Federated Learning for Remote Sensing Image Classification Using Sparse Image Representations

Christina Kopidaki, Grigorios Tsagkatakis, and Panagiotis Tsakalides, *Member, IEEE*

***Abstract*— The increasing scale and complexity of remote sensing (RS) observations demand distributed processing to effectively manage the vast volumes of data generated. However, distributed processing presents significant challenges, including bandwidth limitations, high latency, and privacy concerns, especially when transmitting high-resolution images. To address these issues, we propose a novel scheme leveraging the encoder of a masked autoencoder (MAE) to generate associated embedding (CLS tokens) from masked images, which enables training deep learning models under federated learning (FL) scenarios. This approach enables the transmission of compact image patches instead of full images to processing nodes, drastically reducing bandwidth usage. On the processing nodes, classifiers are trained with the CLS tokens, and model weights are aggregated using FedAvg and FedProx FL algorithms. Experimental results on benchmark datasets demonstrate that the proposed approach significantly reduces data transmission requirements while maintaining and even surpassing the accuracy of systems with access to full data.**

***Index Terms*— Federated learning (FL), masked autoencoders (MAE), remote sensing (RS).**

I. Introduction

REMOTE sensing technologies have become essential for monitoring and understanding the Earth's surface, allowing detailed analysis through large-scale images captured by satellites and drones. However, the immense data volume generated presents significant challenges. Traditional centralized processing methods are increasingly impractical due to the computational burden on servers, along with security and privacy concerns related to centralizing sensitive geospatial data. This highlights the need for efficient and scalable solutions to maximize the potential of remote sensing (RS) data [1].

Federated learning (FL) is a method that enables independent organizations to collaboratively train a machine learning model without exchanging raw data [2]. Instead, models are trained locally on individual devices, and only model updates are shared and aggregated. This approach reduces bandwidth usage by retaining data on local devices, which, when combined with additional security measures, can also enhance privacy protection.

Received 31 January 2025; revised 14 March 2025; accepted 23 March 2025. Date of publication 3 April 2025; date of current version 28 April 2025. This work was supported by the TITAN ERA Chair Project through the Horizon Europe Framework Program of European Commission under Contract 101086741. *(Corresponding author: Grigorios Tsagkatakis.)*

The authors are with the Institute of Computer Science, Foundation for Research and Technology - Hellas (FORTH) and the Department of Computer Science, University of Crete, 70013 Heraklion, Greece (e-mail: csd4656@csd.uoc.gr; greg@ics.forth.gr; tsakalid@ics.forth.gr).

Digital Object Identifier 10.1109/LGRS.2025.3557579

FL has recently been investigated for applications in RS data analysis [3], [4]. A systematic review in [3] examined the integration of FL in RS image classification, focusing on its ability to address challenges related to decentralized and unshared data archives. The review highlighted key advances in FL methodologies for RS, emphasizing privacy preservation, scalability, and the ability to work with heterogeneous data. In [4], the convergence between FL and RS was explored through extensive experimentation to evaluate the influence of heterogeneous and disjoint data among collaborating clients. This study assessed scalability for increasing numbers of clients, resilience against Byzantine attacks, and the overall efficiency of FL-based RS applications, offering insight and future directions for this emerging paradigm. An FL-based scheme for the multilabel classification of RS images was investigated in [5]. This study introduced a transformer-based FL framework, demonstrating its effectiveness in handling complex multilabel classification tasks while ensuring data privacy and reducing communication overhead.

The impact of the imbalanced class distribution (non-IID data) was also recently explored in FedPM and Fed-PHC [6], [7], which introduced novel strategies to address the challenges arising from heterogeneous data distributions between federated clients. These methods demonstrated improved model convergence and robustness, particularly in decentralized learning environments with significant variability in client data. Meanwhile, privacy-preserving approaches have been a key focus in FL, with works such as [8], which propose innovative techniques to safeguard sensitive client information during training, including personalized privacy mechanisms and secure aggregation protocols. In addition, advancements in multimodal FL were highlighted in [9], where a novel framework was introduced to address the challenge of decentralized multimodal RS image archives.

Although FL offers significant advantages, such as preserving data privacy and distributing computational demands, it also faces challenges when training models with large, high-resolution RS images. The high resolution and extensive geographic coverage of these images generate significant data volumes, which require substantial bandwidth to transmit them from a central server to edge devices [3], [4]. This can lead to network congestion, strain resources, and degrade other network-dependent services. In addition, the uneven distribution of class examples across devices poses challenges for effectively training classifiers with unbalanced data [6], [7].

To address these challenges, this work introduces input sparsity within the FL paradigm to improve the efficiency of distributed training in RS applications. Instead of transferring entire high-resolution images, we propose sending only relevant image patches. This significantly reduces data volume, eases bandwidth demands, and accelerates data transfer. Using masked autoencoders (MAEs) [10], edge devices generate [CLS] tokens from masked images, enabling efficient encoding and transmission of essential image parts. This approach optimizes data transfer, preserves critical classification information, and enhances security by masking a large portion of each image.

The impact of input and network sparsity was explored in [11], demonstrating that sparsity techniques can maintain or even enhance model performance under resource-constrained conditions. Building on this foundation, our work advances the state of the art by investigating the integration of FL with input sparsity, a novel approach tailored to address the unique challenges of distributed training in RS applications. Furthermore, unlike existing approaches, the proposed scheme allows for a separation between processing nodes and storage nodes, enabling greater flexibility in system design. This decoupling facilitates local model training while significantly reducing data transfer volumes, effectively addressing key challenges of centralized data processing such as high bandwidth usage, latency, and privacy concerns. The key contributions are given as follows.

1) We introduce an FL approach that leverages sparsity to minimize bandwidth usage and reduce the need for transmitting full images. While this method provides certain privacy benefits, additional security measures, such as differential privacy [12] or secure aggregation [13], may be necessary to ensure stronger privacy guarantees.
2) We demonstrate that reducing the number of patches for feature extraction can maintain and, in some cases, even improve classification accuracy.
3) We highlight the resilience of the proposed approach in handling class imbalance among clients across diverse decentralization scenarios.

II. Methodology

A. Overview

The proposed framework, as depicted in Fig. 1, consists of three main components: a data distribution server, k clients, and a central parameter server. The data distribution server provides data to the clients, each of whom has a local deep learning model consisting of a vision transformer (ViT)-based encoder and a small number of fully connected layers of multilayer perceptron (MLP). The model is trained in an FL setup where the role of the central parameter server is to aggregate client model weights for producing a global model.

Instead of transferring entire RS images, the data distribution server sends patches to clients based on their needs. Each client processes these masked image patches through the MAE encoder to generate appropriate embeddings (CLS tokens), which are used throughout the training process. The central server initially provides global model parameters to a subset of clients. These clients independently train their local models using private data, tailoring the parameters to their specific datasets. After training, the clients send their updated parameters back to the server, which consolidates them to refine the global model. This collaborative, iterative process is repeated over several rounds, with each iteration further improving the global model's performance.

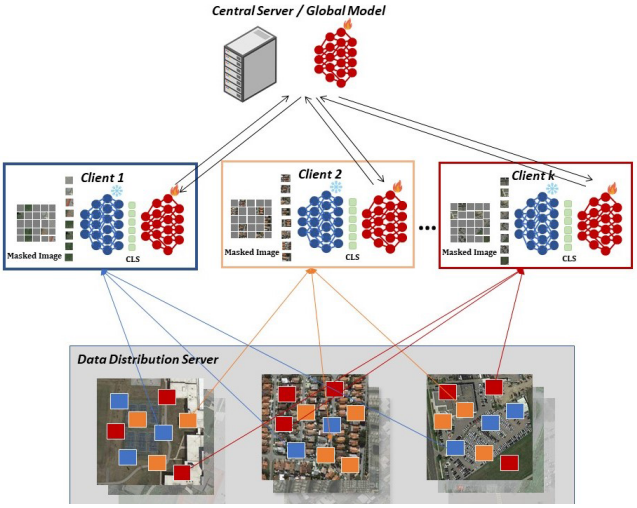

Fig. 1. Diagram of the proposed architecture, illustrated with Client 1 as an example: i) Client 1 randomly selects a set of image patches with a specified masking ratio from the data distribution server; ii) encoder of the MAE architecture processes these patches to generate a CLS token for each image; iii) after receiving initial weights from the central server, the client trains the model locally using the CLS tokens as input; and iv) client then sends its updated weights to the central server, where the weights from all local models are aggregated. Steps iii) and iv) are repeated over multiple rounds until the model converges.

B. Problem Definition

We assume a set of K clients, where each client $k \in \{1, 2, \ldots, K\}$ holds local data D_k, consisting of masked images. Let n_k denote the number of data points (i.e., image patches) at client k and n represent the total number of data points across all clients, with $n = \sum_{n=1}^{K} n_k$. The objective is to minimize the global loss function

$$\min_{w} \sum_{k=1}^{K} \frac{n_k}{n} L_k(w) \tag{1}$$

where $L_k(w)$ is the local loss function for client k. For the case of classification explored in this letter, we consider the categorical cross-entropy as the loss function.

C. Training Process

For training the global model, we employed two state-of-the-art algorithms: FedAvg and FedProx. FedAvg [14] is directly derived from the objective outlined in (1), using stochastic gradient descent (SGD) to learn the parameters w. FedProx [15], on the other hand, can be seen as a generalization and reparameterization of FedAvg. It introduces a proximal term (2) in the local objective function, which helps

address the system heterogeneity that arises in FL

$$\min_{w} h_k(w; w^t) = L_k(w) + \frac{\mu}{2}\|w - w^{(t)}\|^2. \quad (2)$$

Our global classification module is built as an MLP with two hidden layers. The input to this MLP is a 768-shape vector, corresponding to the CLS token generated by the ViT. Each hidden layer contains 1000 nodes, offering significant capacity to learn complex patterns from the input data. The output layer is tailored to the number of classes in our classification task, ensuring that the model is set up to make accurate predictions.

D. Feature Extraction/Input Sparsity

While previous studies have focused on improving aggregation strategies to improve convergence rates [16], [17] and reduce communication costs in FL by introducing *model* sparsity [18], [19], *input* sparsity has not been explored in the context of FL. In our approach, the data distribution server introduces the sparsity of input, sending a subset of image patches from each training image. These patches are processed by each client using the encoder part of the MAE. MAEs [10] utilize ViTs [20], trained to predict pixel values for a large portion of masked image patches. We use pretrained weights from the self-supervised training of the MAE model on ImageNet-1K2, which enables ViTs to achieve superior performance compared to traditional supervised pretraining after fine-tuning. Unlike previous work that focused on image reconstruction, our goal is image classification in an FL framework, and therefore, we only use the encoder and the CLS tokens that it generates without utilizing the MAE decoder.

E. Data Heterogeneity

In FL, the way data are distributed among clients is essential for the effectiveness and efficiency of the learning process [21]. This distribution is a key factor, as clients typically have non-IID data, resulting in diverse and sometimes imbalanced data distributions. To assess the impact of sparse input in realistic scenarios, we performed FL in decentralized and nonshared training sets, defining clients under three different decentralization scenarios as follows.

1) *Decentralization Scenario 1 (DS1):* Balanced sampling. Data for each class are evenly distributed among all clients, ensuring that each client receives an equal proportion of data from every class.
2) *Decentralization Scenario 2 (DS2):* Skewed sampling. Each client is given 80% of the data from two randomly selected primary classes (which vary for each client) and 20% of the data from the remaining classes.
3) *Decentralization Scenario 3 (DS3):* Highly skewed sampling. Each client is assigned data exclusively from five classes, with no data from the other classes.

The design of the decentralization scenarios results in varying levels of heterogeneity of training data between clients. DS1 exhibits the lowest level of heterogeneity, as the images of different classes are evenly distributed among the clients. In contrast, DS3 demonstrates the highest level of heterogeneity due to the lack of training data from most classes in each client's dataset.

TABLE I
APPROXIMATE SIZE OF DATASETS FOR RAW IMAGES WITHOUT ANY COMPRESSION APPLIED (R45 STANDS FOR RESISC45)

	Masking ratio	0.0	0.3	0.6	0.9
AID	Masked Images Size (in GB)	12.240	8.568	4.896	1.224
R45	Masked Images Size (in GB)	7.078	4.954	2.831	0.708

III. RESULTS

A. Datasets

The datasets employed to evaluate the performance of our approach are the aerial image dataset (AID) [22] and the NWPU-RESISC45 (RESISC45) [23]. These datasets are widely used in RS image scene classification for their diversity in spatial resolution, geographic coverage, and seasonal variations, making them essential resources for advancing and assessing scene classification techniques in RS. The local data for each client were divided into training and test sets using the standard 80%–20% split.

Table I presents the approximate sizes of the masked image datasets (in GB) for the AID and RESISC45 datasets across varying masking ratios, ranging from 0.0 (no masking) to 0.9 (high masking). For the AID dataset, the total size of unmasked images (masking ratio 0.0) is 12.240 GB, and each client manages 1.224 GB after partitioning. As the masking ratio increases, the dataset size decreases substantially because a larger portion of each image is masked. At the highest masking ratio (0.9), the dataset size reduces to 1.224 GB, with only 0.122 GB per client—ten times smaller than the size per client for the unmasked dataset. A similar pattern is observed for the RESISC45 dataset. When the images are 90% masked, the dataset size for each client is reduced to one-tenth of the size compared to when no masking is applied. Specifically, the dataset decreases from 0.7 GB per client without masking to 0.07 GB per client with 90% masking.

This approach enables us to assess the scalability of our methods and analyze how different levels of masking affect the performance of distributed models in terms of storage, processing, and communication demands.

B. Experimental Configurations

The experiments are performed on a deep learning server equipped with NVIDIA A100 GPUs. A total of ten communication rounds are conducted, each round involving ten epochs per client to optimize their respective models using the Adam optimizer. The hyperparameters are configured as follows: the batch size is set to 16, the Adam optimizer uses a learning rate of 0.001, and the weight decay is also 0.001. After local training, these parameters are aggregated on the global server to update the global model. For FedAvg and FedProx, the total number of clients K is set to 10, with five clients participating in each round. The evaluation metric used is the average classification accuracy of the global model.

C. Experimental Results

In this section, we utilize the two previously described datasets and FL algorithms to examine how increasing the

masking ratio impacts model accuracy under each decentralization scenario. Tables II–IV showcase the performance of the two algorithms, FedAvg and FedProx, across three decentralization scenarios, namely, DS1, DS2, and DS3 under varying masking ratios. The masking ratio indicates the proportion of data concealed from the clients. While it is generally expected that accuracy decreases as the masking ratio increases, the results reveal an intriguing trend: in several instances, accuracy either remains constant or even improves as the masking ratio grows, particularly within certain decentralization scenarios and datasets.

In DS1 (see Table II), where data distribution is more balanced across clients, both FedAvg and FedProx demonstrate an increase in accuracy as the masking ratio rises from 0.0 to 0.6, particularly with the RESISC45 dataset. For FedAvg with RESISC45, accuracy improves from 49.27% at a masking ratio of 0.0 to 76.77% at 0.6. This unexpected performance enhancement may stem from the algorithm's ability to generalize better when part of the data is masked, potentially driving the model to learn more robust features. Similarly, FedProx exhibits a steady increase in accuracy on RESISC45 from 45.05% at 0.0 masking to 78.49% at 0.6 masking. In contrast, for the AID dataset, there is no significant improvement in accuracy with higher masking ratios, but the performance remains quite stable for both methods. For instance, FedAvg's accuracy fluctuates between 83% and 88%, while FedProx consistently achieves over 89%, regardless of the increasing masking ratios.

DS2 (see Table III) introduces a more uneven data distribution across clients, yet similar trends are observed where the masking ratio does not produce the anticipated drop in accuracy. For example, in FedAvg with the RESISC45 dataset, accuracy improves from 21.20% at a masking ratio of 0.0 to 58.91% at a ratio of 0.6, after which it begins to decline. This indicates that partial data masking, up to a certain threshold, can promote the algorithm's ability to learn more generalized patterns from the data. FedProx exhibits a similar pattern, with accuracy increasing from 23.32% at 0.0 masking to 56.52% at 0.6 masking. On the AID dataset, accuracy generally improves as the masking ratio increases. For instance, FedAvg's accuracy rises from 54.55% at 0.0 masking to 73.74% at 0.4 masking before showing a slight decline.

Finally, DS3 (see Table IV) represents the most extreme case of data imbalance, resulting in lower overall accuracy for both algorithms. Nevertheless, there are cases where masking does not immediately degrade performance. For instance, with RESISC45, FedAvg achieves a notable accuracy increase from 7.53% at a 0.0 masking ratio to 19.11% at 0.4 masking. This suggests that masking can be beneficial in highly imbalanced scenarios by encouraging the model to extract more meaningful information from limited data. FedProx demonstrates a similar trend.

Figs. 2 and 3 visualize the global model accuracy across ten communication rounds for the three different decentralization scenarios under masking ratios of 0.5 and 0.9 for the AID dataset. In most scenarios, the accuracy improves as communication rounds progress, indicating that more communication rounds allow the model to better aggregate and refine knowledge. This suggests that extending the number of communication rounds could further enhance model performance.

An experiment was conducted to evaluate the effect of including all clients in each FL round instead of a subset. For the AID dataset under the DS2 scenario with FedProx,

TABLE II
COMPARISON OF METHODS BASED ON ACCURACY METRIC FOR DS1

Masking ratio	0.0	0.2	0.4	0.6	0.8
FedAvg + AID	83.93	82.44	**88.86**	85.70	81.78
FedProx + AID	**90.05**	88.95	87.41	87.19	81.21
FedAvg + RESISC45	49.27	23.43	**76.77**	74.98	66.90
FedProx + RESISC45	45.05	26.32	**78.49**	76.33	63.52

TABLE III
COMPARISON OF METHODS BASED ON ACCURACY METRIC FOR DS2

Masking ratio	0.0	0.2	0.4	0.6	0.8
FedAvg + AID	54.55	55.75	**73.74**	63.21	45.54
FedProx + AID	**62.34**	59.24	61.74	55.39	60.73
FedAvg + RESISC45	21.20	10.10	58.91	**60.31**	47.65
FedProx + RESISC45	23.32	13.46	**56.52**	55.20	44.03

TABLE IV
COMPARISON OF METHODS BASED ON ACCURACY METRIC FOR DS3

Masking ratio	0.0	0.2	0.4	0.6	0.8
FedAvg + AID	20.78	18.06	**21.56**	21.33	19.22
FedProx + AID	12.78	**20.60**	19.43	13.43	14.17
FedAvg + RESISC45	7.53	5.61	**19.11**	12.51	9.46
FedProx + RESISC45	8.27	5.66	9.96	10.14	**10.63**

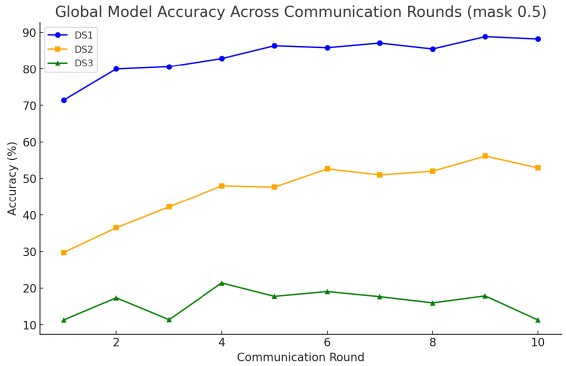

Fig. 2. Global model's accuracy across communication rounds for the AID dataset with 0.5 masking ratio using FedProx.

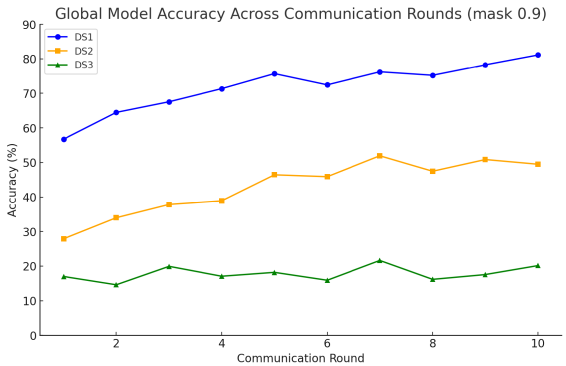

Fig. 3. Global model's accuracy across communication rounds for the AID dataset with 0.9 masking ratio using FedProx.

accuracy improved significantly when all ten clients participated in each round, achieving 69.34%, 72.64%, 67.07%, and 78.64% for masking ratios of 0.0, 0.2, 0.4, and 0.6, respectively. These results indicate that full client participation enhances model convergence and stability by reducing variance introduced by client selection and enabling a more comprehensive aggregation of local updates.

IV. Conclusion

This letter addressed the challenge of transferring large-scale, high-resolution RS images in distributed environments by using MAEs to generate embeddings, allowing the efficient transfer of image patches instead of entire images. Classifiers trained on these tokens, with model weights aggregated using FedAvg and FedProx, demonstrated robustness under IID and non-IID distributions. Experimental results showed significant bandwidth reduction while maintaining high classification accuracy.

The observed increase in accuracy with higher masking ratios, particularly in scenarios such as DS1 and DS2 where the data are more balanced or moderately unbalanced, is a significant finding. This indicates that data masking, which often mimics missing or occluded information, can conditionally improve the models' generalization abilities by prompting them to concentrate on the most essential and resilient features.

The results reveal an intriguing relationship between data sparsity and accuracy in FL environments. Contrary to expectations, increasing the masking ratio does not always lead to a decline in performance; in some cases, it can even improve accuracy. This underscores the potential of leveraging data sparsity in decentralized setups to reduce bandwidth and computational costs without significantly degrading, or even improving, model performance. These findings suggest that selective data masking may function as a regularization technique, helping to prevent overfitting and improve generalization, particularly in heterogeneous distribution scenarios.

While our main objective is to enhance the efficiency of FL in RS applications, we also acknowledge the importance of privacy considerations. Although our approach inherently reduces the transmission of raw image data, FL by itself does not offer strong privacy guarantees. To mitigate this, privacy-preserving techniques such as differential privacy [12] and secure aggregation [13] can be incorporated into our framework. These methods help safeguard sensitive data by introducing controlled noise or encrypting model updates, thereby enhancing privacy protection.

Future research could explore the integration of adaptive masking strategies and advanced aggregation techniques to optimize model performance further in diverse decentralized environments.

References

[1] M. Aspri, G. Tsagkatakis, and P. Tsakalides, "Distributed training and inference of deep learning models for multi-modal land cover classification," *Remote Sens.*, vol. 12, no. 17, p. 2670, Aug. 2020.

[2] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Process. Mag.*, vol. 37, no. 3, pp. 50–60, May 2020.

[3] B. Büyüktaş, G. Sumbul, and B. Demir, "Federated learning across decentralized and unshared archives for remote sensing image classification: A review," *IEEE Geosci. Remote Sens. Mag.*, vol. 12, no. 3, pp. 64–80, Sep. 2024.

[4] S. Moreno-Álvarez, M. E. Paoletti, A. J. Sanchez-Fernandez, J. A. Rico-Gallego, L. Han, and J. M. Haut, "Federated learning meets remote sensing," *Expert Syst. Appl.*, vol. 255, Dec. 2024, Art. no. 124583.

[5] B. Büyüktaş, K. Weitzel, S. Völkers, F. Zailskas, and B. Demir, "Transformer-based federated learning for multi-label remote sensing image classification," in *Proc. IGARSS-IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2024, pp. 8726–8730.

[6] X. Zhang, B. Zhang, W. Yu, and X. Kang, "Federated deep learning with prototype matching for object extraction from very-high-resolution remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, pp. 1–16, 2023, Art. no. 5603316, doi: 10.1109/TGRS.2023.3244136.

[7] B. Zhang, X. Zhang, M.-O. Pun, and M. Liu, "Prototype-based clustered federated learning for semantic segmentation of aerial images," in *Proc. IGARSS-IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2022, pp. 2227–2230.

[8] S. Wang et al., "Personalized multiparty few-shot learning for remote sensing scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, pp. 1–15, 2024, Art. no. 4506115, doi: 10.1109/TGRS.2024.3386978.

[9] B. Büyüktaş, G. Sumbul, and B. Demir, "Learning across decentralized multi-modal remote sensing archives with federated learning," in *Proc. IGARSS-IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2023, pp. 4966–4969.

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

[11] E. Kariotakis, G. Tsagkatakis, P. Tsakalides, and A. Kyrillidis, "Leveraging sparse input and sparse models: Efficient distributed learning in resource-constrained environments," in *Proc. Conf. Parsimony Learn.*, 2024, pp. 554–569.

[12] A. Banse, J. Kreischer, and X. Oliva i Jürgens, "Federated learning with differential privacy," 2024, *arXiv:2402.02230*.

[13] H. Fereidooni et al., "SAFELearn: Secure aggregation for private FEderated learning," in *Proc. IEEE Secur. Privacy Workshops (SPW)*, May 2021, pp. 56–62.

[14] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. 20th Int. Conf. Artif. Intell. Statist.*, 2017, pp. 1273–1282.

[15] T. Li, A. Kumar Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," 2018, *arXiv:1812.06127*.

[16] P. Han, S. Wang, and K. K. Leung, "Adaptive gradient sparsification for efficient federated learning: An online learning approach," 2020, *arXiv:2001.04756*.

[17] J. Koneč ný, H. Brendan McMahan, F. X. Yu, P. Richtárik, A. Theertha Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," 2016, *arXiv:1610.05492*.

[18] X. Sun, X. Ren, S. Ma, and H. Wang, "MeProp: Sparsified back propagation for accelerated deep learning with reduced overfitting," 2017, *arXiv:1706.06197*.

[19] N. Goli and T. M. Aamodt, "ReSprop: Reuse sparsified backpropagation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1545–1555.

[20] A. Kolesnikov et al., "An image is worth 16×16 words: Transformers for image recognition at scale," 2021, *arXiv:2010.11929*.

[21] H. Zhu, J. Xu, S. Liu, and Y. Jin, "Federated learning on non-IID data: A survey," *Neurocomputing*, vol. 465, pp. 371–390, Nov. 2021.

[22] G.-S. Xia et al., "AID: A benchmark data set for performance evaluation of aerial scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 3965–3981, Jul. 2017.

[23] G. Cheng, J. Han, and X. Lu, "Remote sensing image scene classification: Benchmark and state of the art," *Proc. IEEE*, vol. 105, no. 10, pp. 1865–1883, Oct. 2017.