

Federated feature distillation for Non-IID remote sensing scene classification

JIN Jing, QIN Weibo, LI Zifei, and WANG Feng*

Key Laboratory for Information Science of Electromagnetic Waves (Ministry of Education),
College of Future Information Technology, Fudan University, Shanghai 200433, China

Abstract: The rapid growth in satellite and aerial remote sensing platforms has created a growing need for distributed remote sensing scene classification. Conventional centralized scene classification methods, which involve transmitting remote sensing data to a ground station for processing, encounter limitations in both transmission efficiency and data privacy. Federated learning (FL) has emerged as a promising approach by enabling terminals to collaboratively train models without exchanging raw data. However, the non-independent and identically distributed (Non-IID) nature of remote sensing data significantly impedes FL performance. To address these challenges, a federated framework with feature distillation (FD) (FedFD) is proposed for FL-based remote sensing scene classification. Specifically, FedFD facilitates collaborative training by aggregating model parameters from multiple terminals to the cloud, thereby optimizing a global model. To further alleviate the impact of Non-IID data, an innovative partial feature-sharing strategy based on FD is designed, which divides features into globally shared essential features and locally maintained supplementary features. Moreover, to cope with object and scene scale variation, the squeeze and excitation module and the pyramid pooling module are incorporated into the scene classification network to enhance multiscale feature extraction. Extensive experiments on the Northwestern Polytechnical University Remote Sensing Image Scene Classification 45 (NWPU-RESISC45) dataset and University of California, Merced Land Use (UC-Merced) dataset, under varying numbers of terminals and Non-IID levels, validate the effectiveness and scalability of FedFD, and demonstrate its superior performance in FL-based remote sensing scene classification.

Keywords: remote sensing, scene classification, federated learning (FL), non-independent and identically distributed (Non-IID), feature distillation.

DOI: 10.23919/JSEE.2026.000026

Manuscript received September 16, 2025.

*Corresponding author.

This work was supported by Shanghai Collaborative Innovation Program (XTCX-KJE001-2025-08).

1. Introduction

Remote sensing (RS) scene classification (SC), as a technique for analyzing semantic content and target properties to categorize RS images into appropriate scene categories, has broad applications in earth observation, including environmental monitoring, urban management, and disaster assessment [1–4]. Recent advancements in geoscience and earth observation technologies have led to substantial improvements in RS SC. The rapid increase of satellite and aerial platforms leads to an exponential growth of RS data, creating an urgent demand for accurate and efficient interpretation methods.

Deep learning (DL) is widely recognized as an effective method for RS SC due to its remarkable feature extraction capability [5]. However, most existing methods [1–10] typically rely on centralized architectures, in which RS data from terminals is downloaded to a ground station for processing, and the model is trained on the collected data. Such centralized training introduces several limitations. Firstly, transferring massive RS data from satellite platforms to the ground station is inefficient due to limited communication bandwidth [11]. Secondly, satellite-to-ground communication links commonly suffer from latency, complicating real-time RS data processing [12]. Additionally, transmitting raw RS data raises privacy concerns [13], as RS data frequently contains sensitive geographic and infrastructural information, potentially violating privacy regulations such as the general data protection regulation (GDPR) [14].

To cope with these problems, federated learning (FL) has emerged as a promising paradigm for distributed training of RS SC. It is a privacy-preserving paradigm that enables multiple terminals to collaboratively train a model by exchanging model parameters instead of raw data [15]. In the FL-based setup, each terminal performs local training and uploads model parameters to the cloud

for global model updating, thus avoiding raw RS data transmission. Recently, FL has gained increasing attention in RS, with applications such as object extraction and landslide detection [16,17]. Despite these advancements, the application of FL to RS SC remains largely underexplored, leaving a research gap in its potential use.

In addition, DL methods are typically based on the assumption that training data are independent and identically distributed (IID), which is essential for ensuring model convergence and generalization [18,19]. However, this assumption is often violated in RS due to imbalanced label distributions, varying data quantities, and domain shifts induced by geographic and imaging diversity across terminals [20,21]. These characteristics result in inherently non-independent and identically distributed (Non-IID) RS data across terminals. Such data heterogeneity poses significant challenges to FL, as local model updates may deviate from the global optimum, hindering both convergence and generalization [22–25]. Moreover, the lack of access to raw data during model aggregation further exacerbates these issues, ultimately degrading global model performance [26].

To address Non-IID challenges, prior work has explored various approaches, including the formulation of learning objectives [22,23] and the design of aggregation strategies [24,25]. For instance, Li et al. [23] introduced a proximal term to prevent local updates from diverging significantly from the global model, thereby stabilizing the training process. Wang et al. [25] normalized client updates to reduce variations in local updates, thereby improving convergence under heterogeneous data distributions. Nevertheless, these approaches still exhibit limitations in scalability and sensitivity to severe data heterogeneity, motivating further research. More recently, inter-terminal information sharing has emerged as a simple yet effective approach, as it improves model consistency while mitigating data heterogeneity without exposing raw data [27].

Inspired by this, a federated framework with feature distillation (FedFD) is proposed for FL-based RS SC. The training process comprises four stages: feature distillation (FD), feature sharing, local training, and model updating. To address Non-IID issues, FedFD adopts a partial feature-sharing (PFS) strategy based on FD. Specifically, the extracted features are divided into two categories: essential features, which significantly contribute to model performance, and supplementary features with limited performance impact. Essential features are globally shared across terminals, while supplementary features are retained locally. This strategy facilitates

collaborative training on both locally retained and globally shared data. Furthermore, to effectively handle the multiscale features present in RS images, caused by differences in imaging distance and intrinsic properties of objects and scenes, the squeeze and excitation module (SEM) and pyramid pooling module (PPM) are integrated into the SC network. This effectively extracts multiscale features, further enhancing classification performance. Extensive experiments on Northwestern Polytechnical University Remote Sensing Image Scene Classification 45 (NWPU-RESISC45) and University of California, Merced Land Use (UC-Merced) datasets under varying terminal numbers and Non-IID levels validate the effectiveness and scalability of FedFD, demonstrating its superior performance in FL-based RS SC.

To sum up, the main contributions of this paper are summarized as follows:

- (i) FedFD, a distributed RS SC framework, is proposed to enable collaborative training by aggregating model parameters from terminals to the cloud without sharing raw data, optimizing an effective global model.
- (ii) A PFS strategy based on FD is designed to mitigate Non-IID issues, where each terminal extracts essential features for global sharing and retains supplementary features locally, thus reducing data heterogeneity and facilitating collaborative training.
- (iii) The SEM and PPM are incorporated into the SC network to address scale variations in RS images, thereby improving multiscale feature extraction.

The remainder of this paper is structured as follows. Section 2 reviews related work. Section 3 presents the FedFD framework in detail. Section 4 describes the datasets and experimental settings, and provides extensive comparative and ablation analyses. Section 5 concludes the paper.

2. Related work

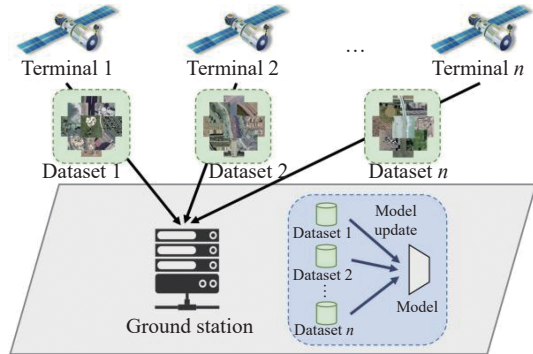
2.1 RS SC with DL

As a fundamental yet challenging task, SC aims to extract high-level semantic information from RS images to categorize them into corresponding scene types, supporting applications like land monitoring and urban planning [28–31]. Recent advances in DL have significantly enhanced SC performance. For instance, Hu et al. [32] employed pre-trained convolutional neural networks (CNNs) as feature extractors for SC tasks. Wang et al. [6] enhanced SC performance by proposing the attention recurrent convolutional network (ARCNet), which incorporates attention mechanisms.

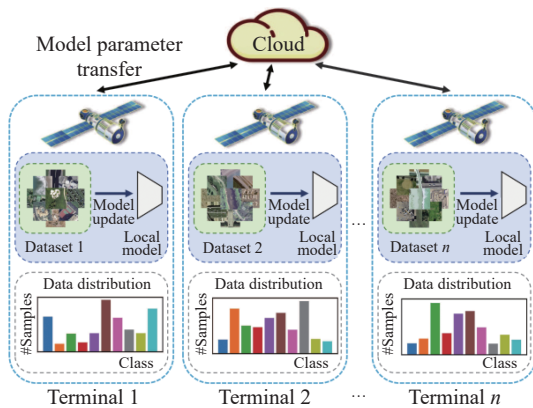
Despite these achievements, scale variation remains a

critical challenge in RS SC. Objects and scenes in RS images often appear at different spatial scales due to varying imaging distances and intrinsic scene properties. Although various approaches such as skip connections [9], adaptive receptive fields [10], and deep layer aggregation [33] have been proposed, existing methods are still insufficient for effective multiscale representation. For example, although deeper CNNs can extract high-level semantic features, they may lose part of the fine-grained details during hierarchical abstraction, which challenges accurate multiscale representation.

Moreover, most existing DL-based SC methods [1–10] typically rely on centralized architectures, which require the transmission of RS data to a ground station for processing and training, as illustrated in Fig. 1(a). This centralized paradigm faces several limitations, including high transmission costs due to limited bandwidth [11], latency in satellite-to-ground communication [12], and privacy risks arising from the exposure of sensitive geographic data [13,14]. With the proliferation of satellite and aerial platforms, the volume of RS data continues to grow exponentially, further intensifying the need for distributed and efficient SC methods.



(a) Centralized RS SC



(b) FL-based distributed RS SC

Fig. 1 Comparison between conventional centralized and FL-based distributed RS SC methods

2.2 FL

FL has emerged as an effective approach for distributed DL, enabling collaborative model training across multiple terminals without sharing raw data [15]. As illustrated in Fig. 1(b), a typical FL system consists of a cloud server and N terminals. The training process generally follows three iterative steps: (i) the cloud server initializes and distributes a global model to all participating terminals; (ii) each terminal trains the model locally and sends the updated model parameters to the cloud server; (iii) the cloud server aggregates the parameters and broadcasts the updated global model back to all terminals. This process repeats until the global model converges or the maximum number of iterations is reached [34].

Depending on how data is distributed across the sample and feature spaces, FL can be categorized into three types: horizontal FL (HFL), vertical FL (VFL), and federated transfer learning (FTL) [13]. HFL applies when distributed datasets share similar feature spaces but minimal sample overlap. VFL suits scenarios with substantial sample overlap, yet different feature spaces. FTL addresses cases with minimal overlap in both sample and feature spaces by transferring knowledge from a source to a target domain. Among these categories, HFL has received the most practical attention, as it aligns with common FL scenarios where terminals share similar feature spaces while holding different local datasets. Therefore, most foundational FL algorithms have been developed within the HFL paradigm.

Building on the HFL paradigm, a range of representative algorithms have been developed to enhance the effectiveness and generalization of FL. Among them, federated averaging (FedAvg) [15] remains one of the most foundational and widely adopted FL algorithms, leveraging average aggregation of local model parameters to achieve a balance between scalability and performance. To address the Non-IID challenge, FedProx [23] introduces a proximal term to reduce update divergence across terminals, while federated normalized averaging (FedNova) [25] normalizes local updates to improve global convergence under heterogeneous data distributions. Beyond algorithmic advancements, FL has also found broad applications in healthcare [35], finance [36], communication systems [37,38], autonomous driving [39], and more recently in RS, where its potential is beginning to be recognized.

2.3 FL applications in RS

With the rapid deployment of satellite and aerial platforms, distributed RS interpretation has emerged as a promising paradigm, driving growing interest in FL for

RS. Several recent studies have investigated its feasibility and effectiveness in this domain. For instance, Wang et al. [11] proposed a distributed collaborative method (DCM), which integrates a sample proximity network and a grouped-concatenation module to enhance cross-terminal feature fusion for RS image classification under a cloud-terminal collaborative setting. To support personalized adaptation, Li et al. [12] introduced multiply personalized collaborative learning (MPCL), a framework that allows participants to train models tailored to their local RS tasks, including classification and segmentation with synthetic aperture radar (SAR), hyperspectral, and optical data. Zhang et al. [16] developed a federated learning scheme with prototype matching (FedPM), a scheme designed to address label scarcity and representation consistency in object extraction from very-high-resolution RS images. Tang et al. [17] further proposed federated learning-based landslide detection (FedLD), a HFL framework for collaborative landslide detection across regional institutions.

These studies demonstrate the potential of FL for various RS applications such as multiparty personalized learning, object extraction, and landslide detection. However, their problem formulations differ fundamentally from federated RS SC, which is the focus of this paper. More specifically, they do not explicitly address feature-level privacy preservation or feature alignment for multi-class scene-level representation learning.

Despite these efforts, the application of FL to RS SC remains relatively limited, revealing a research gap that deserves further exploration. Moreover, the inherent heterogeneity of Non-IID RS data poses significant challenges to effective FL training [16,40,41], hindering both convergence and generalization in distributed RS systems [26]. Fig. 2 visualizes Non-IID data distributions generated via latent Dirichlet allocation (LDA) under different α values. Each subplot shows the sample distribution across terminals, where the x-axis indicates the number of samples, the y-axis denotes terminal IDs, and colors represent different classes. As α decreases, the distribution becomes increasingly skewed, with $\alpha \rightarrow 0$ indicating the most imbalanced case. In contrast, larger α values yield more uniform distributions, approaching the IID condition as $\alpha \rightarrow \infty$. Aggregating models trained under these skewed conditions, without access to raw data, amplifies local biases and significantly degrades overall performance. Thus, it is of significance to develop scalable and effective methods for FL-based Non-IID RS SC.

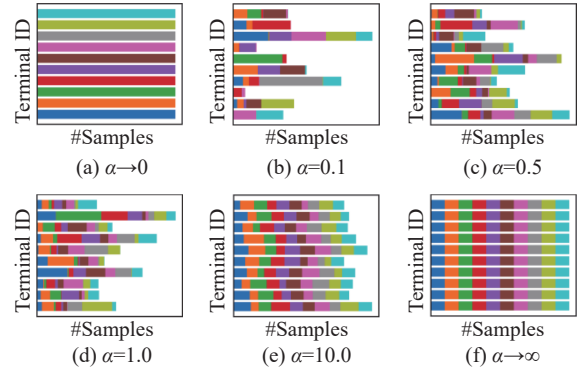


Fig. 2 Visualization of Non-IID data distribution across terminals via LDA for different α values

3. Methodology

3.1 Problem statement

FedFD adopts a terminal-cloud architecture, enabling multiple terminals to train a global model collaboratively in a distributed manner. Specifically, the framework consists of K terminals, denoted as $C = \{C_1, C_2, \dots, C_K\}$, and a cloud server S . Each terminal C_k maintains a local dataset D_k , represented as

$$D_k = \{(\mathbf{x}_{i,k}, \mathbf{y}_{i,k}) \mid i = 1, 2, \dots, |D_k|\} \quad (1)$$

where $|D_k|$ denotes the size of the local dataset, $\mathbf{x}_{i,k}$ is the input data, and $\mathbf{y}_{i,k}$ is the corresponding label. The global dataset across all terminals is defined as $D = \{D_1, D_2, \dots, D_K\}$.

The global objective of FedFD is to minimize the weighted sum of local objectives from all participating terminals:

$$\min_{\Theta} L(\Theta) = \sum_{k=1}^K \omega_k L_k(\Theta_k) \quad (2)$$

where Θ represents the global model parameters, $L_k(\Theta_k)$ denotes the local objective function of terminal C_k , and ω_k is the corresponding aggregation weight, defined as

$$\omega_k = \frac{|D_k|}{\sum_{k=1}^K |D_k|}. \quad (3)$$

The local objective function for terminal C_k over its data distribution P_k is defined as

$$L_k(\Theta_k) \triangleq \mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim P_k} l(\Theta_k; \mathbf{x}, \mathbf{y}) \quad (4)$$

where $l(\cdot)$ denotes the cross-entropy loss function:

$$l(\Theta; \mathbf{x}, \mathbf{y}) = - \sum_{j=1}^N y_j \ln \hat{y}_j \quad (5)$$

with y_j representing the true label and $\hat{y}_j$ the predicted probability for class j . Here, N is the number of classes, and $(\mathbf{x}, \mathbf{y})$ is a data pair sampled from P_k .

In each communication round t , each terminal C_k updates its local model Θ_k^t by minimizing the local loss $L_k(\Theta_k^t)$. The updated local models from all terminals $\{\Theta_k^t\}_{k=1}^K$ are then transmitted to the cloud server, which aggregates them with a predefined aggregation mechanism $M(\cdot)$ to update the global model:

$$\Theta^{t+1} = M(\{\Theta_k^t\}_{k=1}^K). \quad (6)$$

The global model Θ^{t+1} is subsequently broadcast to all

terminals for the next round. This process repeats until the maximum number of communication rounds is reached.

3.2 Overview of the FedFD framework

FedFD leverages an innovative PFS strategy inspired by the information bottleneck (IB) to address Non-IID challenges in FL-based RS SC, selectively sharing essential information across terminals while retaining supplementary information locally. As shown in Fig. 3, the FedFD framework operates in four main steps: FD, feature sharing, local training, and model updating.

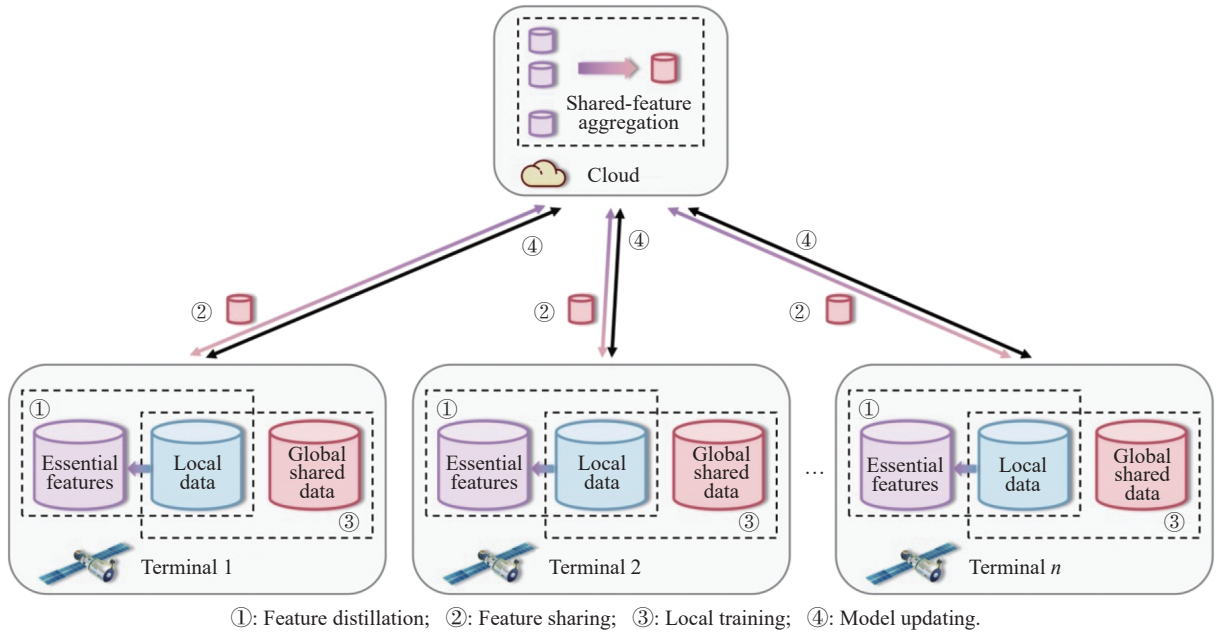


Fig. 3 Illustration of the proposed FedFD framework for FL-based Non-IID RS SC

To begin with, each terminal C_k performs FD to extract essential $\mathbf{x}_{\text{ess}}$ and supplementary $\mathbf{x}_{\text{sup}}$ features from its input $\mathbf{x}$. Subsequently, the essential features $\mathbf{x}_{\text{ess}}$ are transmitted to the cloud server S , where they are aggregated into a global shared dataset D_s and distributed to all terminals. Each terminal then performs local training using both its local dataset D_k and the shared dataset D_s , and uploads its updated model Θ_k^t to the cloud server S . The server aggregates the received local models via an aggregation mechanism $M(\cdot)$, such as weighted averaging in FedAvg, to update the global model Θ^{t+1} , which is then broadcast to all terminals. The local training and model updating steps are repeated for T communication rounds. Detailed implementation is provided in Algorithm 1.

Algorithm 1 FedFD Framework for FL-based Non-IID RS SC

Input Communication rounds T , local datasets $\{D_k\}_{k=1}^K$, aggregation mechanism $M(\cdot)$

Output Global model parameters Θ^T

- 1: **Initialization:** The cloud server S initializes the global model parameters Θ^0 and distributes them to all terminals
 - 2: # FD pre-training
 - 3: Each terminal performs FD
 - 4: # Feature sharing based on the distilled features
 - 5: The cloud server S aggregates essential features $\{\mathbf{x}_{\text{ess},k}\}_{k=1}^K$ to construct the global shared dataset D_s
 - 6: The cloud server S distributes D_s to all terminals C
-

```

7: for each communication round  $t = 1, 2, \dots, T$  do:
8:   # Local training
9:   for each terminal  $C_k$  do:
10:    Train the local model using both the local dataset
        $D_k$  and the global shared dataset  $D_s$ 
11:    Update local parameters  $\Theta_k^t$  using SGD
12:   end for
13:   # Model updating
14:   The cloud server  $S$  aggregates local model parameters
        $\{\Theta_k^t\}_{k=1}^K$  and updates the global model using (6)
15:   The cloud server  $S$  distributes the updated global
       model  $\Theta^{t+1}$  to all terminals  $C$ 
16: end for
17: Return Optimized global model parameters  $\Theta^T$ 

```

It should be noted that the FD step illustrated in Fig. 3 is implemented as a federated pre-training stage rather than being repeated in every main communication round. After this pre-training stage converges, the resulting global FD model is fixed and subsequently used to extract essential features and construct the shared dataset D_s . The following T communication rounds in Algorithm 1 then perform the main federated optimization for SC using D_k and D_s , without further updating the FD module. This design ensures that the FD stage provides a unified, privacy-preserving feature space to facilitate Non-IID feature alignment across terminals.

3.3 PFS strategy

3.3.1 Theoretical basis

To mitigate Non-IID issues, FedFD introduces a PFS strategy that employs FD to extract and selectively transmit the most informative features from input data. The IB principle provides an information-theoretic foundation for FD, aiming to extract a compact latent representation Z of input X that preserves crucial information for predicting the output label Y , while minimizing redundant information. The IB objective is formally defined as

$$\begin{aligned}
 L_{\text{IB}} &= I(X; Y|Z) \\
 \text{s.t. } I(X; Z) &\leq I_{\text{IB}}
 \end{aligned} \quad (7)$$

where $I(\cdot)$ denotes mutual information, and I_{IB} is a predefined constant that limits the information content of Z , thereby enforcing the extraction of compact yet informative features.

3.3.2 Definitions of feature partitioning

Based on the IB principle, FedFD decomposes the input data $\mathbf{x} \in \mathbf{R}^d$ into two disjoint feature components: essential features $\mathbf{x}_{\text{ess}} \in \mathbf{R}^d$ and supplementary features $\mathbf{x}_{\text{sup}} \in$

$\mathbf{R}^d$, such that $\mathbf{x} = \mathbf{x}_{\text{ess}} + \mathbf{x}_{\text{sup}}$. To mitigate Non-IIDness, only the essential features $\mathbf{x}_{\text{ess}}$, capturing minimal yet sufficient information necessary for effective generalization, are shared among terminals. The supplementary features $\mathbf{x}_{\text{sup}}$, containing almost all original information but primarily redundant or less relevant to performance, are retained locally at each terminal.

To support this decomposition, a valid feature partition strategy is formally defined. Specifically, given a variable X , a valid partition divides it into two disjoint components, X_1 and X_2 , such that the following conditions hold: (i) $H(X_1, X_2 | X) = 0$; (ii) $H(X | X_1, X_2) = 0$; (iii) $I(X_1; X_2) = 0$, where $H(\cdot)$ denotes entropy.

Building upon this, let $X = X_{\text{ess}} + X_{\text{sup}}$ be a valid partition. The component X_{ess} is defined as essential features if it satisfies $I(X; Y | X_{\text{ess}}) = 0$, indicating that X_{ess} captures all information in X relevant to predicting the label Y . Consequently, the remaining component X_{sup} is termed supplementary features.

3.3.3 Objective of FD

With these definitions established, the theoretical objective of FD from the IB perspective is formally presented as follows:

$$\begin{aligned}
 &\min_Z I(X; Y|Z) \\
 &\text{s.t. } I(X; X - Z|Z) \geq I_{\text{FD}}
 \end{aligned} \quad (8)$$

where Z represents the extracted essential features, and $X - Z$ represents the supplementary features retained locally, and I_{FD} is a constant. In (8), the $X - Z$ should represent data mostly conditioned on essential features Z , while the Z should contain necessary information about the label Y . Consequently, learning with this objective can divide data features into essential features and supplementary features, achieving feature partitioning.

To make FD tractable, an objective equal to the original objective in (8) is derived for terminal C_k as follows:

$$\begin{aligned}
 &\min_{\theta} -\mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim P_k} \log p(\mathbf{y}|\mathbf{z}(\mathbf{x}; \theta)) \\
 &\text{s.t. } \|\mathbf{z}(\mathbf{x}; \theta)\|_2^2 \leq \lambda
 \end{aligned} \quad (9)$$

where θ denotes the parameters used to generate the essential features $\mathbf{z}(\cdot; \theta)$, $(\mathbf{x}, \mathbf{y})$ is the input-label pair from the data distribution P_k , $p(\mathbf{y}|\cdot)$ is the predicted probability for label $\mathbf{y}$, and $\lambda > 0$ stands for a constant. Specifically, the learned essential features $\mathbf{z}(\cdot; \theta)$ are designed to predict the label $\mathbf{y}$ with minimal l_2 -norm.

3.3.4 FD via competition mechanism

However, (9) does not explicitly ensure that the supplementary features $\mathbf{x} - \mathbf{z}(\mathbf{x}; \theta)$ remain close to the original

input features $\mathbf{x}$. Therefore, a competition mechanism is introduced explicitly designed to maintain similarity between supplementary and original features in the feature space. Specifically, the supplementary features are explicitly modeled as $g(\mathbf{x};\theta)$, while the essential features are implicitly defined as $\mathbf{z}(\mathbf{x};\theta) \triangleq \mathbf{x} - g(\mathbf{x};\theta)$. This leads to the following formulation:

$$\begin{aligned} \min_{\theta} & -\mathbb{E}_{(\mathbf{x},\mathbf{y}) \sim P_k} \log p(\mathbf{y}|\mathbf{x} - g(\mathbf{x};\theta)) \\ \text{s.t. } & \|\mathbf{x} - g(\mathbf{x};\theta)\|_2^2 \leq \lambda. \end{aligned} \quad (10)$$

Consequently, the essential features maintain predictive capabilities for the label, while the supplementary features closely approximate the original input data.

3.3.5 VAE-based implementation of FD

To practically implement this competition mechanism, a variational autoencoder (VAE) is adopted, parameterized by θ , to explicitly model and reconstruct supplementary features $g(\mathbf{x};\theta)$. Fig. 4 illustrates the VAE-based FD workflow within FedFD.

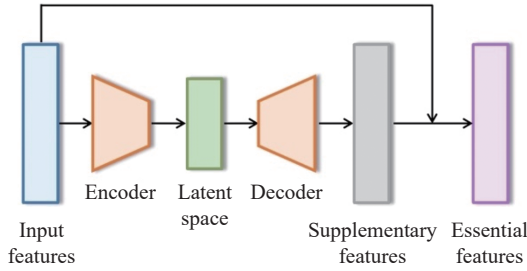


Fig. 4 Illustration of the VAE-based FD workflow within FedFD

The adopted VAE comprises an encoder-decoder structure: the encoder compresses the input data $\mathbf{x}$ into a latent representation, and the decoder reconstructs supplementary features $g(\mathbf{x};\theta)$ based on this representation. The essential features are then computed as $\mathbf{z}(\mathbf{x};\theta) \triangleq \mathbf{x} - g(\mathbf{x};\theta)$. Simultaneously, a local classifier $f(\cdot; \mathbf{w}_k)$, parameterized by $\mathbf{w}_k$, is trained for each terminal C_k to predict the label based on the essential features. The objective for FD is consequently formalized as

$$\begin{aligned} \min_{\theta, \mathbf{w}_k} & -\mathbb{E}_{(\mathbf{x},\mathbf{y}) \sim P_k} l(f(\mathbf{z}(\mathbf{x};\theta); \mathbf{w}_k), \mathbf{y}) \\ \text{s.t. } & \|\mathbf{z}(\mathbf{x};\theta)\|_2^2 \leq \lambda \end{aligned} \quad (11)$$

where $l(\cdot)$ represents the cross-entropy loss, and λ is a tunable hyperparameter.

Based on (11), FD can be realized by decomposing the input data into supplementary features $g(\mathbf{x};\theta)$, locally retained, and essential features $\mathbf{z}(\mathbf{x};\theta) \triangleq \mathbf{x} - g(\mathbf{x};\theta)$, globally shared among terminals for collaborative training. The complete FD process is summarized in Algorithm 2.

Algorithm 2 FD pre-training in the FedFD framework

Note: This FD pre-training stage is executed once before the main T communication rounds in Algorithm 1.

Input Number of FD communication rounds R , local FD epochs E , local datasets $\{D_k\}_{k=1}^K$, aggregation mechanism $M(\cdot)$

Output Updated global model parameters Θ^R

- 1: **Initialization** Each terminal receives the global model parameters Θ^0 from the server S as initialized in Algorithm 1.
 - 2: **for** each FD communication round $r = 1, 2, \dots, R$ **do**:
 - 3: # Global model synchronization
 - 4: The cloud server S sends Θ^r to all terminals C
 - 5: # Local FD (Executed in parallel)
 - 6: **for** each terminal C_k **do**:
 - 7: # Perform local FD training
 - 8: **for** each local FD epoch $e = 1, 2, \dots, E$ **do**:
 - 9: $\Theta_k^{r+1} \leftarrow \text{SGD Update local model using stochastic gradient descent with (11)}$
 - 10: **end for**
 - 11: Each terminal C_k sends Θ_k^{r+1} to the cloud server S
 - 12: **end for**
 - 13: # Model aggregation at the cloud server
 - 14: Update global model:

$$\Theta^{r+1} \leftarrow M(\{\Theta_k^{r+1}\}_{k=1}^K)$$
 - 15: **end for**
 - 16: **Return** Optimized global model parameters Θ^R
-

3.4 SC network architecture

In FL-based RS SC, Non-IID data distributions exacerbate the difficulty of multi-scale feature extraction, as heterogeneous terminals may capture scenes with varied spatial resolutions, viewing angles, and contextual dependencies. To address these challenges, the proposed SC network adopts residual network (ResNet)-10 as its backbone, augmented with the SEM and the PPM. Subsection 3.4.1 outlines the overall network architecture, while Subsections 3.4.2 and 3.4.3 describe SEM and PPM in detail.

3.4.1 Network architecture overview

Fig. 5 illustrates the overall architecture of the proposed SC network within the FedFD framework, which is uniformly deployed across all terminals C and the cloud server S , where ResNet-10 serves as the backbone, augmented with the SEM and PPM. The network adopts ResNet-10 as its backbone, comprising sequentially stacked basic convolutional blocks designed to progressively extract hierarchical spatial and semantic features.

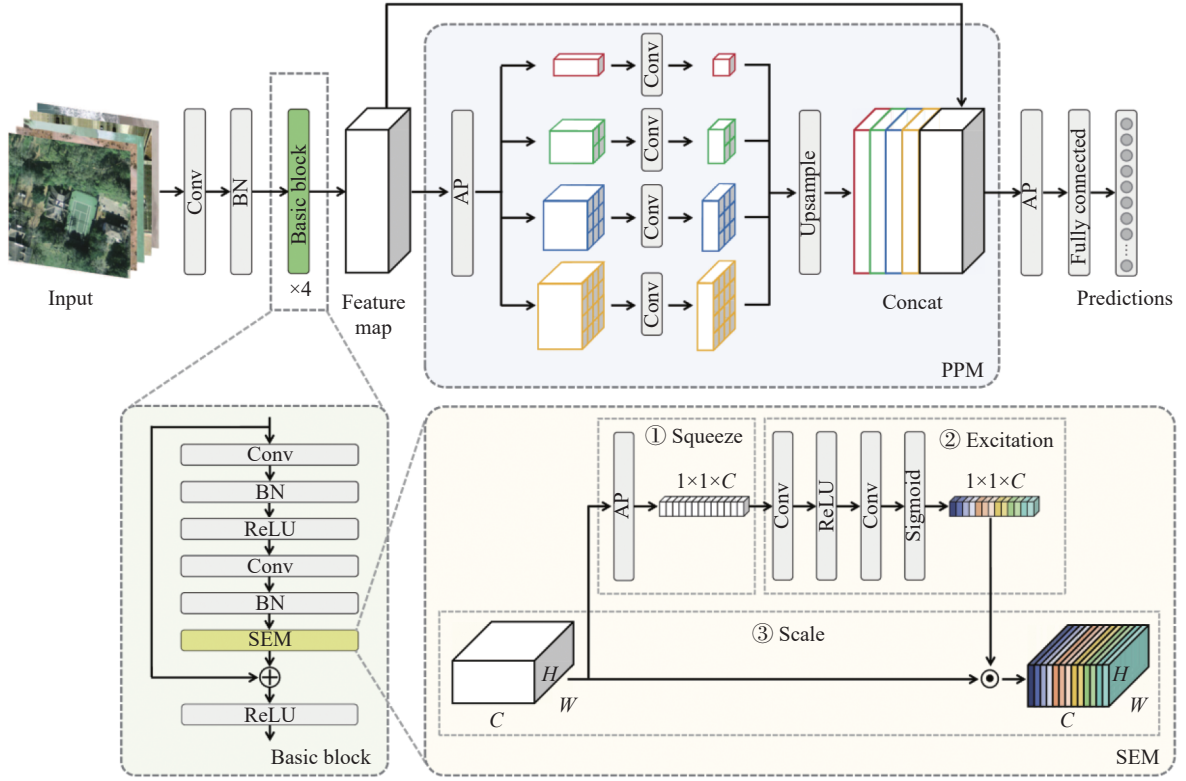


Fig. 5 Architecture of the SC network in the FedFD framework

Specifically, each basic convolutional block consists of two consecutive convolutional layers. The first convolutional layer is followed by a batch normalization (BN) and a rectified linear unit (ReLU) activation function, while the second convolutional layer is followed by a BN and a SEM, which adaptively recalibrates channel-wise feature responses. A residual connection is additionally incorporated within each block to enhance feature reuse and facilitate stable gradient propagation during training.

Formally, the computations within the l th convolutional block are performed in two steps. First, an intermediate feature map $\mathbf{X}_{\text{conv}}^{(l)}$ is computed as

$$\mathbf{X}_{\text{conv}}^{(l)} = \text{BN}(\mathbf{W}_2^{(l)} * \sigma(\text{BN}(\mathbf{W}_1^{(l)} * \mathbf{X}^{(l)}))) \quad (12)$$

where $\mathbf{X}^{(l)} \in \mathbf{R}^{C \times H \times W}$ denotes the input feature map of the l th convolutional block with C channels and spatial dimensions $H \times W$; $\mathbf{W}_1^{(l)}$ and $\mathbf{W}_2^{(l)}$ represent the learnable weights associated with the first and second convolutional layers, respectively; $*$ denotes the convolution operation; and $\sigma(\cdot)$ represents the ReLU activation function.

Next, by integrating the SEM and residual connection, the final output of the l th block is given by

$$\mathbf{X}^{(l+1)} = \sigma(\text{SEM}(\mathbf{X}_{\text{conv}}^{(l)}) + \mathbf{X}^{(l)}). \quad (13)$$

Following the stacked convolutional blocks, the PPM aggregates both local and global contextual information

across multiple receptive field scales, enhancing the ability of the SC network to understand complex multiscale scene features. Detailed descriptions of the SEM and PPM are provided in Subsections 3.4.2 and 3.4.3, respectively. Subsequently, an average pooling (AP) layer aggregates spatial information by computing the mean across all spatial locations:

$$\mathbf{X}_{\text{AP}} = \frac{1}{HW} \sum_{i=1}^H \sum_{j=1}^W \mathbf{X}_{i,j} \quad (14)$$

where H and W denote the height and width of the feature map.

Finally, the aggregated feature vector is passed through a fully connected (FC) layer, followed by a softmax function, to produce the predicted probability distribution over N scene categories:

$$\mathbf{P} = \text{Softmax}(\mathbf{W}_{\text{FC}} \mathbf{X}_{\text{AP}} + \mathbf{b}_{\text{FC}}) \quad (15)$$

where $\mathbf{W}_{\text{FC}} \in \mathbf{R}^{N \times C}$ and $\mathbf{b}_{\text{FC}} \in \mathbf{R}^N$ denote learnable weights and biases of the FC layer, respectively.

3.4.2 SEM

To enhance the discriminative capability of learned features, a SEM is integrated after the second convolutional layer and its subsequent BN layer within each convolutional block. By explicitly modeling inter-channel dependencies, the SEM enables the network to selectively

emphasize informative features while suppressing redundant ones, thereby effectively supporting multiscale representation learning for RS SC. Fig. 5 illustrates the schematic structure of the SEM, which consists of three sequential steps: squeeze, excitation, and scale.

Given an input feature map $X \in \mathbf{R}^{C \times H \times W}$, where C , H , and W represent the number of channels, height, and width, respectively, the SEM recalibrates channel-wise features through the following operations. First, the squeeze step applies AP, as defined in (14), independently across each channel, producing a compact global descriptor $X_{sq} \in \mathbf{R}^{C \times 1 \times 1}$ that summarizes channel-level spatial information:

$$X_{sq} = \frac{1}{HW} \sum_{i=1}^H \sum_{j=1}^W X_{i,j}. \quad (16)$$

Next, the excitation step explicitly models inter-channel dependencies using two FC layers implemented as 1×1 convolutions with a bottleneck structure:

$$X_{ex} = \delta(W_2 \sigma(W_1 X_{sq})) \quad (17)$$

where $W_1 \in \mathbf{R}^{r \times C}$ and $W_2 \in \mathbf{R}^{C \times r}$ are learnable weight matrices of the convolutional layers, r is the reduction ratio controlling the dimensionality of the bottleneck, and $\sigma(\cdot)$ and $\delta(\cdot)$ denote ReLU and sigmoid activation functions, respectively. The resulting attention weights vector $X_{ex} \in \mathbf{R}^{C \times 1 \times 1}$ captures the significance of each channel for the given feature representation.

Finally, during the scale step, the original feature map X is recalibrated by performing channel-wise multiplication between X and the attention weights vector X_{ex} :

$$X_{SEM} = X \odot X_{ex} \quad (18)$$

where $\odot$ denotes element-wise multiplication applied independently for each channel, and X_{SEM} is the recalibrated feature map after adaptive channel-wise weighting.

3.4.3 PPM

To effectively handle multiscale features inherent in RS

SC tasks, we integrate the PPM, inspired by the pyramid scene parsing network (PSPNet) [42]. As illustrated in Fig. 5, the PPM is positioned after the final convolutional block to aggregate spatial context information across multiple scales, thus capturing diverse scene characteristics.

Given an input feature map $X \in \mathbf{R}^{C \times H \times W}$, the PPM applies AP at four spatial scales $S = \{1 \times 1, 2 \times 2, 3 \times 3, 6 \times 6\}$, producing pooled feature maps $X_{pool}^{(s)} \in \mathbf{R}^{C \times H_s \times W_s}$ that effectively represent multiscale spatial features. Each pooled feature map is further refined via a 1×1 convolutional layer, BN, and a ReLU activation:

$$X^{(s)} = \sigma(\text{BN}(W^{(s)} * X_{pool}^{(s)})) \quad (19)$$

where $X^{(s)}$ represents learnable convolutional weights at scale s , and $\sigma(\cdot)$ denotes the ReLU activation. These refined feature maps are subsequently bilinearly upsampled to match the original spatial dimensions $H \times W$.

The original input feature map X is then concatenated along the channel dimension with all upsampled multiscale feature maps $\{X^{(s)}\}_{s \in S}$, yielding the final output of the PPM:

$$X_{PPM} = \text{Concat}(X, X^{(s)} | s \in S) \quad (20)$$

where $\text{Concat}(\cdot)$ denotes channel-wise concatenation. This integration strategy effectively combines fine-grained local information from X with coarse-to-fine contextual features from the multiscale branches, significantly enhancing the comprehensive representation of spatial dependencies and improving multiscale scene understanding.

4. Experiments and discussions

4.1 Dataset

The proposed FedFD framework is evaluated on two widely adopted RS SC datasets: UC-Merced [43] and NWPU-RESISC45 [44], which exhibit different scene characteristics. Details of these datasets are summarized in Table 1.

Table 1 Details of UC-Merced and NWPU-RESISC45 RS SC datasets

Dataset	Source	Number of images per class	Number of classes	Spatial resolution/m	Image size	Training/Test ratio
UC-Merced	USGS National Map	100	21	0.3	256×256	50% / 50%, 80% / 20%
NWPU-RESISC45	Google Earth	700	45	0.2 ~ 30	256×256	10% / 90%, 20% / 80%

4.1.1 UC-Merced

The UC-Merced dataset [43] is extensively used for evaluating RS SC methodologies due to its diverse categories and high-resolution imagery. It comprises 2100 aerial orthoimages collected from the United States Geo-

logical Survey (USGS) National Map, featuring 21 scene classes. Each class contains 100 samples, and each image is cropped to a uniform size of 256×256 pixels with a spatial resolution of 0.3 m.

The dataset spans diverse urban and suburban regions across the United States, including cities such as Birm-

ingham, Boston, and Los Angeles. Scene categories cover various land-use types, including agricultural, residential,

industrial, and recreational areas. Fig. 6 presents two representative images from each category.

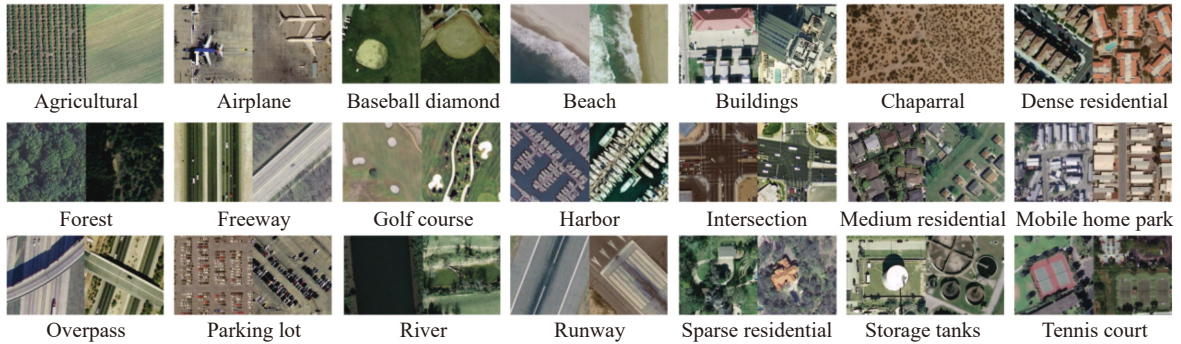


Fig. 6 Sample images from the UC-Merced dataset, illustrating two examples from each of the 21 scene categories

4.1.2 NWPU-RESISC45

The NWPU-RESISC45 dataset [44], developed by North-western Polytechnical University, is among the largest publicly available benchmarks for RS SC tasks. It contains 31 500 images grouped into 45 scene categories, each category containing 700 images of size 256×256 pixels. Spatial resolution ranges from 0.2 m to 30 m.

The images are collected from Google Earth, covering over 100 countries across diverse geographic and economic regions, including developing, transitional, and

developed areas. Representative scene categories encompass natural landscapes, residential zones, transportation infrastructure, and industrial regions. Additionally, NWPU-RESISC45 features significant variations in weather conditions, seasonal changes, and spatial resolutions, presenting notable challenges due to high intra-class diversity, inter-class similarity, and multiscale representations. This complexity makes it a valuable benchmark for assessing the robustness and generalization capability of RS SC methods. Fig. 7 illustrates sample images from different categories within the dataset.

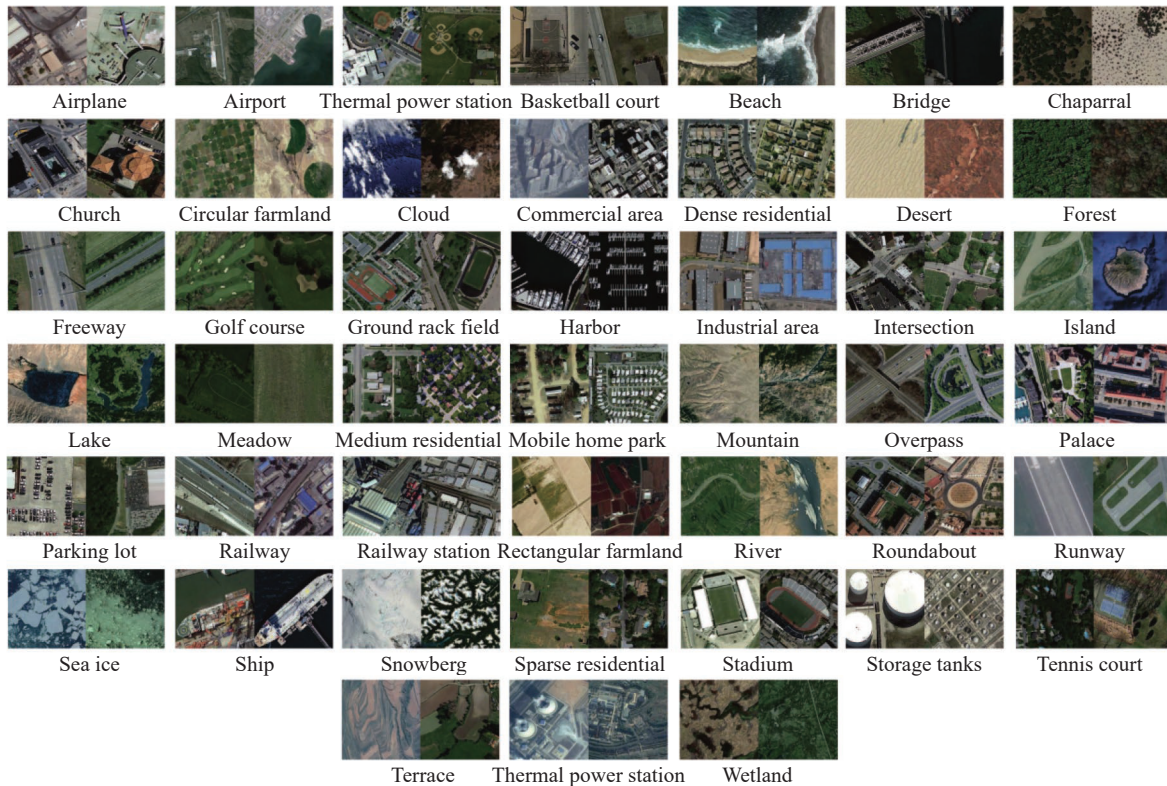


Fig. 7 Sample images from the NWPU-RESISC45 dataset, illustrating two examples from each of the 45 scene categories

4.2 Experimental settings

4.2.1 Dataset distribution

Following the approach in [25,45,46], LDA is utilized to simulate Non-IID data distributions, a widely adopted method for partitioning datasets among terminals in a statistically controlled manner.

LDA is a probabilistic method that assigns terminal-specific class distributions based on the Dirichlet distribution [47]. Given a dataset with N classes, the distribution for each terminal C_k is sampled as

$$P_k \sim \text{Dir}(\alpha) \quad (21)$$

where P_k represents the class distribution for terminal C_k , $\text{Dir}(\cdot)$ represents Dirichlet distribution, and $\alpha > 0$ is a concentration parameter controlling the Non-IID degree. Fig. 2 illustrates the effect of LDA sampling with varying α . As α decreases, the resulting data distribution becomes increasingly imbalanced. Conversely, larger values of α produce a more uniform distribution across terminals, approximating an IID scenario as $\alpha \rightarrow \infty$. To evaluate the effectiveness of the proposed FedFD, we apply LDA-based partitioning to the UC-Merced and NWPU-RESISC45 datasets with $\alpha = 0.1, 0.5$, and 1.0 , representing different levels of class imbalance under various scenarios.

4.2.2 Implementation details

The FedFD framework is implemented using PyTorch, and all experiments are executed on a single NVIDIA A800 GPU. The FL system is configured with five terminals. During the FD pre-training stage, each terminal performs one local epoch across 15 FD communication rounds, and the VAE employed is trained with a batch size of 64. After this stage, the FD module is frozen and only used to generate the shared feature dataset for subsequent federated training. For the SC network, ResNet-10 is employed as the backbone architecture, which consists of four convolutional blocks with channel dimensions of $\{32, 64, 128, 256\}$. The SC network training also utilizes a batch size of 64, with initial learning rates set to 2×10^{-2} for the NWPU-RESISC45 dataset and 1×10^{-2} for the UC-Merced dataset.

FedFD is designed as a plug-in approach, enabling seamless integration with various existing FL algorithms

via different aggregation mechanisms $M(\cdot)$. To comprehensively assess its effectiveness, FedFD is integrated with three representative FL methods, including FedAvg [15], FedProx [23], and FedNova [25], across various configurations and Non-IID levels.

4.3 Evaluation metrics

To quantitatively assess the performance of the proposed FedFD, two widely recognized metrics, overall accuracy (OA) and the confusion matrix (CM), are adopted. These metrics provide comprehensive insights into both overall performance and per-class classification details.

4.3.1 OA

The OA is defined as the ratio between the number of correctly classified samples and the total number of samples in the dataset. Higher OA values indicate better overall classification performance across all classes.

4.3.2 CM

The CM is a two dimensional representation of per-class classification performance. In the matrix, rows represent the ground-truth class labels, while columns correspond to the predicted class labels. Each matrix cell indicates the proportion of samples from the true class (row) predicted as the corresponding class (column), normalized for consistency. Diagonal cells represent the per-class accuracies, reflecting the proportion of correctly classified samples within each class. A color bar alongside the matrix visually conveys these proportions, with darker colors indicating higher values.

4.4 SC experimental results

The performance of the proposed FedFD framework is evaluated by integrating it with three representative baseline FL methods. For each method, the OA of its original (vanilla) version and the FedFD-integrated variant under varying training ratios and Non-IID conditions are compared.

4.4.1 UC-Merced dataset

Table 2 presents the OA comparison between FedFD-integrated methods and their vanilla counterparts on the UC-Merced dataset. Experiments are conducted under two training ratios (50% and 80%) with varying Non-IID levels controlled by the parameter α .

Table 2 OA comparison of FedFD-integrated and vanilla methods on the UC-Merced dataset under varying training ratios and Non-IID levels

Method	Training ratio = 50%			Training ratio = 80%		
	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 1.0$	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 1.0$
FedAvg (vanilla) [15]	77.045	81.606	85.371	78.571	86.190	87.619
FedProx (vanilla) [23]	78.658	82.280	83.478	80.609	86.575	87.781

Continued

Method	Training ratio = 50%			Training ratio = 80%		
	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 1.0$	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 1.0$
FedNova (vanilla) [25]	79.305	82.582	84.668	84.013	86.667	90.476
FedFD+FedAvg	83.619 (6.574 $\uparrow$)	86.762 (5.156 $\uparrow$)	89.764 (4.394 $\uparrow$)	85.938 (7.366 $\uparrow$)	92.976 (6.786 $\uparrow$)	94.976 (7.357 $\uparrow$)
FedFD+FedProx	83.376 (4.718 $\uparrow$)	88.750 (6.470 $\uparrow$)	89.359 (5.881 $\uparrow$)	87.235 (6.626 $\uparrow$)	91.850 (5.275 $\uparrow$)	95.652 (7.871 $\uparrow$)
FedFD+FedNova	82.211 (2.906 $\uparrow$)	87.848 (5.266 $\uparrow$)	89.840 (5.172 $\uparrow$)	87.143 (3.130 $\uparrow$)	92.115 (5.449 $\uparrow$)	95.238 (4.762 $\uparrow$)

(i) Training ratio=50%. FedFD consistently delivers substantial OA improvements at the 50% training ratio. In the most imbalanced scenario ($\alpha = 0.1$), FedAvg integrated with FedFD achieves an accuracy of 83.619%, surpassing vanilla FedAvg (77.045%) by 6.574%. Similar improvements are observed for FedProx and FedNova, with accuracy increases of 4.718% and 2.906%, respectively.

As the imbalance moderates ($\alpha = 0.5$), FedFD continues to exhibit strong performance gains: FedProx integrated with FedFD reaches an accuracy of 88.750%, improving vanilla FedProx by 6.470%, and FedAvg integrated with FedFD reaches 86.762%, improving vanilla FedAvg by 5.156%. Even in the least imbalanced scenario ($\alpha = 1.0$), FedFD remains effective, with FedProx integrated with FedFD achieving an accuracy of 89.359%, outperforming its vanilla version by 5.881%.

(ii) Training ratio=80%. When increasing the training ratio to 80%, FedFD maintains its superiority across varying Non-IID levels. In the highly imbalanced scenario

($\alpha = 0.1$), FedAvg integrated with FedFD achieves 85.938%, which is a 7.366% improvement over vanilla FedAvg (78.571%). Similarly, FedProx and FedNova integrated with FedFD attain accuracies of 87.235% and 87.143%, improving their vanilla counterparts by 6.626% and 3.130%, respectively. At moderate imbalance ($\alpha = 0.5$), FedAvg integrated with FedFD further elevates accuracy to 92.976%, representing a notable improvement of 6.786%, while FedNova integrated with FedFD reaches 92.115%, improving vanilla FedNova by 5.449%. In the least imbalanced setting ($\alpha = 1.0$), FedProx integrated with FedFD achieves 95.652%, surpassing vanilla FedProx (87.781%) by a remarkable 7.871%.

4.4.2 NWPU-RESISC45 dataset

Table 3 reports the OA results for FedFD-integrated methods and their corresponding vanilla baselines on the NWPU-RESISC45 dataset. Evaluations are conducted under two training ratios, 10% and 20%, with varying Non-IID levels.

Table 3 OA comparison of FedFD-integrated and vanilla methods on the NWPU-RESISC45 dataset under varying training ratios and Non-IID levels

Method	Training ratio = 10%			Training ratio = 20%		
	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 1.0$	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 1.0$
FedAvg (vanilla) [15]	62.891	67.068	69.278	74.516	79.600	80.087
FedProx (vanilla) [23]	65.241	69.379	70.643	72.101	80.047	80.852
FedNova (vanilla) [25]	65.054	69.691	70.826	74.282	78.363	80.306
FedFD+FedAvg	70.261(7.370 $\uparrow$)	75.216(8.148 $\uparrow$)	76.658(7.380 $\uparrow$)	76.568(2.053 $\uparrow$)	84.829(5.230 $\uparrow$)	85.202(5.115 $\uparrow$)
FedFD+FedProx	70.230(4.989 $\uparrow$)	76.795(7.415 $\uparrow$)	77.731(7.087 $\uparrow$)	76.898(4.797 $\uparrow$)	84.927(4.881 $\uparrow$)	84.935(4.083 $\uparrow$)
FedFD+FedNova	71.243(6.189 $\uparrow$)	76.952(7.261 $\uparrow$)	77.580(6.755 $\uparrow$)	76.312(2.030 $\uparrow$)	84.724(6.361 $\uparrow$)	85.448(5.142 $\uparrow$)

(i) Training ratio=10%. At a training ratio of 10%, FedFD integration consistently yields significant OA improvements. With the highest class imbalance ($\alpha = 0.1$), FedAvg integrated with FedFD achieves 70.261%, outperforming its vanilla version by 7.370%. Similar performance gains are noted for FedProx and FedNova, with improvements of 4.989% and 6.189%, respectively. As the Non-IID level decreases ($\alpha = 0.5$), FedFD remains highly effective: FedAvg integrated with FedFD improves by 8.148%, achieving an OA of

75.216%, while FedProx integrated with FedFD reaches 76.795%, improving baseline FedProx by 7.415%. Under milder imbalance ($\alpha = 1.0$), FedAvg integrated with FedFD achieves an OA of 76.658%, representing a 7.380% improvement over the vanilla FedAvg.

(ii) Training ratio=20%. At an increased training ratio of 20%, FedFD integration continues to deliver noticeable OA improvements, albeit slightly smaller. Under severe imbalance ($\alpha = 0.1$), FedAvg integrated with FedFD achieves 76.568%, a modest gain of 2.053%,

while FedProx and FedNova integrated with FedFD achieve accuracies of 76.898% and 76.312%, improving their vanilla baselines by 4.797% and 2.030%, respectively. As the data distribution becomes less imbalanced ($\alpha = 0.5$), FedAvg integrated with FedFD achieves 84.829%, an increase of 5.230%, and FedNova integrated with FedFD reaches 84.724%, improving by 6.361%. At $\alpha = 1.0$, FedNova integrated with FedFD achieves 85.448%, showing a 5.142% improvement over the vanilla baseline.

4.5 Scalability with increasing terminal numbers

This subsection evaluates the scalability of the proposed FedFD by varying the number of terminals and compares its performance with the vanilla FedAvg baseline. As shown in Fig. 8, the OA of FedFD-integrated methods and the baseline are presented across different numbers of terminals (K) and Non-IID levels. Lines in different colors represent different methods, as indicated in the legend. Specifically, the left, middle, and right columns correspond to $\alpha=1$, $\alpha=0.5$, and $\alpha=1.0$, respectively.

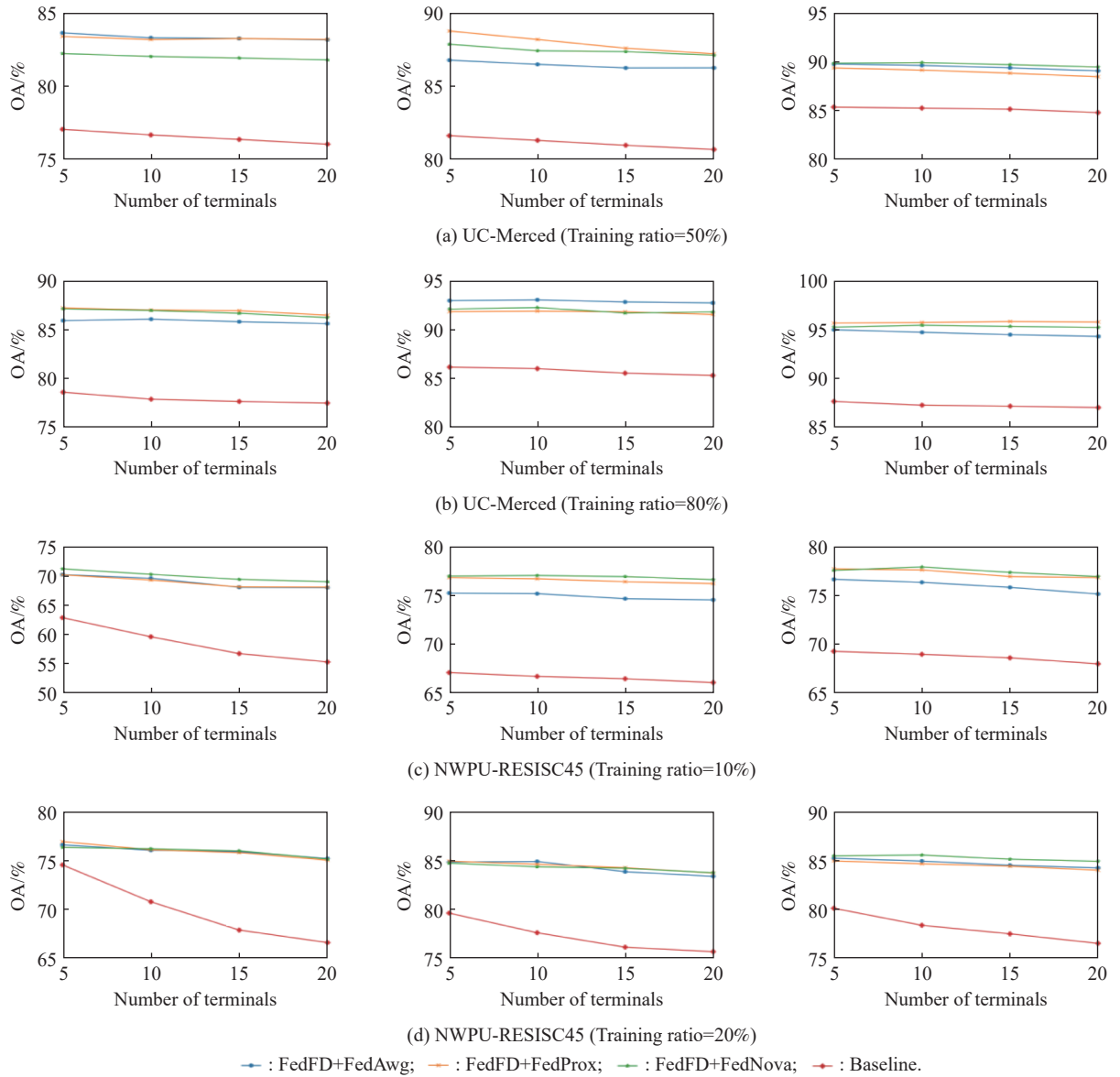


Fig. 8 Scalability comparison of the proposed FedFD and baseline method on two datasets under various Non-IID levels with varying terminal numbers K

4.5.1 UC-Merced dataset

Fig. 8(a) and Fig. 8(b) illustrate that FedFD consistently

outperforms the baseline across all evaluated terminal numbers and training ratios. Specifically, Fig. 8(a) shows

that under severe data imbalance ($\alpha = 0.1$) with a 50% training ratio, FedFD achieves notable improvements in OA compared to the baseline method. For instance, FedAvg integrated with FedFD attains a 7.123% accuracy improvement at $K = 20$, while FedProx and FedNova integrated with FedFD achieve gains of 7.148% and 5.752%, respectively. At a higher training ratio of 80%, as shown in Fig. 8(b), FedFD further enhances its performance advantage. At $\alpha = 0.1$ and $K = 20$, FedProx integrated with FedFD achieves a substantial accuracy improvement of 9.037%. Additionally, at $\alpha = 1.0$ and $K = 20$, FedNova integrated with FedFD attains an accuracy of 95.212%, representing an 8.219% improvement over the baseline.

Furthermore, FedFD demonstrates greater stability in OA performance as the number of terminals increases. For example, at $\alpha = 0.1$ and a 50% training ratio, as shown in Fig. 8(a), FedAvg integrated with FedFD shows only a slight OA decrease from 83.619% at $K = 5$ to 83.151% at $K = 20$, corresponding to a reduction of 0.468%. The baseline shows a more pronounced decline, with OA decreasing from 77.045% to 76.028%, representing a reduction of 1.017%.

4.5.2 NWPU-RESISC45 dataset

Similar to the UC-Merced dataset, FedFD also demonstrates advantages across all training ratios and terminal numbers, which is shown in Fig. 8(c) and Fig. 8(d). In particular, the performance improvements are more pronounced under challenging conditions with lower training ratios and higher Non-IID degrees. As shown in Fig. 8(c), at a 10% training ratio and $\alpha = 0.1$, FedFD yields substantial improvements at $K = 20$, with FedAvg, FedProx, and FedNova integrated with FedFD surpassing the baseline by 12.725%, 12.798%, and 13.752%, respectively. As the number of terminals increases, the performance gain becomes more pronounced. For instance, as illustrated in Fig. 8(d), at a 20% training ratio and $\alpha = 0.1$, the OA improvements for FedAvg with FedFD at $K = 5, 10, 15, 20$ are 2.053%, 5.269%, 8.020%, and 8.598%, respectively.

Moreover, FedFD also exhibits superior stability as the terminal number increases. While FedFD shows a slight decline in OA, the drop is considerably smaller than that of the baseline. For example, as illustrated in Fig. 8(c), at a 10% training ratio and $\alpha = 0.1$, the OA of FedNova with FedFD decreases slightly from 71.243% at $K = 5$ to 69.044% at $K = 20$, resulting in a drop of 2.199%. In contrast, the baseline OA drops from 62.891% to 55.312%, with a reduction of 7.579%.

4.6 Ablation study for the SEM and PPM

The SEM and PPM are incorporated into the SC network to effectively address scale variations in RS images and enhance multiscale feature extraction capabilities. Specifically, the SEM models inter-channel dependencies to emphasize informative features and suppress redundant ones, thereby improving feature discrimination and multiscale representation. Meanwhile, the PPM aggregates contextual information at multiple spatial scales via pyramid pooling, enhancing the capability of the SC network to recognize objects and scenes with diverse scales.

To evaluate the individual and combined contributions of SEM and PPM, we conduct ablation experiments on the UC-Merced and NWPU-RESISC45 datasets with different training ratios under three Non-IID levels. The experimental results are summarized in Table 4–Table 6, corresponding to $\alpha = 0.1, 0.5$, and 1.0 respectively.

Table 4 Ablation experiments of the SEM and PPM in the SC network on two datasets ($K = 5, \alpha = 0.1$) %

Configuration		UC-Merced		NWPU-RESISC45	
SEM	PPM	50%	80%	10%	20%
–	–	73.046	80.238	62.483	72.089
–	✓	76.227	83.066	63.559	73.926
✓	–	77.714	82.619	64.657	72.545
✓	✓	83.619	85.938	70.261	76.568

Table 5 Ablation experiments of the SEM and PPM in the SC network on two datasets ($K = 5, \alpha = 0.5$) %

Configuration		UC-Merced		NWPU-RESISC45	
SEM	PPM	50%	80%	10%	20%
–	–	80.194	83.061	71.139	78.764
–	✓	84.767	85.967	72.373	81.609
✓	–	82.525	84.143	71.823	80.962
✓	✓	86.762	92.976	75.216	84.829

It is observed that each module individually contributes to OA improvements, while the combination consistently delivers the best performance. As shown in Table 4, under severe data imbalance with $\alpha = 0.1$, combining SEM and PPM increases the OA on UC-Merced from 73.046% to 83.619% when 50% of the training data is used. Similarly, on NWPU-RESISC45 with a 10% training ratio, OA improves from 62.483% to 70.261%.

This trend persists under the moderate Non-IID level, as presented in Table 5. On UC-Merced, the OA rises from 83.061% to 92.976% with 80% training data, and on

NWPU-RESISC45, from 71.139% to 75.216% with 10% training data. At the mildest Non-IID level, as shown in Table 6, the combination of SEM and PPM continues to provide performance gains. Specifically, the OA on UC-Merced increases from 86.761% to 89.764% with 50% of the training data, while on NWPU-RESISC45, it rises from 81.156% to 85.202% using 20% of the data. These consistent improvements across various experimental settings confirm the effectiveness of SEM and PPM in enhancing multiscale feature extraction in RS SC.

Table 6 Ablation experiments of the SEM and PPM in the SC network on two datasets ($K = 5$, $\alpha = 1.0$) %

Configuration		UC-Merced		NWPU-RESISC45	
SEM	PPM	50%	80%	10%	20%
–	–	86.761	91.051	74.562	81.156
–	✓	87.142	92.839	75.045	82.505
✓	–	88.855	91.904	75.554	82.979
✓	✓	89.764	94.976	76.658	85.202

To further interpret the differences between the two modules, we analyze their performance variations across different Non-IID levels as follows. In Tables 4–6, the PPM achieves stronger robustness under highly Non-IID settings (smaller α) by aggregating multi-scale global context to alleviate cross-terminal feature inconsistency. In contrast, as α increases and data distributions become more uniform, the SEM demonstrates superior performance by refining local feature representations and improving fine-grained scene discrimination.

4.7 Visual analysis

In distributed FL systems, each terminal observes its private and Non-IID data slices. For RS SC tasks, such Non-IIDness amplifies the existing multiscale challenge, as identical scene categories can appear at vastly different spatial scales across terminals, while distinct categories might share similar local features. Consequently, misclassifications typically occur when scale variations intersect with cross-terminal data biases.

To comprehensively evaluate the performance and interpretability of the proposed FedFD framework, two levels of visualization analysis are performed. First, CMs are presented to illustrate the overall classification performance of FedFD under representative settings of $K=5$ and $\alpha = 0.5$. Then, feature heatmaps of the SEM and PPM modules are visualized to reveal their respective attention patterns, providing insights into how these modules contribute to the network. The CMs for the 21-class UC-

Merced dataset are shown in Fig. 9, and those for the NWPU-RESISC45 dataset are presented in Fig. 10. The SEM and PPM heatmap visualizations are provided in Fig. 11.

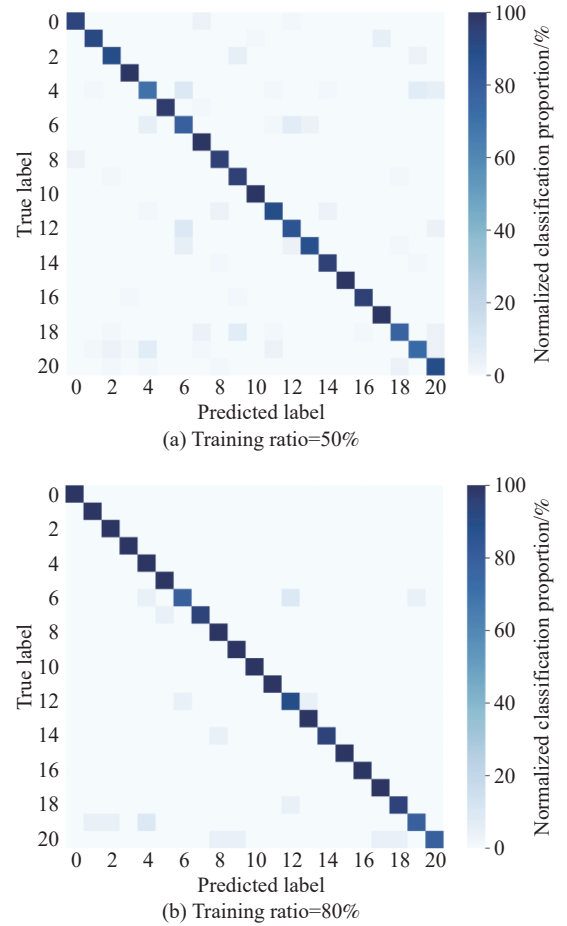
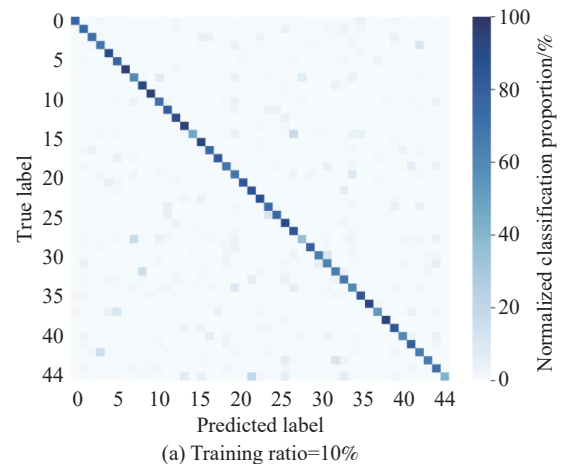


Fig. 9 CMs of classification results achieved by FedAvg integrated with FedFD on the UC-Merced dataset ($K = 5$, $\alpha = 0.5$)



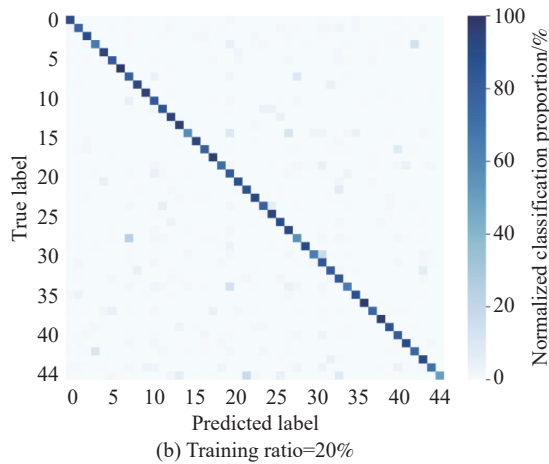


Fig. 10 CMs of classification results achieved by FedAvg integrated with FedFD on the NWPU-RESISC45 dataset ($K = 5$, $\alpha = 0.5$)

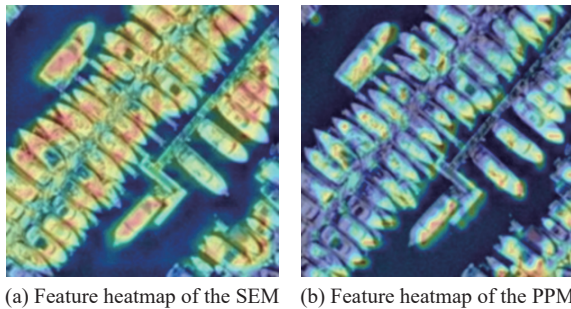


Fig. 11 Grad-CAM visualizations of the SEM and PPM modules on a representative harbor sample from the UC-Merced dataset

4.7.1 UC-Merced dataset

Fig. 9(a) reveals clear diagonal dominance, indicating accurate classification for most scenes. Nevertheless, a few visually similar classes such as “medium residential” and “dense residential” show slight confusion, visible as lighter off-diagonal entries.

With the training ratio increased to 80%, as presented in Fig. 9(b), these confusions diminish significantly, and the diagonal intensifies noticeably. This improvement shows that additional training data helps FedFD better resolve subtle inter-class ambiguities, particularly those arising from multiscale features.

4.7.2 NWPU-RESISC45 dataset

Fig. 10 illustrates similar observations for NWPU-RESISC45. When each terminal utilizes only 10% of available data, as depicted in Fig. 10(a), most categories remain distinguishable, but sporadic confusion still presents, especially between visually related classes such as “river” versus “lake” and various residential sub-types. These misclassifications primarily occur at the intersec-

tion of significant scale variations and inter-terminal data biases.

Upon increasing the training ratio to 20%, shown in Fig. 10(b), diagonal intensities clearly strengthen and off-diagonal confusions are largely reduced. This demonstrates that FedFD can effectively leverage the richer, though still Non-IID, training data to enhance the consistency of multiscale feature representations across distributed terminals.

4.7.3 Visualization of SEM and PPM modules

To further illustrate the interpretability of the network, Fig. 11 presents the Grad-CAM feature heatmaps of the SEM and PPM modules using a representative harbor sample from the UC-Merced dataset. The SEM heatmap concentrates along fine-grained structures such as boat hull edges and masts, indicating that channel-wise recalibration enhances locally discriminative details. The PPM heatmap exhibits broader activations across boat clusters and water, capturing the global layout. These complementary attention patterns demonstrate that the SEM refines local cues, while the PPM supplies global context, jointly enhancing the model’s scene understanding.

5. Conclusions

This paper proposed FedFD, a novel federated framework with FD for FL-based Non-IID RS SC. FedFD enables collaborative training by aggregating model parameters from distributed terminals without transmitting raw RS data. To address Non-IIDness, the PFS strategy that globally distills and shares essential features while retaining locally specific information is introduced. Moreover, the integration of the SEM and PPM enhances the capability of the SC network to effectively capture multiscale scene features. Extensive experiments conducted on the UC-Merced and NWPU-RESISC45 datasets under various Non-IID degrees demonstrate that FedFD achieves superior classification accuracy and scalability compared to baseline methods. While FedFD effectively mitigates data Non-IIDness and multiscale variations, residual misclassifications remain among scene categories that exhibit highly similar spatial features across scales. Future research could explore adaptive feature-sharing strategies and dynamic aggregation methods to further enhance the discriminative capability of FedFD under severe heterogeneity. In addition, system-level optimization for practical deployment, such as adaptive model partitioning and efficiency balancing across heterogeneous terminals and cloud servers, will be further investigated in future work.

References

- [1] WANG J J, LI W, ZHANG M M, et al. Remote-sensing

- scene classification via multistage self-guided separation network. *IEEE Trans. on Geoscience and Remote Sensing*, 2023, 61: 5615312.
- [2] ZHENG X T, YUAN Y, LU X Q. A deep scene representation for aerial scene classification. *IEEE Trans. on Geoscience and Remote Sensing*, 2019, 57(7): 4799–4809.
 - [3] XIA J M, ZHOU Y, TAN L. DBGA-Net: dual-branch global-local attention network for remote sensing scene classification. *IEEE Geoscience and Remote Sensing Letters*, 2023, 20: 7502305.
 - [4] WANG Q, HUANG W, XIONG Z T, et al. Looking closer at the scene: multiscale representation learning for remote sensing image scene classification. *IEEE Trans. on Neural Networks and Learning Systems*, 2020, 33(4): 1414–1428.
 - [5] ZHANG L P, ZHANG L F, DU B. Deep learning for remote sensing data: a technical tutorial on the state of the art. *IEEE Geoscience and Remote Sensing Magazine*, 2016, 4(2): 22–40.
 - [6] WANG Q, LIU S T, CHANUSSOT J, et al. Scene classification with recurrent attention of VHR remote sensing images. *IEEE Trans. on Geoscience and Remote Sensing*, 2018, 57(2): 1155–1167.
 - [7] ZHAO Z C, LI J Q, LUO Z, et al. Remote sensing image scene classification based on an enhanced attention module. *IEEE Geoscience and Remote Sensing Letters*, 2020, 18(11): 1926–1930.
 - [8] LIU Y F, ZHONG Y F, QIN Q Q. Scene classification based on multiscale convolutional neural network. *IEEE Trans. on Geoscience and Remote Sensing*, 2018, 56(12): 7109–7121.
 - [9] HE N J, FANG L Y, LI S T, et al. Skip-connected covariance network for remote sensing scene classification. *IEEE Trans. on Neural Networks and Learning Systems*, 2019, 31(5): 1461–1474.
 - [10] GAO S H, CHENG M M, ZHAO K, et al. Res2net: a new multi-scale backbone architecture. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2019, 43(2): 652–662.
 - [11] WANG Y L, WANG Z R, CHENG P R, et al. DCM: a distributed collaborative training method for the remote sensing image classification. *IEEE Trans. on Geoscience and Remote Sensing*, 2023, 61: 5605018.
 - [12] LI J Z, GONG M G, LIU Z T, et al. Toward multiparty personalized collaborative learning in remote sensing. *IEEE Trans. on Geoscience and Remote Sensing*, 2024, 62: 4503616.
 - [13] XU Y H, BAI T, YU W K, et al. AI security for geoscience and remote sensing: challenges and future trends. *IEEE Geoscience and Remote Sensing Magazine*, 2023, 11(2): 60–85.
 - [14] LI D X, XIE W Y, LI Y S, et al. FedFusion: manifold-driven federated learning for multi-satellite and multi-modality fusion. *IEEE Trans. on Geoscience and Remote Sensing*, 2023, 62: 5500813.
 - [15] MCMAHAN H B, MOORE E, RAMAGE D, et al. Communication-efficient learning of deep networks from decentralized data. *Proc. of the Artificial Intelligence and Statistics*, 2017: 1273–1282.
 - [16] ZHANG X K, ZHANG B N, YU W K, et al. Federated deep learning with prototype matching for object extraction from very-high-resolution remote sensing images. *IEEE Trans. on Geoscience and Remote Sensing*, 2023, 61: 5603316.
 - [17] TANG X C, YAN X C, YUAN X J, et al. FedLD: federated learning for privacy-preserving collaborative landslide detection. *IEEE Geoscience and Remote Sensing Letters*, 2024, 21: 8003105.
 - [18] KAWAGUCHI K, KAEHLBLING L P, BENGIO Y. Generalization in deep learning. <https://arxiv.org/abs/1710.05468>.
 - [19] JAKUBOVITZ D, GIRYES R, RODRIGUES M R D. Compressed sensing and its applications. 2017. Cham: Springer International Publishing, 2019.
 - [20] KAIROUZ P, MCMAHAN H B, AVENT B, et al. Advances and open problems in federated learning. *Foundations and Trends in Machine Learning*, 2021, 14(1/2): 1–210.
 - [21] LI X X, JIANG M R, ZHANG X F, et al. FedBN: federated learning on non-IID features via local batch normalization. <https://arxiv.org/abs/2102.07623>.
 - [22] KARIMIREDDY S P, KALE S, MOHRI M, et al. Scaffold: stochastic controlled averaging for federated learning. *Proc. of the International Conference on Machine Learning*, 2020: 5132–5143.
 - [23] LI T, SAHU A K, ZAHEER M, et al. Federated optimization in heterogeneous networks. <https://arxiv.org/abs/1812.06127>.
 - [24] WANG H Y, YUROCHKIN M, SUN Y K, et al. Federated learning with matched averaging. <https://arxiv.org/abs/2002.06440>.
 - [25] WANG J Y, LIU Q H, LIANG H, et al. Tackling the objective inconsistency problem in heterogeneous federated optimization. *Advances in Neural Information Processing Systems*, 2020, 33: 7611–7623.
 - [26] XIE Z J, SONG S H. FedKL: tackling data heterogeneity in federated reinforcement learning by penalizing KL divergence. *IEEE Journal on Selected Areas in Communications*, 2023, 41(4): 1227–1242.
 - [27] TAN Y, LIU Y X, LONG G D, et al. Federated learning on non-IID graphs via structural knowledge sharing. *Proc. of the AAAI Conference on Artificial Intelligence*, 2023, 37: 9953–9961.
 - [28] YAO X W, HAN J W, CHENG G, et al. Semantic annotation of high-resolution satellite images via weakly supervised learning. *IEEE Trans. on Geoscience and Remote Sensing*, 2016, 54(6): 3660–3671.
 - [29] GAO Y H, ZHANG M M, LI W, et al. Adversarial complementary learning for multisource remote sensing classification. *IEEE Trans. on Geoscience and Remote Sensing*, 2023, 61: 5505613.
 - [30] WANG J J, LI W, GAO Y H, et al. Hyperspectral and SAR image classification via multiscale interactive fusion network. *IEEE Trans. on Neural Networks and Learning Systems*, 2022, 34(12): 10823–10837.
 - [31] HUANG X, LIU H, ZHANG L P. Spatiotemporal detection and analysis of urban villages in mega city regions of China using high-resolution remotely sensed imagery. *IEEE Trans. on Geoscience and Remote Sensing*, 2015, 53(7): 3639–3657.
 - [32] HU F, XIA G S, HU J W, et al. Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery. *Remote Sensing*, 2015, 7(11): 14680–14707.
 - [33] YU F, WANG D Q, SHELHAMER E, et al. Deep layer aggregation. *Proc. of the IEEE Conference on Computer*

- Vision and Pattern Recognition, 2018: 2403–2412.
- [34] MILLS J, HU J, MIN G Y. Multi-task federated learning for personalised deep neural networks in edge computing. *IEEE Trans. on Parallel and Distributed Systems*, 2021, 33(3): 630–641.
 - [35] CHEN Y Q, QIN X, WANG J D, et al. Fedhealth: a federated transfer learning framework for wearable healthcare. *IEEE Intelligent Systems*, 2020, 35(4): 83–93.
 - [36] LONG G D, TAN Y, JIANG J, et al. *Federated learning: privacy and incentive*. Cham: Springer International Publishing, 2020.
 - [37] YAN J T, CHEN T, XIE B W, et al. Hierarchical federated learning: architecture, challenges, and its implementation in vehicular networks. *ZTE Communications*, 2023, 21(1): 38–45.
 - [38] YAN J T, CHEN T, SUN Y X, et al. Dynamic scheduling for vehicle-to-vehicle communications enhanced federated learning. *IEEE Trans. on Wireless Communications*, 2025, 24(11): 9373–9390.
 - [39] CHELLAPANDI V P, YUAN L Q, BRINTON C G, et al. Federated learning for connected and automated vehicles: a survey of existing approaches and challenges. *IEEE Trans. on Intelligent Vehicles*, 2023, 9(1): 119–137.
 - [40] ZHANG B N, ZHANG X K, PUN M O, et al. Prototype-based clustered federated learning for semantic segmentation of aerial images. *Proc. of the IEEE International Geoscience and Remote Sensing Symposium*, 2022: 2227–2230.
 - [41] BUYUKTAS B, SUMBUL G, DEMIR B. Learning across decentralized multi-modal remote sensing archives with federated learning. *Proc. of the IEEE International Geoscience and Remote Sensing Symposium*, 2023: 4966–4969.
 - [42] ZHAO H S, SHI J P, QI X J, et al. Pyramid scene parsing network. *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017: 6230–6239.
 - [43] YANG Y, NEWSAM S. Bag-of-visual-words and spatial extensions for land-use classification. *Proc. of the 18th International Conference on Advances in Geographic Information Systems*, 2010: 270–279.
 - [44] CHENG G, HAN J W, LU X Q. Remote sensing image scene classification: benchmark and state of the art. *Proceedings of the IEEE*, 2017, 105(10): 1865–1883.
 - [45] LI Q B, DIAO Y Q, CHEN Q, et al. Federated learning on non-IID data silos: an experimental study. *Proc. of the IEEE 38th International Conference on Data Engineering*, 2022: 965–978.
 - [46] REDDI S, CHARLES Z, ZAHEER M, et al. Adaptive federated optimization. <https://arxiv.org/abs/2003.00295>.
 - [47] HSU T M H, QI H, BROWN M. Measuring the effects of non-identical data distribution for federated visual classification. <https://arxiv.org/abs/1409.06335>.