, and Q. Du, "Representation-enhanced status replay network for multisource remote-sensing image classification," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Jun. 28, 2023, doi: 10.1109/TNNLS.2023.3286422.

[29] M. Zhang, X. Zhao, W. Li, Y. Zhang, R. Tao, and Q. Du, "Cross-scene joint classification of multisource data with multilevel domain adaption network," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Apr. 6, 2023, doi: 10.1109/TNNLS.2023.3262599.

[30] M. Zhang, X. Zhao, W. Li, and Y. Zhang, "Multi-source remote sensing data cross scene classification based on multi-graph matching," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2022, pp. 827–830.

[31] Z. Li, M. Liu, Y. Chen, Y. Xu, W. Li, and Q. Du, "Deep cross-domain few-shot learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5501618.

[32] B. Xi, J. Li, Y. Li, R. Song, D. Hong, and J. Chanussot, "Few-shot learning with class-covariance metric for hyperspectral image classification," *IEEE Trans. Image Process.*, vol. 31, pp. 5079–5092, 2022.

[33] Y. Zhang, W. Li, M. Zhang, S. Wang, R. Tao, and Q. Du, "Graph information aggregation cross-domain few-shot learning for hyperspectral image classification," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Jun. 30, 2022, doi: 10.1109/TNNLS.2022.3185795.

[34] M. Zhang, H. Liu, M. Gong, H. Li, Y. Wu, and X. Jiang, "Cross-domain self-taught network for few-shot hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 4501719.

[35] Y. Zhang and Q. Yang, "A survey on multi-task learning," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 12, pp. 5586–5609, Dec. 2022.

[36] M. Long, Z. Cao, J. Wang, and P. S. Yu, "Learning multiple tasks with multilinear relationship networks," in *Proc. Adv. Neural. Inf. Process. Syst.*, vol. 30, Dec. 2017, pp. 1594–1603.

[37] Y. Lu, A. Kumar, S. Zhai, Y. Cheng, T. Javidi, and R. Feris, "Fully-adaptive feature sharing in multi-task networks with applications in person attribute classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5334–5343.

[38] I. Misra, A. Shrivastava, A. Gupta, and M. Hebert, "Cross-stitch networks for multi-task learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3994–4003.

[39] R. Cipolla, Y. Gal, and A. Kendall, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 7482–7491.

[40] Y. Yang and T. Hospedales, "Deep multi-task representation learning: A tensor factorisation approach," 2016, *arXiv:1605.06391*.

[41] L. Mou, P. Ghamisi, and X. X. Zhu, "Unsupervised spectral–spatial feature learning via deep residual conv–deconv network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 1, pp. 391–406, Jan. 2018.

[42] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 16000–16009.

[43] S. Singh et al., "Self-supervised feature learning for semantic segmentation of overhead imagery," in *Proc. BMVC*, Sep. 2018, vol. 1, no. 2, p. 4.

[44] S. Vincenzi et al., "The color out of space: Learning self-supervised representations for Earth observation imagery," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Jan. 2021, pp. 3034–3041.

[45] L. Zhao, W. Luo, Q. Liao, S. Chen, and J. Wu, "Hyperspectral image classification with contrastive self-supervised learning under limited labeled samples," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.

[46] P. Guan and E. Y. Lam, "Cross-domain contrastive learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5528913.

[47] H. Guo, Q. Shi, B. Du, L. Zhang, D. Wang, and H. Ding, "Scene-driven multitask parallel attention network for building extraction in high-resolution remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 5, pp. 4287–4306, May 2021.

[48] W. Li, H. Chen, and Z. Shi, "Semantic segmentation of remote sensing images with self-supervised multitask representation learning," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 6438–6450, 2021.

[49] Y. Sun, X. Zhang, J. Huang, H. Wang, and Q. Xin, "Fine-grained building change detection from very high-spatial-resolution remote sensing images based on deep multitask learning," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.

[50] S. Liu, Q. Shi, and L. Zhang, "Few-shot hyperspectral image classification with unknown classes using multitask deep learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 6, pp. 5085–5102, Jun. 2021.

[51] R. Hang, F. Zhou, Q. Liu, and P. Ghamisi, "Classification of hyperspectral images via multitask generative adversarial networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 2, pp. 1424–1436, Feb. 2021.

[52] L. Song, Z. Feng, S. Yang, X. Zhang, and L. Jiao, "Self-supervised assisted semi-supervised residual network for hyperspectral image classification," *Remote Sens.*, vol. 14, no. 13, p. 2997, Jun. 2022.

[53] B. Tu, X. Liao, C. Zhou, S. Chen, and W. He, "Feature extraction using multitask superpixel auxiliary learning for hyperspectral classification," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–16, 2021.

[54] K. Li, D. Dai, and L. Van Gool, "Hyperspectral image super-resolution with RGB image super-resolution as an auxiliary task," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2022, pp. 3193–3202.

[55] N. Li, D. Zhou, J. Shi, X. Zheng, T. Wu, and Z. Yang, "Graph-based deep multitask few-shot learning for hyperspectral image classification," *Remote Sens.*, vol. 14, no. 9, p. 2246, May 2022.

[56] Y. Ganin et al., "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, no. 59, pp. 1–35, 2016.

[57] Y. Yuan, J. Lin, and Q. Wang, "Hyperspectral image classification via multitask joint sparse representation and stepwise MRF optimization," *IEEE Trans. Cybern.*, vol. 46, no. 12, pp. 2966–2977, Dec. 2016.

[58] M. Ye, Y. Qian, and J. Zhou, "Multitask sparse nonnegative matrix factorization for joint spectral–spatial hyperspectral imagery denoising," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 5, pp. 2621–2639, May 2015.

[59] Z. He, Q. Wang, Y. Shen, and M. Sun, "Kernel sparse multitask learning for hyperspectral image classification with empirical mode decomposition and morphological wavelet-based features," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 8, pp. 5150–5163, Aug. 2014.

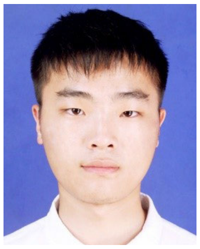
[60] B. Liu, X. Yu, A. Yu, P. Zhang, G. Wan, and R. Wang, "Deep few-shot learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 4, pp. 2290–2304, Apr. 2019.

[61] R. Caruana, "Multitask learning: A knowledge-based source of inductive bias1," in *Proc. 10th Int. Conf. Mach. Learn.*, 1993, pp. 41–48.

[62] M. He, B. Li, and H. Chen, "Multi-scale 3D deep convolutional neural network for hyperspectral image classification," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 3904–3908.

[63] B. Deng, S. Jia, and D. Shi, "Deep metric learning-based feature embedding for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 2, pp. 1422–1435, Feb. 2020.

[64] J. Yang, Y.-Q. Zhao, and J. C. Chan, "Learning and transferring deep joint spectral–spatial features for hyperspectral classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 8, pp. 4729–4742, Aug. 2017.

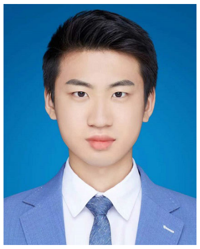


Hao Liu received the B.Eng. degree in electronic information engineering from Xidian University, Xi'an, China, in 2021, where he is currently pursuing the M.Eng. degree in electronic information with the School of Electronic Engineering.

His research interests include multitask learning, few-shot learning, and remote sensing image processing

Mingyang Zhang (Member, IEEE) received the B.S. degree in automation and the Ph.D. degree in pattern recognition and intelligent systems from Xidian University, Xi'an, China, in 2012 and 2018, respectively.

Since 2018, he has been a Lecturer with the School of Electronic Engineering, Xidian University. His research interests include computational intelligence and remote sensing image understanding.



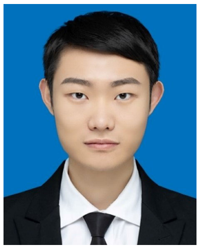
Ziqi Di received the B.Eng. degree in electronic engineering from Xidian University, Xi'an, China, in 2021, where he is currently pursuing the Ph.D. degree in electronic science and technology with the School of Electronic Engineering.

His research interests include remote sensing image interpretation and Bayesian deep learning.

Maoguo Gong (Fellow, IEEE) received the B.S. degree in electronic engineering and the Ph.D. degree in electronic science and technology from Xidian University, Xi'an, China, in 2003 and 2009, respectively.

Since 2006, he has been a Teacher with Xidian University, where he was promoted as an Associate Professor and as a Full Professor in 2008 and 2010, respectively, with exceptive admission. His research interests are in the area of computational intelligence with applications to optimization, learning, data mining, and image understanding.

Prof. Gong is an Executive Committee Member of the Chinese Association for Artificial Intelligence and a Senior Member of the Chinese Computer Federation. He was a recipient of the Prestigious National Program for Support of the Leading Innovative Talents from the Central Organization Department of China, the Leading Innovative Talent in Science and Technology from the Ministry of Science and Technology of China, the Excellent Young Scientist Foundation from the National Natural Science Foundation of China, the New Century Excellent Talent from the Ministry of Education of China, and the National Natural Science Award of China. He is an Associate Editor or an Editorial Board Member for over five journals including the IEEE Transactions on Evolutionary Computation and the IEEE Transactions on Neural Networks and Learning Systems.



Tianqi Gao (Student Member, IEEE) received the B.S. degree in electronic engineering from Xidian University, Xi'an, China, in 2019, where he is currently pursuing the Ph.D. degree in electronic science and technology with the School of Electronic Engineering.

His research interests include deep learning, evolutionary computation, remote sensing image interpretation, and computational intelligence.

A. K. Qin (Senior Member, IEEE) received the B.Eng. degree in automatic control from Southeast University, Nanjing, China, in 2001, and the Ph.D. degree in computer science and engineering from Nanyang Technology University, Singapore, in 2007.

From 2007 to 2017, he was with the University of Waterloo, Waterloo, ON, Canada, INRIA Grenoble Rhône-Alpes, Montbonnot-Saint-Martin, France, and RMIT University, Melbourne, VIC, Australia. He joined the Swinburne University of Technology, Hawthorn, VIC, Australia, in 2017, where he is currently a Professor, the Director of Swinburne Intelligent Data Analytics Laboratory, and the Deputy Director of Swinburne Space Technology and Industry Institute. His major research interests include machine learning, evolutionary computation, computer vision, remote sensing, services computing, and edge computing.

Dr. Qin was a recipient of the 2012 IEEE Transactions on Evolutionary Computation Outstanding Paper Award. He is currently the Vice-Chair of the IEEE Computational Intelligence Society (CIS) Neural Networks Technical Committee, the Vice-Chair of the IEEE CIS Emergent Technologies Task Force on "Multitask Learning and Multitask Optimization," the Vice-Chair of the IEEE CIS Neural Networks Task Force on "Deep Edge Intelligence," and the Chair of the IEEE CIS Neural Networks Task Force on "Deep Vision in Space."