remote sensing

Article

Federated Learning Approach for Remote Sensing Scene Classification

Belgacem Ben Youssef *, Lamyaa Alhmidi, Yakoub Bazi and Mansour Zuair

Department of Computer Engineering, College of Computer and Information Sciences, King Saud University, P.O. Box 51178, Riyadh 11543, Saudi Arabia; ybazi@ksu.edu.sa (Y.B.)
* Correspondence: bbenyoussef@ksu.edu.sa

Abstract: In classical machine learning algorithms, used in many analysis tasks, the data are centralized for training. That is, both the model and the data are housed within one device. Federated learning (FL), on the other hand, is a machine learning technique that breaks away from this traditional paradigm by allowing multiple devices to collaboratively train a model without each sharing their own data. In a typical FL setting, each device has a local dataset and trains a local model on that dataset. The local models are next aggregated at a central server to produce a global model. The global model is then distributed back to the devices, which update their local models accordingly. This process is repeated until the global model converges. In this article, a FL approach is applied for remote sensing scene classification for the first time. The adopted approach uses three different RS datasets while employing two types of CNN models and two types of Vision Transformer models, namely: EfficientNet-B1, EfficientNet-B3, ViT-Tiny, and ViT-Base. We compare the performance of FL in each model in terms of overall accuracy and undertake additional experiments to assess their robustness when faced with scenarios of dropped clients. Our classification results on test data show that the two considered Transformer models outperform the two models from the CNN family. Furthermore, employing FL with ViT-Base yields the highest accuracy levels even when the number of dropped clients is significant, indicating its high robustness. These promising results point to the notion that FL can be successfully used with ViT models in the classification of RS scenes, whereas CNN models may suffer from overfitting problems.

Keywords: federated learning; remote sensing; scene classification; deep learning; CNN; transformer

check for updates

Citation: Ben Youssef, B.; Alhmidi, L.; Bazi, Y.; Zuair, M. Federated Learning Approach for Remote Sensing Scene Classification. *Remote Sens.* **2024**, *16*, 2194. https://doi.org/10.3390/rs16122194

Academic Editor: Pedro Melo-Pinto

Received: 20 April 2024
Revised: 14 June 2024
Accepted: 14 June 2024
Published: 17 June 2024

1. Introduction

Remote sensing (RS) is a powerful technology that plays an essential role in monitoring and understanding our planet's surface, oceans, and atmosphere. This represents a method that deals with gathering information about the Earth's properties, processes, and changes over time through the collection of data from different distances, usually using satellites or aircraft. The fact that RS can overcome the limitations imposed by traditional data collection methods is a motivating factor for its use [1]. One of the key motivations for using RS is the ability to access remote or inaccessible areas. This would include areas from which there may be difficulty or risk in gathering data, i.e., forests, polar regions, and areas hit by natural disasters [2]. Through remote sensing, researchers and scientists are able to obtain valuable information about these areas, which will allow them to gain a better understanding of their characteristics and dynamics.

RS is utilized in a diverse array of disciplines, encompassing numerous areas of study. For instance, it is employed in environmental monitoring, where it plays a crucial role in evaluating climate change patterns, mapping alterations in land cover, monitoring deforestation, and assessing the overall health of ecosystems. In addition, it plays a pivotal role in resource management by overseeing the utilization of water resources, monitoring agricultural output, and evaluating soil conditions. RS is also widely utilized in disaster management by facilitating the prompt evaluation of impacted regions, tracking the progression of wildfires, and assisting in post-disaster recovery endeavors. It is heavily used in the fields of urban planning, transportation, and infrastructure development. Furthermore, it facilitates archaeological surveys, enables the exploration of natural resources, and aids in climate modeling [3–5].

RS scene classification entails the classification of satellite or aerial images into distinct land-cover or land-use categories with the objective of deriving valuable insights about the Earth's surface. The accurate realization of this task permits the use of various applications such as land-cover mapping, urban planning, environmental monitoring, and natural resource management. Scene classification offers valuable information regarding the spatial distribution and temporal changes in various land-cover categories, such as forests, agricultural fields, water bodies, urban areas, and natural landscapes. This information is vital for decision-making processes concerning land management, disaster response, and sustainable development. Moreover, the application of scene classification in RS plays an important role in conducting change-detection analysis. This allows for the identification and monitoring of temporal variations in land cover over a period of time via the utilization of advanced image analysis techniques to obtain valuable land-cover data to be applied for various purposes including the enhancement of land management practices and the facilitation of well-informed decision making [6].

The impetus for federated learning (FL), which places significant emphasis on safeguarding data privacy, stems from the difficulty of readily sharing data between different entities. Distributed data across multiple devices and organizations pose challenges in terms of their efficient centralization and their proper utilization for machine learning models in various domains. Data privacy concerns and legal restrictions frequently impede the sharing of these data. FL offers a solution by enabling collaborative model training while ensuring that the data remain in their original location without being transferred elsewhere. FL ensures privacy and confidentiality by maintaining decentralized and local data. This approach aims to utilize the collective knowledge stored in distributed data sources, while also ensuring that the privacy rights of data owners are respected. Hence, the provision of FL would ensure that these data owners are complying with different types of regulations. FL facilitates a framework that prioritizes privacy and security, allowing organizations to collaborate and gain insights from each other's data while ensuring that data privacy remains intact. FL currently offers a range of solutions, which include horizontal and vertical federated learning [7,8].

In this regard, horizontal FL is specifically designed for situations in which multiple devices or entities share similar characteristics but have distinct sets of data samples. In this methodology, the models are trained individually on each device using their respective datasets, and subsequently, the model updates are combined to generate a global model. This approach guarantees that the data remain stored on the individual devices, thus addressing concerns related to privacy. At the same time, it enables collaboration and the sharing of knowledge among the devices [7].

On the other hand, vertical FL is well suited for scenarios in which multiple entities possess complementary information regarding the same features. Vertical FL involves the horizontal partitioning of data, where each entity has ownership of a specific subset of features. By jointly training models using their own datasets, the entities can acquire knowledge from the combined data without directly sharing their individual data. FL encompasses other variations such as federated transfer learning, which involves refining pre-trained models using federated methods, and federated reinforcement learning, which expands the concept to reinforcement learning scenarios. In general, the current solutions in FL consist of various methods designed for different situations where data are distributed. These methods allow for the collaborative training of models while also ensuring the privacy and security of the data. The techniques are constantly evolving to meet the requirements of various applications and tackle the difficulties related to decentralized and privacy-preserving machine learning [8].

It is worth recalling that FL is applicable in diverse domains. FL has a notable application in healthcare, specifically in facilitating collaboration among various healthcare providers. This collaboration allows for the training of models using patient data that are distributed across different sources, all while ensuring the preservation of privacy [9]. FL is utilized in various industries, including energy, to optimize smart grids, and in personalized recommendation systems to enhance user experiences [10]. Additionally, FL is highly beneficial in the financial services industry for the purposes of fraud detection and risk assessment. It enables institutions to exchange valuable insights without jeopardizing the confidentiality of sensitive customer data. In the domain of autonomous vehicles [11], FL can be employed to train models using data collected from numerous vehicles, thereby enhancing safety and performance. Furthermore, FL can be implemented in various other domains such as natural language processing (NLP), agriculture, intelligent homes, and scene classification in RS imagery. The flexibility of FL renders it a potent method for collaborative machine learning, all the while guaranteeing the confidentiality and protection of data.

As far as we know, there are no previous studies that specifically utilize FL in the field of RS. By utilizing FL techniques, it is feasible to exploit the combined expertise from dispersed data sources in recommender systems while guaranteeing the confidentiality and protection of the data. Additional research in this domain can enhance the progress of remote sensing by facilitating cooperative machine learning methods that exploit the potential of decentralized data without jeopardizing confidential information.

Overall, the work described herein makes the following two main research contributions:

- Evaluation of using the FL technique in the classification of RS scenes from three different datasets: Optimal-31, UCMerced, and NWPU in conjunction with four deep learning (DL) models. Two of these models, EfficientNet-B1 and EfficientNet-B3, belong to the CNN family while the remaining two, ViT-Tiny and ViT-Base, are part of Vision Transformers. To the best of our knowledge, this may be the first time that FL is being utilized in the context of a RS application.
- Analysis of classification results yielded by the highest performing model (ViT-Base) by considering multiple scenarios of dropped clients. In particular, we focused our examination on assessing the performance of two cases of FL: one with 10 clients and the other with 40 clients. By varying the number of dropped clients, we were able to affirm the observation that selectively dropping clients during training enhances the robustness and performance of the model in FL scenarios.

The rest of the paper is organized as follows: Section 2 gives a short review of some recent works related to scene classification in RS that employ the centralized learning paradigm. Then, Section 3 describes our methodology used in undertaking this research by applying FL in the context of RS scene classification. This is followed by Section 4, where our experimental results are presented and analyzed. Finally, before concluding the paper and outlining our future work, we further discuss our results by considering multiple FL scenarios, where the number of dropped clients is varied, for the highest-performing model among the four considered networks.

2. Related Work

We review in this section some of the recent research works dealing with RS scene classification using centralized methods. In this context, Cheng et al. introduced in [12] a technique for classifying RS scenes using convolutional features. This method extracts depth features and generates visual words. Wang et al. presented the concept of attention by proposing the Attention Recurrent Convolutional Network [13]. The method employs adaptive attention region selection and sequential processing to generate highly accurate predictions. Additionally, the authors develop a recurrent attention framework to compress high-level semantic and spatial characteristics into a few simplex vectors, thereby reducing the number of parameters required for learning. Lu et al. in [14] showed that feature learning and aggregation can improve network performance.

Bazi et al. propose a simple yet effective method for fine-tuning deep convolutional neural networks (CNNs) [15]. This method uses an auxiliary classification loss function to inject gradients into an earlier layer of the network, helping to mitigate the problem of vanishing gradients and improving classification accuracy. The authors demonstrate that their method is efficient in several benchmark datasets. Ji et al. introduce a technique that utilizes the attention mechanism to identify and combine features from multiscale discriminative regions [16]. Guo et al. in [17] disclose a method called Saliency Dual Attention Residual Network to effectively capture both cross-channel and spatial saliency information. Spatial attention is incorporated into low-level features to highlight the location of important information and reduce the influence of irrelevant background information. Channel attention, on the other hand, is applied to high-level features to extract significant information.

Alswayed et al. introduce a deep attention model that utilizes the pre-trained SqueezeNet CNN [18]. A distinct branch is added to the network, which incorporates an attention mechanism and acquires optimal weights for the features learned in the primary branch. The authors in [19] suggest a CNN with an attention mechanism and multiple augmentation schemes to enhance the problem of scene classification. The augmentation operation applied to attention mechanism feature maps serves to compel the model to capture features specific to each class and remove unnecessary information. It also encourages the model to focus on discriminative regions as much as possible, rather than relying solely on global information without any preference.

The study described in [20] introduces an attention-based approach. This method effectively distinguishes the important information from the intricate content of the scene. The approach relies on the DenseNet CNN model as its foundation and is referred to as channel-attention-based DenseNet. DenseNet is capable of extracting spatial features at various scales and establishing correlations between them. A channel attention mechanism is implemented to enhance the weights of significant feature channels and suppress the less important ones.

The authors of the research presented in [21] suggest a dual attention-aware network. They employ two types of attention modules, namely, channel attention and spatial attention. The attention-aware feature representation, which is crucial for enhancing classification performance, is obtained by combining the outputs of two attention modules. The classification network consists of three subnetworks. Each subnetwork is trained using specific scaled regions. The feature outputs of these subnetworks are then combined before the final classification. Xue et al. introduced a technique that utilizes three deep networks to independently extract deep features from the RS image [22]. The features were combined to form a unified feature vector for classification. Further, AlHichri et al. described an improved CNN structure called EfficientNet-B3-Attn, specifically designed for scene classification in RS [23]. This model incorporates an attention mechanism into the pre-trained EfficientNet-B3 CNN, effectively tackling the difficulties related to large datasets and varied scene types.

A method based on Vision Transformers has been proposed by Bazi et al. in [24]. The Transformer model, unlike CNNs, is capable of detecting long-range correlations between patches using an attention module. The proposed method has been assessed using four publicly available remote sensing image datasets, and the test results showed that these new types of networks will improve classification accuracies better than state-of-the-art methods. More recently, Zhao et al. [25] proposed a novel approach using aerial and ground-based dual-view images. They use Dempster–Shafer theory to combine information from both views, overcoming single-view limitations and improving reliability. Chen et al. [26] propose a new model, called BiShuffleNeXt. The model is a lightweight bi-path network combining spatial and spectral paths for improved feature representation. It balances model complexity and classification accuracy, making it suitable for resource-constrained applications. The work in [27] reviews deep learning techniques in RS scene classification and analyzes various architectures and methodologies. It discusses strengths and weaknesses, data augmentation, transfer learning, and fusion strategies. The paper also addresses related challenges like data scarcity and class imbalance and suggests potential solutions and future research directions.

3. Materials and Methods

In the context of centralized learning, the dataset comprises N pairs denoted as $(x_{i,}y_i)$, where $x_{i,}$ represents an image and y_i represents its related label. The index i varies from 1 to N, representing the overall number of images in the dataset. The training process entails optimizing the network parameters by utilizing a loss function, specifically the multiclass entropy loss $L_{cross-entropy}(y,p)$, which is formulated as below:

$$L_{cross-entropy}(y,p) = -\sum_{k=1}^{K} y_k \, log\,(p_k), \quad (1)$$

which is a measure of the difference between the predicted distribution (p) and the true distribution of the classes (y), with K representing the number of classes in the dataset.

Centralized machine learning involves the connection of different clients to a central server as shown in Figure 1, where they can upload their data. From one perspective, centralized training offers computational efficiency for participants by relieving them of computational duties that typically demand significant resources. However, the privacy of users' data is significantly compromised because of the potential for harmful activities or unauthorized access by adversaries on the server. In the context of data uploading, it is important to note that a substantial volume of data can result in increased communication overhead between participants and the server [28].

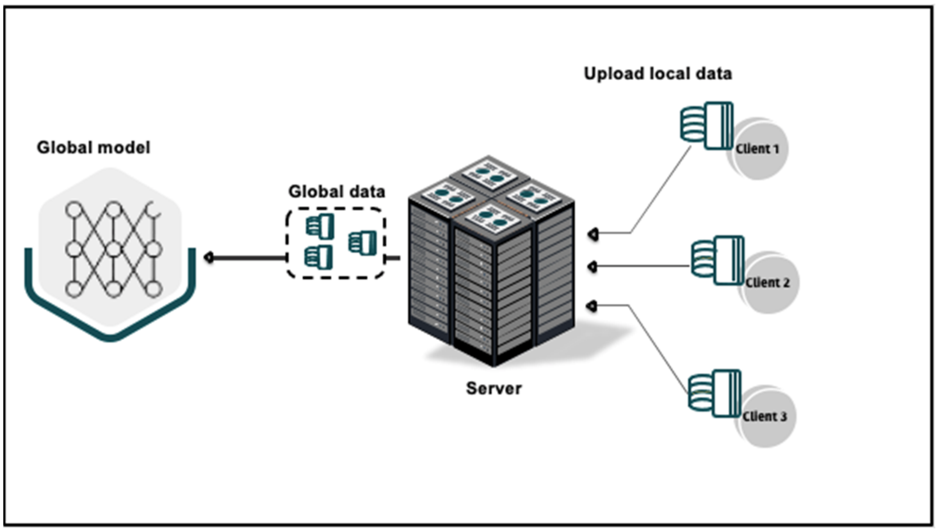


Figure 1. Illustration of the classical centralized machine learning paradigm.

In order to tackle the issue of data privacy manifest in centralized machine learning, FL becomes a viable option. Here, a decentralized approach, in which the training procedure occurs on separate devices or clients, is employed. We will denote these clients as Client 1, Client 2, etc., as shown in Figure 2. Rather than transmitting unprocessed data to a central server, the models undergo local training on each client, utilizing their particular datasets. The models' updates are then transmitted in a safe manner to a central server, where they are consolidated to generate a comprehensive global model. Subsequently, the global model is transmitted to the clients, enabling them to make updates to their respective local models. The utilization of FL effectively addresses problems related to data privacy and security, as it ensures that raw data remain on the client's side, and is not sent to a central server. The decentralized approach employed in this context also serves to decrease communication latency, as it solely transmits model updates. Therefore, FL offers notable advantages in situations where data are of a sensitive nature or dispersed across various locations. Another advantage is observed when there are constraints on connection capacity.

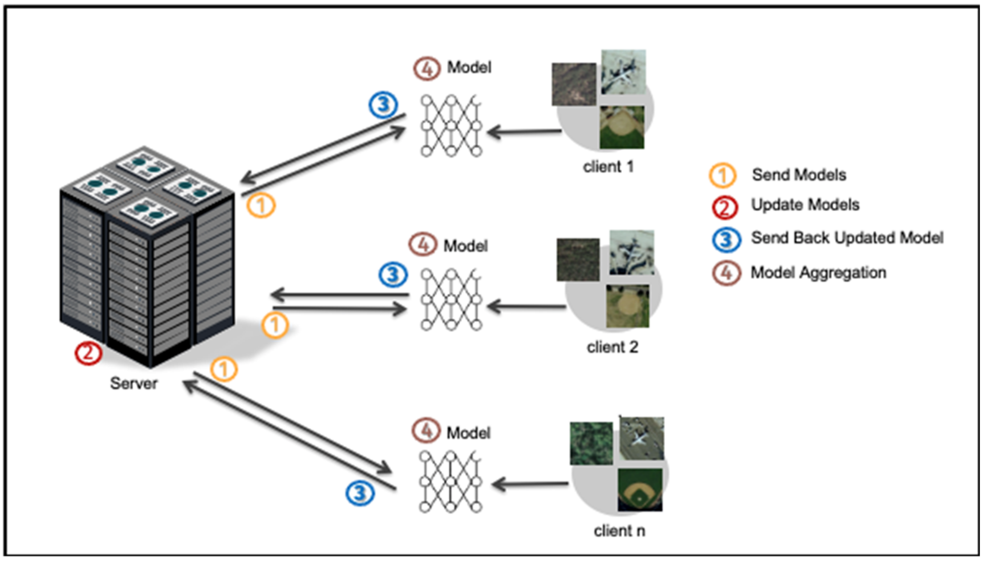


Figure 2. Illustration of federated learning as a machine learning paradigm.

In FL, the global model is updated using the following global loss function L:

$$L = \sum_{i=1}^{m} \frac{N_i}{N}.L_i \tag{2}$$

This is the result of the federated averaging process across multiple (= m) clients, $\frac{N_i}{N}$ is equal to the weight assigned to the contribution of the client i's local model update. The term N_i signifies the number of samples in the local dataset of client i while N represents the whole number of samples across all clients, and L_i refers to the local loss function for client i. The suggested approach entails the initialization of global parameters, the distribution of the model to clients for local training, the aggregation of local updates, and the collaborative iteration. This enables collaborative learning in a decentralized manner, allowing for modification to address privacy concerns or communication limitations in the federated learning environment.

In this work, we plan to implement and contrast two different architectures, one based on CNNs and the other based on Vision Transformers. For the CNN-based architecture, we will utilize two models, namely, EfficientNet-B1 and EfficientNet-B3. Likewise, for the Vision Transformer architecture, we will include two models, namely, ViT-Tiny and ViT-Base. These four models will be applied in the task of RS scene classification in both the centralized and FL paradigms of machine learning across three well-known datasets. We present below a brief synopsis of these DL models.

It is worth recalling that EfficientNet is a model developed by Google [15] that aims to scale up CNNs. It employs a straightforward and highly efficient compound coefficient. The EfficientNet algorithm operates in a distinct manner compared to conventional techniques that adjust the dimensions of networks, including width, depth, and resolution. It achieves this by uniformly scaling each dimension using a predetermined set of scaling coefficients. In practical terms, the enhancement of model performance can be achieved by scaling individual dimensions. However, achieving a balanced distribution of all dimensions within the network, taking into account the available resources, leads to an overall improvement in performance. The effectiveness of model scaling is heavily influenced by the quality of the baseline network. In order to achieve this objective, a novel baseline network is established by employing the AutoML framework, which enhances both accuracy and efficiency. EfficientNet, similar to MobileNetV2 and MnasNet, uses mobile inverted bottleneck convolution (MBConv) as its primary component. Furthermore, the activation function employed by this network is called Swish, which replaces the Rectifier Linear Unit (ReLU) activation function. Figure 3 shows the structure of the EfficientNet-B0 model, used as a baseline.

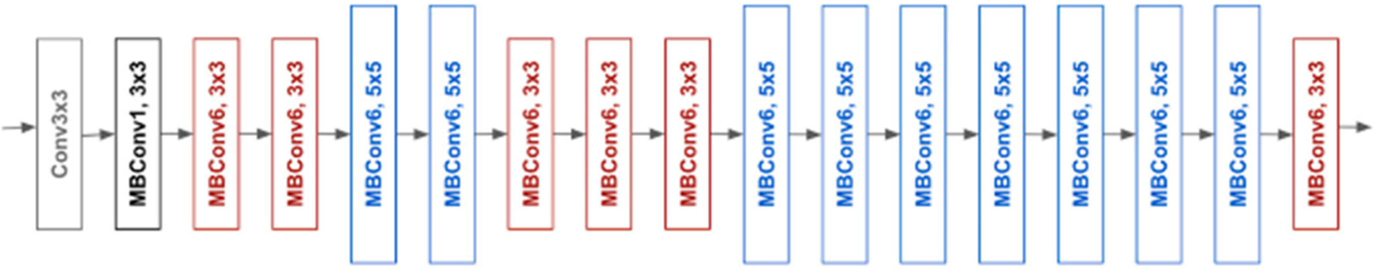


Figure 3. Architecture of the baseline EfficientNet-B0 model.

On the other side, Transformers have recently gained popularity in various domains due to their ability to capture long-range dependencies and effectively process sequential data. We incorporated these Transformer models to explore their potential in image classification tasks and compare them with the CNN-based approach. The Vision Transformer design is derived from the vanilla Transformer architecture [24], which has garnered significant attention in recent years due to its exceptional performance in machine translation and other natural language processing (NLP) tasks. The Transformer model follows an encoder–decoder architecture, enabling concurrent processing of sequential input without the need for a recurrent network. The efficacy of Transformer models has been significantly enhanced by the incorporation of the self-attention mechanism, which is suggested as a means to capture extensive connections among the elements within a sequence.

The suggested Vision Transformer aims to expand the application of the conventional Transformer model to the domain of picture categorization. The primary objective is to extend their applicability to modalities beyond text, while avoiding the incorporation of any data-specific design. The Vision Transformer model employs the encoder module of the Transformer architecture, as seen in Figure 4 below, to carry out classification tasks by associating a series of patches with their corresponding semantic labels. The Vision Transformer employs an attention mechanism that enables it to attend to diverse regions of the image and integrate information throughout the full image, unlike standard CNN architectures that often use filters with a limited receptive field.

The Vision Transformer-based architecture was implemented using two models: ViT-Tiny and ViT-Base. The ViT-Base model has 86 million parameters, with a last feature representation dimension of 768. The ViT-Tiny model, on the other hand, utilized a Transformer variant with 5.7 million parameters. The last feature representation has a dimension of 192. For the EfficientNet-B1 model, we employed a model with 7.8 million parameters with its last feature representation having a dimension of 1280. On the other hand, the utilized EfficientNet-B3 model comprises 12 million parameters with its last feature representation having a dimension of 1408. We provide in Table 1 a summary of the key characteristics of each model employed in our experiments.

Table 1. Characteristics of the four DL models used in our experiments.

DL Model	Feature Representation Size	No. of Parameters
ViT-Base [29]	768	86 M
ViT-Tiny [30]	192	5 M
EfficientNet-B1 [31]	320	7.8 M
EfficientNet-B3 [31]	384	12 M

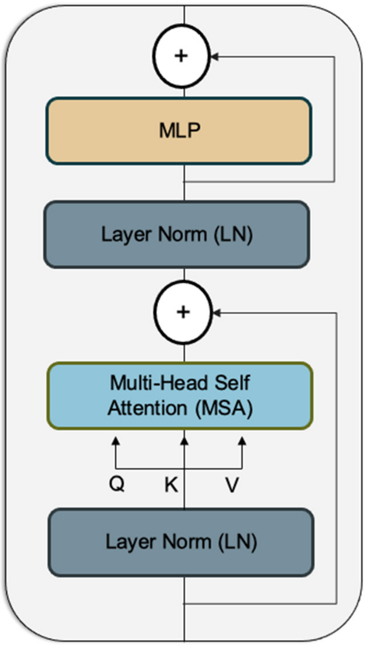


Figure 4. Illustration of the encoder module within the Transformer model.

4. Results

In this section, we outline the experimental work conducted in this study. We first describe the employed RS datasets, followed by detailing the experimental setup and concluding with a discussion of the outcomes from each experiment.

4.1. Dataset Description

To evaluate the utilized scene classification models, we selected three well-known datasets in RS, called Optimal-31 [24], UCMerced [24], and NWPU-RESISC451 [32]. The Optimal-31 dataset was obtained from Google Earth images and includes 31 different scene classes. Each class consists of 60 images with dimensions of 256 × 256 pixels in the RGB color model. The resolution of an image is equal to 0.3 m per pixel. The UCMerced dataset comprises 2100 land-use images that were manually chosen from aerial ortho-imagery. These images are classified into 21 categories. The images were obtained from the United States Geological Survey National Map and then resized to smaller areas. The dataset is extensively utilized for aerial image classification owing to its varied spatial land-use patterns and significantly overlapping classes.

The NWPU-RESISC45 dataset was created by Northwestern Polytechnical University (NWPU) for REmote Sensing Image Scene Classification (RESISC). There are a total of 31,500 images in this dataset. It is composed of 45 classes such as airplane, airport, beach, bridge, forest, and desert. Each class includes 700 RGB images extracted from Google Earth imagery with each image size being equal to 256 × 256 pixels. Within the classes of this dataset, the spatial resolution decreases from 30.0 m to 0.2 m per pixel. In Table 2, we disclose the various characteristics of these three datasets. In addition, we exhibit some examples of images from each dataset in Figure 5.

Table 2. Details of the three RS datasets used in this work.

Dataset Feature	Optimal-31	UCMerced	NWPU
No. of Images	1860	2100	31,500
No. of Classes	31	21	45
Images per Class	60	100	700
Image Size	256 × 256	256 × 256	256 × 256
Resolution	0.3 m/pixel	0.3 m/pixel	0.2–30 m/pixel

4.2. Experimental Setup

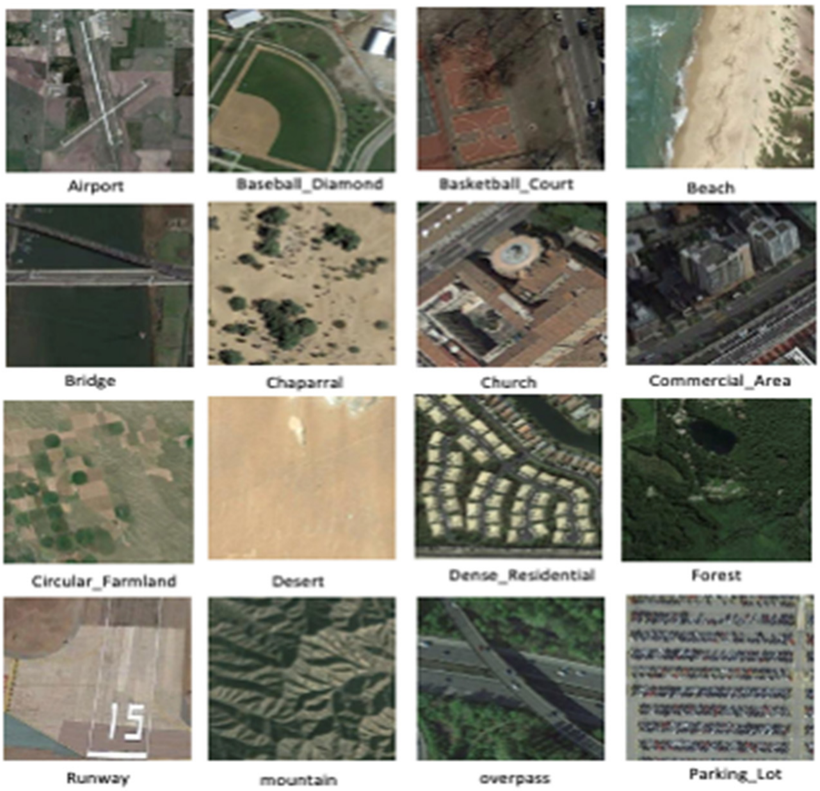
In our experiments, we resized all dataset images to 224 × 224 as is performed by most other works to fit the models' input dimensions. Each dataset was split randomly into two subsets, one used for training and the other for testing. In particular, we conducted experiments with a 50–50 split for UCMerced and Optimal-31 and a 20–80 split for NWPU, where the first value is equal to the percentage of training data while the second one represents the percentage of the testing data. We ran this split randomly five times and then considered the average classification results.

To evaluate the overall performance, we summarized the results using the overall accuracy (*OA*) metric, calculated as the ratio of correctly identified samples to the total number of examined samples [32]. It is defined by the following formula:

$$OA = \frac{\sum_{i=1}^{K} n_{ii}}{|N_{test}|}, \tag{3}$$

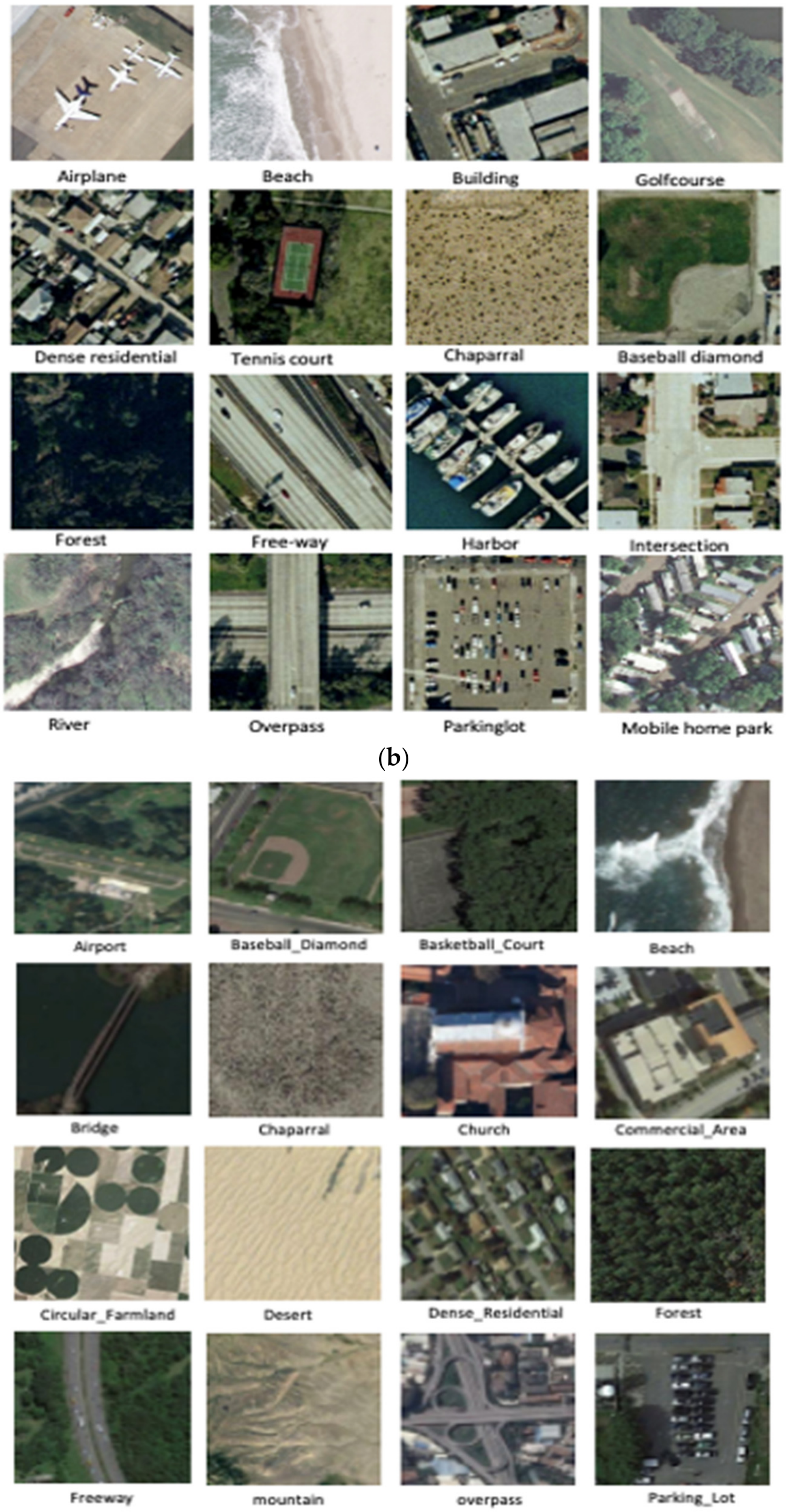
where K is equal to the number of classes and n_{ii} is equal to the count of accurate classifications for class i within the test dataset while the total number of test samples is equal to $\mid N_{test} \mid$.

All our experiments were conducted on the Google Colab environment, utilizing the PyTorch deep learning library written in Python. The AdamW optimization algorithm was utilized with default parameter values. In order to accommodate memory limitations of the computing platform, the model was trained in batches of 16 images at a time in centralized mode and 10 images at a time in federated learning mode. In Table 3, we reveal the different model parameters used in our experiments.



(**a**)

Figure 5. *Cont.*



(**b**)

(**c**)

Figure 5. Some example images from the three RS datasets used in this work: (**a**) Optimal-31, (**b**) UCMerced, and (**c**) NWPU, respectively.

Table 3. Overall model parameters used in both learning modes during the execution of our experiments.

Feature	Centralized Mode	Federated Learning Mode
Optimizer	AdamW	AdamW
DL Models	ViT-(Tiny, Base) EfficientNet (-B1 and -B3)	ViT-(Tiny, Base) EfficientNet (-B1 and -B3)
Train–Test Split	50–50 [1]	50–50 [1]
Trials per Run	5	5
Number of Epochs	20	40
New Image Sizes	224 × 224	224 × 224
Batch Size	16	16

[1] A 20–80 split was used for the NWPU dataset.

4.3. Results of Centralized Mode

We conducted experiments using two different models with each encompassing two variant networks: the Vision Transformer, with the Tiny-Base variants; and CNNs with EfficientNet-B1 and EfficientNet-B3 as the two selected networks. As stated previously, each dataset was split evenly into a 50–50 train–test split, except for the NWPU dataset. For the latter, we adopted a 20–80 split strategy for training and testing, due to the significantly larger size of this dataset. We repeated each experiment five times in a random fashion while running each experiment for 20 epochs. After training all models using the centralized approach, we evaluated each one in terms of overall accuracy (OA) using testing data. The calculated OA values, including their averages across these three datasets, are displayed in Table 4.

Table 4. OA results of the centralized mode for each of the four DL models and across all three RS datasets. The rightmost column holds the average OA values for each DL model over the three datasets. All values are given as percentages (%).

DL Model	RS Datasets			OA Average
	Optimal-31	UCMerced	NWPU	
ViT-Tiny	87.05	96.69	91.08	91.61
ViT-Base	89.46	96.42	90.65	92.18
EfficientNet-B1	87.55	97.70	91.73	92.33
EfficientNet-B3	87.05	95.85	91.69	91.53

4.4. Results of Federated Learning Mode

In this section, we present the results of FL mode, particularly focusing on the behavior of different models across the varying numbers of clients. The latter was chosen from the set {2, 5, 10, 15, 20, 40}. We then compare the results of using the FL mode for each number of clients and compare them with those generated by the centralized mode. In Table 5 and Figure 5 below, we show the OA results for the FL mode using the three different RS datasets for each of the two Transformer models and the two CNN models, respectively. To facilitate the comparison with the results of the centralized mode, we include them again in Table 5 as the leftmost column under numbers of clients.

Analyzing the results reveals interesting trends in model performance as the number of clients increases. In both the Optimal-31 and UCMerced datasets, the Transformer models (ViT-Tiny and ViT-Base) exhibit a reduction in accuracy as the client count increases. However, for ViT-Base, this decrease only takes place up to five clients. In FL mode with a higher number of clients, ViT-Base starts to yield better performance results than in the centralized mode. The observed decrease in accuracy indicates that the FL mode presents difficulties in combining model updates from various clients, leading to a fall in overall accuracy. In contrast, the CNN models, EfficientNet-B1 and EfficientNet-B3, exhibit a more pronounced decline in accuracy with an increasing number of clients in FL mode, while accuracy increases with a decreasing number of clients in comparison to centralized mode.

This behavior could suggest that the CNN models are more sensitive to variations in client data and struggle to maintain accuracy when faced with a larger number of clients.

Table 5. OA results of the FL mode for each of the three datasets and across all four DL models. In FL mode, the number of clients is varied from 2 to 40. For the sake of completeness, we also include the results of the centralized mode. All values are given as percentages (%).

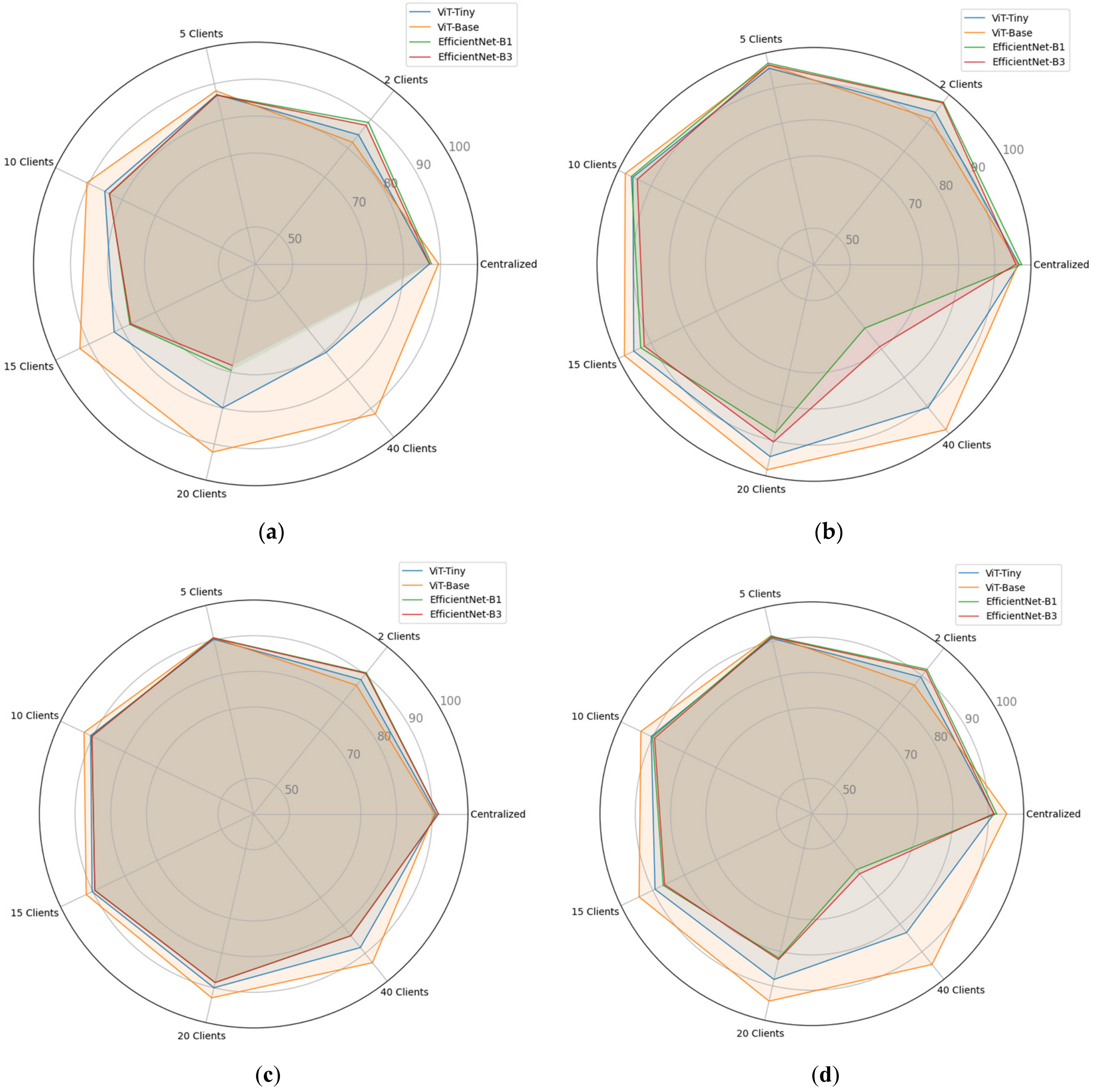
Dataset	Train–Test Split	Model Type	DL Model	Number of Clients						
				Centralized	2	5	10	15	20	40
Optimal-31	50–50	Vision Transformer	ViT-Tiny	87.05	84.67	87.01	85.23	82.41	79.96	70.60
			ViT-Base	89.46	82.09	88.04	90.60	92.69	92.22	91.94
		CNN	EfficientNet-B1	87.55	88.95	86.86	83.76	77.72	69.57	34.69
			EfficientNet-B3	87.05	87.96	86.82	83.87	77.40	68.19	32.43
UCMerced	50–50	Vision Transformer	ViT-Tiny	96.69	93.83	95.66	95.58	95.31	94.53	90.65
			ViT-Base	96.42	91.58	96.57	97.92	98.17	98.27	98.50
		CNN	EfficientNet-B1	97.40	97.41	97.10	96.06	93.18	87.81	62.55
			EfficientNet-B3	95.85	97.14	96.57	94.23	92.08	90.38	69.14
NWPU	20–80	Vision Transformer	ViT-Tiny	91.08	88.19	90.25	90.68	90.20	89.98	87.90
			ViT-Base	90.65	86.16	90.72	92.71	92.05	92.87	93.34
		CNN	EfficientNet-B1	91.73	90.64	90.70	90.34	89.33	88.46	83.63
			EfficientNet-B3	91.69	90.40	90.61	90.21	89.41	88.48	83.69

Furthermore, within the NWPU dataset, the accuracy of the Vision Transformer models, specifically ViT-Tiny and ViT-Base, exhibits a consistent relative level even when the number of clients is increased. This behavior demonstrates that these vision models have greater resilience in managing federated learning scenarios involving a higher number of clients. The CNN models, EfficientNet-B1 and EfficientNet-B3, exhibit a slight decrease in overall accuracy, but the decline is less pronounced compared to the Optimal-31 and UCMerced datasets. This is due to the different sizes of these datasets whereby the NWPU dataset is much larger than the first two datasets. In addition, we computed the average OA results over the three datasets for the four considered deep learning models in both centralized and FL modes. These values are presented in Table 6. We also exhibit the visualizations of these results in the form of radar charts in Figure 6.

Table 6. Average results in terms of OA across all three RS datasets in FL mode with varying numbers of clients. For the sake of completeness, we also provide the average OA values of the centralized mode.

Model Type	DL Model	Number of Clients						
		Centralized	2	5	10	15	20	40
Vision Transformer	ViT-Tiny	91.61	89.50	90.97	90.49	89.30	88.15	83.05
	ViT-Base	92.18	86.61	91.77	93.74	94.30	94.45	94.59
CNN	EfficientNet-B1	92.33	92.33	91.55	90.05	86.74	81.94	60.29
	EfficientNet-B3	91.53	91.83	91.33	89.43	86.29	82.35	61.75

Upon analyzing the average results of the three datasets, as shown in Table 6 and Figure 6, it becomes apparent that the performance of all models consistently decreases as the number of clients in the FL mode increases. Except that this behavior starts to move in the opposite direction for the ViT-Base model, starting with using ten clients in FL mode. Specifically, the average OA values for ViT-Base are greater than or equal to 94.30% when the number of clients is equal to 15, 20, and 40 clients. In addition, the ViT-Base and ViT-Tiny models regularly exhibit on average superior accuracy in comparison to the CNN models starting when the number of clients is ≥10. This implies that Vision Transformer models have the potential to perform effectively in distributed environments.



(**a**) (**b**)

(**c**) (**d**)

Figure 6. Radar charts depicting the OA results of each of the four employed DL models for the (**a**) Optimal-31 dataset, (**b**) UCMerced dataset, (**c**) NWPU dataset, and (**d**) average across these three datasets.

These observations emphasize the impact of the number of clients on model behavior in the FL mode. As the client count grows, the aggregation of model updates becomes more challenging, perhaps leading to a decline in accuracy except for the ViT-Base model. However, different models exhibit varying levels of resilience to this challenge. In FL, vision models, due to their capacity to capture global patterns with a larger number of clients, typically outperform CNN models.

5. Discussion

In this experimental methodology, we propose an efficient solution for enhancing the results of FL. To this end, a client-dropping mechanism is implemented as shown in Figure 7, wherein one or more clients deliberately refrain from transmitting updates to the server. The objective of employing this methodology is to augment the resilience of the model.

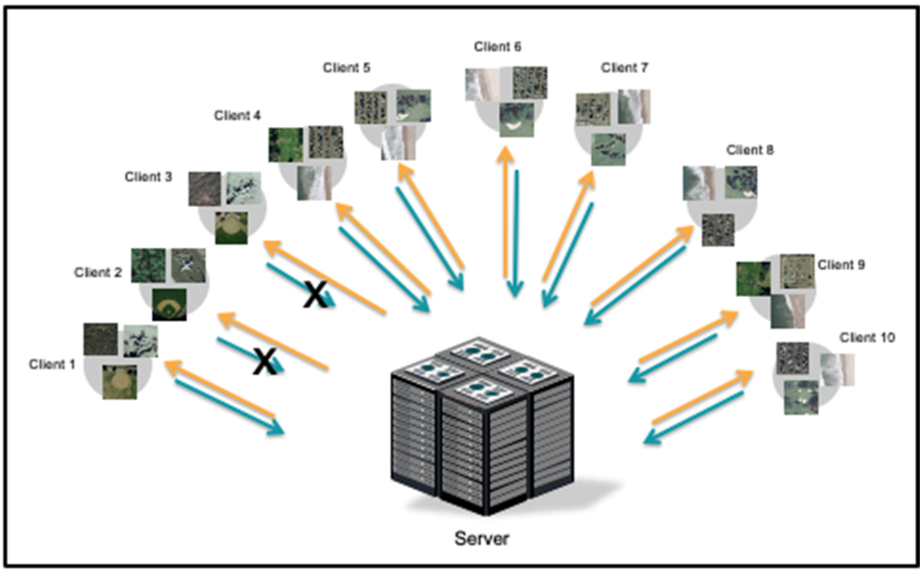


Figure 7. Illustration of the used client-dropping mechanism during training in FL mode.

The experimental procedure entails the utilization of a sample size of 10 clients, with the selection of clients to be eliminated being conducted in a random manner during an epoch. Through the implementation of random client dropping, our objective is to analyze the model's performance across various circumstances and measure its capacity to manage missing updates from individual clients. The use of this technique in FL has demonstrated promising outcomes, as evidenced by the results presented in Table 7 below. We note that the inclusion of client dropout throughout the training phase results in enhanced model accuracy. When doing a comparison between the ViT-Base model and the ViT-Base with Dropped-Clients model, it becomes evident that the latter model exhibits superior accuracy values when excluding two clients in Optimal-31 and UC-Merced datasets while the accuracy in NWPU decreases due to the size of the dataset. We think that this is primarily due to the loss of a significant amount of data engendered by the dropped clients.

Table 7. OA results of the ViT-Base model in FL mode with 10 clients and a varying number of dropped clients for the three RS datasets. For the purpose of comparison, we also include the results of the centralized mode.

RS	Centralized Mode	Number of Dropped Clients					
Dataset		0	2	3	5	6	8
Optimal-31	89.46	90.60	91.01	90.34	89.85	88.39	79.35
UCMerced	96.42	97.92	98.27	98.08	96.95	96.38	91.49
NWPU	90.65	92.71	90.46	90.34	89.32	88.63	85.63

In order to verify the accuracy of our experimental findings, we carried out additional training using ViT-Base in FL mode with 40 clients while employing the same variation in the number of dropped clients. The generated performance results from the testing set are presented in Table 8.

Table 8. OA results of the ViT-Base model in FL mode with 40 clients and a varying number of dropped clients for the three RS datasets. For the purpose of comparison, we also include the results of the centralized mode.

RS	Centralized Mode	Number of Dropped Clients					
Dataset		0	2	3	5	6	8
Optimal-31	89.46	91.94	91.05	92.82	91.72	91.03	88.97
UCMerced	96.42	98.50	98.38	98.36	98.32	97.89	96.93
NWPU	90.65	93.34	93.22	98.87	92.59	92.07	89.88

To affirm our earlier findings when the training of 10 clients was applied, we observe once again that the ViT-Base model with dropped clients consistently outperforms the model in terms of overall accuracy across datasets with small sizes. This further confirms that our new technique of selectively dropping clients during training enhances the robustness and performance of the model in federated learning scenarios. Such results align with our prior experimentation, demonstrating the efficacy and dependability of the suggested method. The enhanced performance found in the dropped clients model provides more support for its capacity to improve the accuracy and robustness of the model in the context of FL paradigms.

The results confirm the efficacy of the proposed technique and emphasize its capacity to improve the precision and robustness of models in FL situations. Further analysis and experimentation could provide deeper insights into the specific mechanisms behind this improvement as well as potentially guide future advancements in the field of federated learning.

6. Conclusions

FL is a cutting-edge machine learning paradigm that has the potential to transform how we train models in a privacy- and security-conscious world. In this article, we describe our research work involving the use of FL in the classification of RS scenes. Four deep models belonging to Transformer- and CNN-based architectures are utilized, including ViT-Tiny, ViT-Base, EfficientNet-B1, and EfficientNet-B3 networks. In addition to the centralized mode, we considered the FL paradigm by varying the number of clients from 2 to 40 and examined the classification performance of these models in terms of overall accuracy. For a number of clients greater than or equal to ten, the two Transformer-based models, and especially ViT-Base, outperform the two CNN-based ones. On the other hand, the latter two models exhibit competitive performance for a small number of clients.

To ascertain further the results generated by the ViT-Base model, we conducted additional experiments to examine its effectiveness and robustness when it is trained in different FL contexts while facing a varied number of dropped clients. That is, we tested the classification performance of ViT-Base by implementing a client-dropping mechanism for two FL scenarios: one with ten clients and a second one with 40 clients. By varying the number of dropped clients in each scenario, we obtained results that attest to the high resiliency and robustness of ViT-Base, particularly when the number of dropped clients is in the range from two to five. For relatively small datasets, such as Optimal-31 and UCMerced, training ViT-Base in FL with dropped clients generated higher classification results than without any dropped-out clients. These results could point toward a new methodology of training DL models in the FL paradigm to enhance their robustness and classification performance. As part of our future work in this realm, we plan to employ FL in other RS tasks such as segmentation and object detection as well as in other application domains, where concerns about the security and privacy of clients' data are more prominent. Another possible research direction is to explore alternative dropout mechanisms, such as those based on data quality rather than random dropout. This could potentially offer another solution to enhance classification results.

Author Contributions: Conceptualization, B.B.Y. and Y.B.; methodology, B.B.Y. and Y.B.; software, B.B.Y., Y.B. and L.A.; validation, B.B.Y., Y.B., L.A. and M.Z.; investigation, B.B.Y., Y.B. and L.A.; resources, M.Z.; writing—original draft preparation, B.B.Y., Y.B. and L.A.; writing—review and editing, B.B.Y., Y.B. and M.Z.; supervision, B.B.Y. and Y.B.; funding acquisition, M.Z. All authors have read and agreed to the published version of the manuscript.

Funding: Researchers Supporting Project number (RSPD2024R607), King Saud University, Riyadh, Saudi Arabia.

Data Availability Statement: The remote sensing datasets employed in this study: Optimal-31, UCMerced, and NWPU, are publicly available.

Acknowledgments: This research was supported by the Researchers Supporting Project number (RSPD2024R607), King Saud University, Riyadh, Saudi Arabia.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Yuan, Q.; Shen, H.; Li, T.; Li, Z.; Li, S.; Jiang, Y.; Xu, H.; Tan, W.; Yang, Q.; Wang, J.; et al. Deep learning in environmental remote sensing: Achievements and challenges. *Remote Sens. Environ.* **2020**, *241*, 111716. [CrossRef]
2. Gao, Y.; Skutsch, M.; Paneque-Gálvez, J.; Ghilardi, A. Remote sensing of forest degradation: A review. *Environ. Res. Lett.* **2020**, *15*, 103001. [CrossRef]
3. Rane, N.L.; Choudhary, S.P.; Giduturi, M.; Pande, C.B. Remote Sensing (RS) and Geographical Information System (GIS) as A Powerful Tool for Agriculture Applications: Efficiency and Capability in Agricultural Crop Management. *Int. J. Innov. Sci. Res. Technol.* **2023**, *8*, 264–274. [CrossRef]
4. Sensing, R.; Change, L.C.; Areas, U.; Detection, C. Remote Sensing-Based Urban Land Use/Land Cover Change Detection and Monitoring. *J. Remote Sens. GIS* **2017**, *6*, 2. [CrossRef]
5. Ibrahim, G.R.F.; Rasul, A.; Abdullah, H. Improving Crop Classification Accuracy with Integrated Sentinel-1 and Sentinel-2 Data: A Case Study of Barley and Wheat. *J. Geovis. Spat. Anal.* **2023**, *7*, 22. [CrossRef]
6. Cheng, G.; Xie, X.; Han, J.; Guo, L.; Xia, G.-S. Remote Sensing Image Scene Classification Meets Deep Learning: Challenges, Methods, Benchmarks, and Opportunities. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 3735–3756. [CrossRef]
7. Li, L.; Fan, Y.; Tse, M.; Lin, K.-Y. A review of applications in federated learning. *Comput. Ind. Eng.* **2020**, *149*, 106854. [CrossRef]
8. Aledhari, M.; Razzak, R.; Parizi, R.M.; Saeed, F. Federated Learning: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Access* **2020**, *8*, 140699–140725. [CrossRef]
9. Ali, M.; Naeem, F.; Tariq, M.; Kaddoum, G. Federated Learning for Privacy Preservation in Smart Healthcare Systems: A Comprehensive Survey. *EEE J. Biomed. Health Inform.* **2022**, *27*, 778–789. Available online: http://arxiv.org/abs/2203.09702 (accessed on 29 November 2023). [CrossRef]
10. Pham, Q.V.; Dev, K.; Maddikunta, P.K.; Gadekallu, T.R.; Huynh-The, T. Fusion of Federated Learning and Industrial Internet of Things: A Survey. *arXiv* **2021**, arXiv:2101.00798. Available online: http://arxiv.org/abs/2101.00798 (accessed on 29 November 2023).
11. Pokhrel, S.R.; Choi, J. Federated Learning with Blockchain for Autonomous Vehicles: Analysis and Design Challenges. *IEEE Trans. Commun.* **2020**, *68*, 4734–4746. [CrossRef]
12. Cheng, G.; Li, Z.; Yao, X.; Guo, L.; Wei, Z. Remote Sensing Image Scene Classification Using Bag of Convolutional Features. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 1735–1739. [CrossRef]
13. Wang, Q.; Liu, S.; Chanussot, J.; Li, X. Scene Classification with Recurrent Attention of VHR Remote Sensing Images. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 1155–1167. [CrossRef]
14. Lu, X.; Sun, H.; Zheng, X. A Feature Aggregation Convolutional Neural Network for Remote Sensing Scene Classification. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 7894–7906. [CrossRef]
15. Bazi, Y.; Al Rahhal, M.M.; Alhichri, H.; Alajlan, N. Simple Yet Effective Fine-Tuning of Deep CNNs Using an Auxiliary Classification Loss for Remote Sensing Scene Classification. *Remote Sens.* **2019**, *11*, 2908. [CrossRef]
16. Ji, J.; Zhang, T.; Jiang, L.; Zhong, W.; Xiong, H. Combining Multilevel Features for Remote Sensing Image Scene Classification with Attention Model. *IEEE Geosci. Remote Sens. Lett.* **2020**, *17*, 1647–1651. [CrossRef]
17. Guo, D.; Xia, Y.; Luo, X. Scene Classification of Remote Sensing Images Based on Saliency Dual Attention Residual Network. *IEEE Access* **2020**, *8*, 6344–6357. [CrossRef]
18. Alswayed, A.S.; Alhichri, H.S.; Bazi, Y. SqueezeNet with Attention for Remote Sensing Scene Classification. In Proceedings of the 2020 3rd International Conference on Computer Applications & Information Security (ICCAIS), Riyadh, Saudi Arabia, 19–21 March 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1–4. [CrossRef]
19. Li, F.; Feng, R.; Han, W.; Wang, L. An Augmentation Attention Mechanism for High-Spatial-Resolution Remote Sensing Image Scene Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 3862–3878. [CrossRef]
20. Tong, W.; Chen, W.; Han, W.; Li, X.; Wang, L. Channel-Attention-Based DenseNet Network for Remote Sensing Image Scene Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 4121–4132. [CrossRef]

21. Gao, Y.; Shi, J.; Li, J.; Wang, R. Remote Sensing Scene Classification with Dual Attention-Aware Network. In Proceedings of the 2020 IEEE 5th International Conference on Image, Vision and Computing (ICIVC), Beijing, China, 10–12 July 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 171–175. [CrossRef]
22. Xue, W.; Dai, X.; Liu, L. Remote Sensing Scene Classification Based on Multi-Structure Deep Features Fusion. *IEEE Access* **2020**, *8*, 28746–28755. [CrossRef]
23. Alhichri, H.; Alswayed, A.S.; Bazi, Y.; Ammour, N.; Alajlan, N.A. Classification of Remote Sensing Images Using EfficientNet-B3 CNN Model with Attention. *IEEE Access* **2021**, *9*, 14078–14094. [CrossRef]
24. Bazi, Y.; Bashmal, L.; Rahhal, M.M.A.; Dayil, R.A.; Ajlan, N.A. Vision Transformers for Remote Sensing Image Classification. *Remote Sens.* **2021**, *13*, 516. [CrossRef]
25. Zhao, K.; Gao, Q.; Hao, S.; Sun, J.; Zhou, L. Credible Remote Sensing Scene Classification Using Evidential Fusion on Aerial-Ground Dual-View Images. *Remote Sens.* **2023**, *15*, 1546. [CrossRef]
26. Chen, Z.; Yang, J.; Feng, Z.; Chen, L.; Li, L. BiShuffleNeXt: A lightweight bi-path network for remote sensing scene classification. *Measurement* **2023**, *209*, 112537. [CrossRef]
27. Kumari, M.; Kaul, A. Deep learning techniques for remote sensing image scene classification: A comprehensive review, current challenges, and future directions. *Concurr. Comput.* **2023**, *35*, e7733. [CrossRef]
28. Asad, M.; Moustafa, A.; Ito, T. Federated Learning Versus Classical Machine Learning: A Convergence Comparison. *arXiv* **2021**, arXiv:2107.10976. [CrossRef]
29. Dosovitskiy, A.; Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; et al. An Image is Worth 16 × 16 Words: Transformers for Image Recognition at Scale. *arXiv* **2021**, arXiv:2010.11929. Available online: http://arxiv.org/abs/2010.11929 (accessed on 29 February 2024).
30. Ren, S.; Wei, F.; Zhang, Z.; Hu, H. TinyMIM: An Empirical Study of Distilling MIM Pre-trained Models. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 17–24 June 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 3687–3697. [CrossRef]
31. Tan, M.; Le, Q.V. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *arXiv* **2020**, arXiv:1905.11946. Available online: http://arxiv.org/abs/1905.11946 (accessed on 29 February 2024).
32. Alosaimi, N.; Alhichri, H.; Bazi, Y.; Youssef, B.B.; Alajlan, N. Self-supervised learning for remote sensing scene classification under the few shot scenario. *Sci. Rep.* **2023**, *13*, 433. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.