FedRS-Net: A Federated Learning Approach for Collaborative Multi-Modal Maritime Analytics

Bole Wilfried Tienin
School of Information and Communication Engineering
University of Electronic Science and Technology of China
Chengdu, China
tieninwilfried@std.uestc.edu.cn

Guolong Cui
School of Information and Communication Engineering
University of Electronic Science and Technology of China
Chengdu, China
cuiguolong@uestc.edu.cn

Yannick Abel Talla Nana
School of Information Science and Technology
Southwest Jiaotong University
Chengdu, China
ytallanana@my.swjtu.edu.cn

Chiagoziem Chima Ukwuoma
College of Nuclear Technology and Automation Engineering
Chengdu University of Technology
Chengdu, China
chiagoziem.chima@zy.cdut.edu.cn

Roldan Mba Esidang
School of Information and Communication Engineering
University of Electronic Science and Technology of China
Chengdu, China
mbaroldan@std.uestc.edu.cn

Mohammed Raouf Senouci
School of Computer Science and Engineering
University of Electronic Science and Technology of China
Chengdu, China
senoucimedraouf@gmail.com

***Abstract*—Ensuring the safety and security of our oceans demands a comprehensive Maritime Situational Awareness (MSA) strategy. However, this task has several challenges, including using multi-modal data, data sharing among agencies, privacy concerns, and bandwidth limitations. To address these challenges, this research introduced FedRS-Net. FedRS-Net is a federated deep learning framework that trains multi-modal remote sensing data without exposing client data. The system employs a communication-efficient federated averaging algorithm and a novel convolutional neural network architecture called Redesigned Skip Connection. It integrates synthetic aperture radar (SAR) and optical satellite imageries to achieve remarkable results. Extensive experiments were conducted on the maritime vessel datasets, resulting in a testing accuracy of 99.8%. Further, applying secure aggregation and momentum-based gradient compression reduced communication costs by 7%. FedRS-Net overcomes privacy concerns and facilitates agency collaboration by enabling collective maritime monitoring through decentralized data. This research provides a robust federated learning solution tailored for multi-modal remote sensing analytics applications.**

***Index Terms*—Federated Learning, Maritime Situational Awareness (MSA), Remote Sensing, Synthetic Aperture Radar (SAR), Optical Satellite Imagery, Multi-modal, Classification, Collaborative Data Analysis.**

I. Introduction

In today's interconnected world, the maritime domain remains a vast and dynamic space. As human activities continue to expand across this domain, from international shipping to offshore energy exploration, the need for comprehensive Maritime Situational Awareness (MSA) has never been more pressing. Essentially, MSA aims to offer a holistic and comprehensive insight into activities within the maritime sector, promoting increased safety, security, and efficiency. Automatic Identification System (AIS) plays a vital role in MSA, providing ships with critical information about their routes. By utilizing MSA and AIS together, ships can safely navigate and reach their destinations punctually. However, achieving optimal MSA is challenging. The maritime domain is vast, conditions are often challenging, and activities are diverse. Relying on a single data source, regardless of its sophistication, is insufficient. Instead, MSA is akin to assembling a puzzle with pieces drawn from diverse data modalities such as satellites, radars, and AIS [1]. While each source is crucial, they each offer a perspective of the maritime domain that, though detailed, has its inherent limitations. Consider, for instance, optical imagery. Through optical sensors, we can capture high-resolution visuals of vessel activities, delving into the visible spectrum to reveal deeper features of the targets [2]. However, its effectiveness is constrained by its dependence on external light sources, making nighttime a significant limitation [3]. Additionally, meteorological challenges like cloud cover amplify these constraints [4]. On the other end, we have Synthetic Aperture Radar (SAR) imagery. SAR's strength lies in its resilience; SAR sensors operate independently of light and weather conditions, ensuring consistent data acquisition [5]. Yet, in most scenarios, it lacks the direct visual clarity and interpretability that optical imagery offers [6]. Therefore, the fusion of optical and SAR data emerges as a compelling proposition. Consider a scenario wherein optical sensors provide detailed vessel classifications during daylight hours, complemented by SAR's continuous nighttime surveillance. This integration of multi-modal remote sensing data can thus provide more enhanced and reliable maritime monitoring capabilities [7,8]. However, realizing this vision presents challenges. Given the geopolitical mosaic of our world, different nations and agencies often operate their own remote sensing infrastructures, including satellites. Without a collaborative multi-modal analysis in place, each entity jealously tends to protectively hoard its data. The barriers to sharing images are manifold, ranging from strategic and legal considerations to the sheer technical challenges posed by bandwidth constraints. Federated learning has recently emerged as a promising approach that enables collaborative modeling without direct data exchange [9]. Within this perspective, rather than centralizing data, federated learning adopts a decentralized approach. This allows individual nodes to utilize their localized models on specific datasets and then share only the updates with a central server, bypassing the need for direct data exchange [10]. In [11], Jia et al. first demonstrated this for object detection using satellite SAR imagery. However, its efficacy in integrating diverse modalities crucial for MSA is yet to be fully explored. While traditional techniques like Federated Averaging have been groundbreaking, they have shown certain limitations in the field of remote sensing, particularly regarding reliability, communication efficiency, and the unique nature of remote sensing data for classification tasks. To address the existing challenges, we present FedRS-Net, a federated learning algorithm specifically developed for remote sensing classification tasks. Designed for a multi-client framework, FedRS-Net synergizes SAR and optical satellite data through a meticulously Convolutional Neural Network (CNN) architecture that integrates a Newly Redesigned Skip Connection approach (RSC). These connections effectively extract features from diverse data streams. But the innovation doesn't stop there. The FedRS-Net algorithm also incorporates advanced techniques, including secure aggregation and momentum-based gradient compression. These enhancements ensure both robustness in model outcomes and efficiency in operation. In summary, this work aims to tackle the practical research gaps that have inhibited the adoption of federated learning in MSA for ship classification thus far. Our contributions are highlighted as follows:

1) An introduction of FedRS-Net: a federated learning model for privacy-preserving fusion of heterogeneous maritime data sources and modalities for a classification task.
2) An introduction of a newly crafted network topology termed Redesigned Skip Connection (RSC).
3) Development of communication-efficient aggregation mechanisms specifically tailored for maritime environments.
4) Extensive experimental analysis across diverse conditions to evaluate convergence, fairness, and efficiency metrics.

The rest of the paper is structured as follows: Section II presents the proposed federated architecture and experimental framework. Section III introduces the problem formulation. Section IV details an experimental analysis of model performance. Lastly, Section V highlights essential insights and charts potential avenues for future research in this domain.

II. FedRS-Net: A Federated Learning Model for Remote Sensing

This section presents FedRS-Net architecture designed for collaborative multi-modal maritime data classification across multiple agencies and systems. As depicted in Fig. 1, the architecture comprises three Clients, each representing a unique agency with its specific infrastructure. For illustrative purposes, Client 1 might represent a naval establishment equipped with a synthetic aperture radar(SAR), while Client 2 is another naval establishment equipped with an optical satellite. Client 3 could be a maritime observation agency managing maritime vessel movement using both SAR and Optical satellite imageries. It's crucial to note that each client exclusively accesses its local datasets, ensuring data privacy and decentralization.

Our system accommodates heterogeneous, multi-modal remote sensing data via the federated framework. To illustrate, Client 1 might possess radar data (Dataset_A), whereas Client 2 is equipped with optical satellite imagery (Dataset_B), and Client 3 possesses both SAR and optical imageries (Dataset_C). The overarching global model is trained to adeptly fuse these varied modalities while ensuring data remains decentralized. This design enables agencies to collectively harness diverse sensors and infrastructure, maximizing mutual benefits.

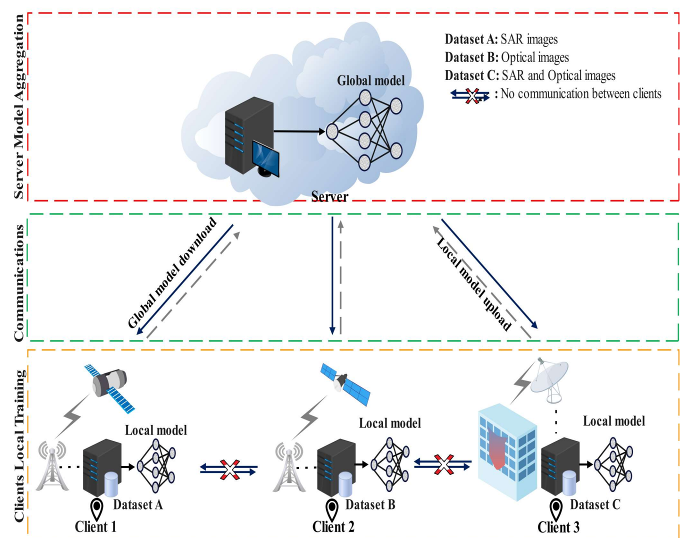

Fig. 1. Schematic representation of the federated learning architecture designed for multi-agency maritime data fusion.

A. The Proposed Federated Averaging Algorithm

The proposed Federated Averaging algorithm used in this work can be summarized as follows:

1) **Initialization:** The global model weights w_0 are initialized.
2) **Client Selection:** During each round, one client from each type of dataset (Dataset_A, Dataset_B, Dataset_C) is selected.

3) **Local Training:** Each selected client k computes an update to the model weights based on its local data using the equation:

$$w_{t+1}^{(k)} = w_t - \eta \nabla L_k(w_t) \quad (1)$$

where L_k is the local loss on client k's data, and η is the learning rate.

4) **Aggregation:** The updates from all the selected clients are aggregated to compute the global update by applying the equation below:

$$w_{t+1} = \frac{1}{m} \sum_{k=1}^{m} w_{t+1}^{(k)} \quad (2)$$

5) **Training loop:** Client Selection step, momentum-based gradient compression, and Aggregation step are repeated for a specified number of rounds, allowing the global model to learn from the entire federation of clients.

Below is Algorithm 1, summarizing the steps involved in the implementation of FedRS-Net.

Algorithm 1 Proposed FedRS-Net to train a Federated Learning model for Remote Sensing Ship image classification

---

Input: K (total number of clients),
C (number of class),
E (number of local epochs),
B (local batch size),
η (learning rate),
w (initial global model weights)
C_D (client_dataset)
Initialize: Global model weights $w_0 \leftarrow w$
for each round $t = 1, 2, \ldots, \text{Num\_Rounds}$ **do**
  $m \leftarrow \max(CE, 1)$ *// At least one client is selected*
  $S_t \leftarrow$ (select one client from each type of datasets : Dataset_A, Dataset_B, and Dataset_C)
  Initialize: Agg_metrics with zeros (aggregated metrics)
  for each client $k \in S_t$ **do**
    Compute the update of the client's local data using federated averaging (fed_avg) which represents the federated averaging process:
    $w_{t+1}^{(k)}, \text{client\_metrics} \leftarrow \text{fed\_avg.next}(w_t, C_D)$
    Update aggregated metrics with client metrics:
    Agg_metrics ← Agg_metrics + client_metrics
  end for
  Average the aggregated metrics:
  Agg_metrics ← Agg_metrics $/m$
  Update the global model weights:
  $w_{t+1} \leftarrow \frac{1}{m} \sum_{k=1}^{m} w_{t+1}^{(k)}$
  Store aggregated_metrics for this round
end for
Output: Final model weights $w_{\text{NUM\_ROUNDS}}$,

B. The Proposed Client Selection for FedRS-Net

The proposed client selection algorithm ensures that in each round of federated learning, one client from each type of data (Dataset_A, Dataset_B, Dataset_C) is selected. This ensures a diverse representation of data in each round. This selection can be achieved by applying the equations below:

$$\text{Dataset\_A\_Client\_Num} = t \times \mod \left(\frac{\text{Num\_Clients}}{3}\right) \quad (3)$$

$$\text{Dataset\_B\_Client\_Num} = t \times \mod \left(\frac{\text{Num\_Clients}}{3}\right) \quad (4)$$

$$\text{Dataset\_C\_Client\_Num} = t \times \mod \left(\frac{\text{Num\_Clients}}{3}\right) \quad (5)$$

Algorithm 2 provides a summary of the steps used in client selection.

Algorithm 2 Client Selection

Input: Num_Clients (total number of clients),
Num_Rounds (total number of training rounds)
List of clients for each dataset: Dataset_A_Clients, Dataset_B_Clients, and Dataset_C_Clients
for each round $t = 1, 2, \ldots, \text{Num\_Rounds}$ **do**
  Compute the client number of each dataset (Dataset_A, Dataset_B, Dataset_C) using equations (3), (4), and (5)
end for
Output: Selected_Clients ← S_C Where,
S_C= $\left[(3), (4) + \frac{\text{Num\_Clients}}{3}, (5) + 2 \times \frac{\text{Num\_Clients}}{3}\right]$

The rationale behind this client selection strategy is to enable effective integration of heterogeneous data modalities during the training process. By selecting one client from each dataset type in each round, the global model is exposed to a diverse set of data, facilitating the fusion of complementary features from different modalities. This approach addresses the non-IID nature of the data distribution, where each client's dataset may exhibit distinct characteristics or distributions.

C. The Data Distribution

IID and non-IID are terms that refer to the distribution of data. For instance, for IID (Independent and Identically Distributed), each data point is independent of all other data points. Therefore, the outcome or label of one data point does not influence or provide any information about the outcome or label of another. And all data points come from the same probability distribution. However, a data distribution is considered non-IID (Non-Independent and Identically Distributed) when the data points depend on one another. For instance, in time-series data, the value at time t might depend on the value at time t-1. Additionally, when different subsets of the data come from different distributions, it can also be considered a non-IID distribution. This is common in scenarios like federated learning, where data is distributed across multiple devices or clients, and each device might have data with different characteristics. Fig. 2 illustrates the differences between the two distributions and presents how FedRS-Net handles the data for each distribution.

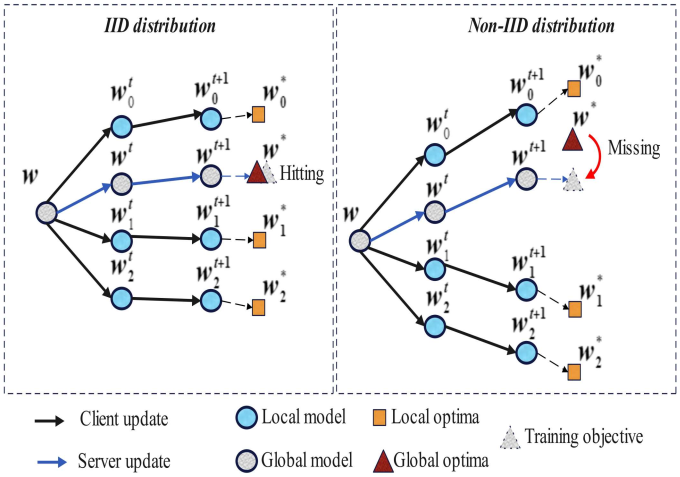

Fig. 2. Illustration of three clients drift in FedRs-Net of IID and non-IID data distribution.

D. Proposed Redesigned Skip Connection (RSC)

To effectively handle the challenges posed by non-IID data distributions, FedRS-Net introduces a novel network architecture called Redesigned Skip Connection (RSC). This architecture is specifically tailored to extract and fuse features from heterogeneous remote sensing data modalities, such as SAR and optical imagery. RSC comprises repeated stacked blocks made up of convolutional layers and skip connections, which bypass layers to avoid the vanishing gradient problem during training. In the implemented architecture, four blocks are utilized. The first block processes the input data to extract initial features. The second block refines these representations through further convolutional operations. The third block is tailored to identify specific-ship features in the data. Finally, the fourth block integrates the hierarchical features to prepare for classification. The configurations and parameter settings are summarized in Table 1.

TABLE I
CONFIGURATION OF THE REDESIGNED SKIP CONNECTION (RSC) MODEL

Block	Layer	Input Dimension	Output Dimension	Parameters
Block 1	Input	$256 \times 256 \times 3$	$256 \times 256 \times 3$	-
	Conv2D	$256 \times 256 \times 3$	$256 \times 256 \times 64$	1792
	Conv2D	$256 \times 256 \times 64$	$256 \times 256 \times 64$	4160
Block 2	MaxPooling2D	$256 \times 256 \times 64$	$128 \times 128 \times 64$	-
	Conv2D	$128 \times 128 \times 64$	$128 \times 128 \times 128$	73856
	Conv2D	$128 \times 128 \times 128$	$128 \times 128 \times 128$	16512
Block 3	Conv2D	$128 \times 128 \times 64$	$128 \times 128 \times 128$	8320
	Conv2D	$64 \times 64 \times 128$	$64 \times 64 \times 256$	295168
	Conv2D	$64 \times 64 \times 256$	$64 \times 64 \times 256$	65792
	MaxPooling2D	$64 \times 64 \times 256$	$32 \times 32 \times 256$	-
Block 4	Conv2D	$32 \times 32 \times 256$	$30 \times 30 \times 512$	1180160
	GlobalAveragePooling2D	$30 \times 30 \times 512$	512	-
Output	Dense	512	256	131328
	Dense	256	3	771

E. Training Loop

The main training loop runs for a specified number of rounds (Num_Rounds). In each round, one client from each type of dataset is selected. The selected client datasets are used to update the global model using the proposed FedRS-Net algorithm. The metrics (loss and accuracy) for each client are recorded and aggregated. The aggregated metrics are printed after each round. In addition, a Momentum-based gradient compression has been implemented. This technique leverages the momentum term in optimization algorithms like SGD (Stochastic Gradient Descent) with momentum. For our scenario, instead of sending the full gradients, each client will send a compressed version of the gradient based on the difference between the current gradient and the accumulated momentum. The following steps described how the Momentum-based gradient compression was implemented in our proposed FedRS-Net:

Each client computes the gradient on its local data and updates its local momentum term:

$$g_b = \nabla L(\theta; b) \quad \text{(Gradient computation on batch } b\text{)} \tag{6}$$

$$v_c = \mu v_c + g_b \quad \text{(Local momentum update)} \tag{7}$$

The difference between the current gradient and the accumulated momentum is computed and then compressed:

$$\Delta g_b = g_b - \mu v_c \quad \text{(Gradient difference computation)} \tag{8}$$

$$\Delta \hat{g}_b = \text{Compress}(\Delta g_b) \quad \text{(Gradient difference compression)} \tag{9}$$

The compressed gradient difference is sent to the server:

$$\text{Send to server: } \Delta \hat{g}_b \tag{10}$$

The server aggregates the received compressed gradients from all clients, decompresses them, and then updates the global model parameters using the aggregated gradient and the global momentum term:

$$G = \sum_{c=1}^{C} \Delta \hat{g}_b^c \text{(Aggregation of compressed gradients)} \tag{11}$$

$$\hat{G} = \text{Decompress}(G) \text{(Decompression of aggregated gradient)} \tag{12}$$

$$v = \mu v + \hat{G} \quad \text{(Global momentum update)} \tag{13}$$

$$\theta = \theta - \eta v \quad \text{(Global model parameter update)} \tag{14}$$

Where: L is the loss function, θ represents the model parameters, g_b is the gradient computed on batch b, v_c is the local momentum term for client c, μ is the momentum coefficient, η is the learning rate, C is the number of clients. The functions Compress () and Decompress() represent the compression and decompression operations, respectively. Algorithm 3 introduced the steps of implementing FedRS-Net with the Momentum-based Gradient Compression strategy.

Algorithm 3 FedRS-Net training loop with Momentum-based Gradient Compression algorithm

Require: Model parameters θ_0, number of rounds T, number of clients C, momentum coefficient μ, learning rate η
Ensure: Trained model parameters θ
  Initialize global model parameters $\theta \leftarrow \theta_0$
  Initialize global momentum term $v \leftarrow 0$
  for $t = 1$ to T **do**
    for each client c in parallel **do**
      Load local data D_c
      Initialize local momentum term $v_c \leftarrow 0$
      for each batch b in D_c **do**
        Compute gradients g_b on batch b
        Update local momentum: $v_c \leftarrow \mu v_c + g_b$
        Compute gradient difference: $\Delta g_b \leftarrow g_b - \mu v_c$
        Compress Δg_b to obtain $\hat{\Delta g_b}$
        Send $\hat{\Delta g_b}$ to server
      end for
    end for
    Server aggregates received compressed gradients: $G \leftarrow \text{Aggregate}(\{\hat{\Delta g_b}\}_{\text{all clients}})$
    Decompress G to obtain $\hat{G}$
    Update global momentum: $v \leftarrow \mu v + \hat{G}$
    Update global model parameters: $\theta \leftarrow \theta - \eta v$
  end for

III. Problem Formulation

Let K be the total number of clients, D_k is the local dataset containing images x_i and labels y_i from C classes. The goal is to train an image classification model with parameters θ in a federated manner to minimize the overall loss:

$$\min_{\theta} L(\theta) \tag{15}$$

where the global loss $L(\theta)$ is the weighted average of local loss functions $L_k(\theta)$:

$$L(\theta) = \frac{1}{K} \sum_{k} L_k(\theta) \tag{16}$$

Each client k trains the model locally using its dataset D_k by minimizing its local loss:

$$\min_{\theta} L_k(\theta) \tag{17}$$

The local loss $L_k(\theta)$ for client k is computed as the average categorical cross-entropy loss over a batch B_k sampled from D_k:

$$L_k(\theta) = \frac{1}{|B_k|} \sum_{(x_i,y_i)\in B_k} \text{CategoricalCrossEntropy}(f(x_i;\theta), y_i) \tag{18}$$

where $f(x;\theta)$ represents the Redesigned Skip Connection model (RSC) that maps images x to class probabilities.

In each federated round t, a subset of clients $K_t \subseteq K$ participate and compute updates $\Delta\theta_k$ that minimize their local loss $L_k(\theta_t)$. The server aggregates these updates to improve the global model:

$$\theta_{t+1} = \theta_t + \eta \sum_{k\in K_t} \Delta\theta_k \tag{19}$$

The process continues until the global loss on validation data converges. The key constraints are:

- Clients only share model updates $\Delta\theta_k$, not images.
- The global model $f(x;\theta_t)$ should perform well on new test data.
- Communication costs should be minimized.

FedRS-Net addresses these constraints to efficiently learn a robust classifier from decentralized heterogeneous maritime datasets. Fig. 3 summarizes FedRS-Net architecture.

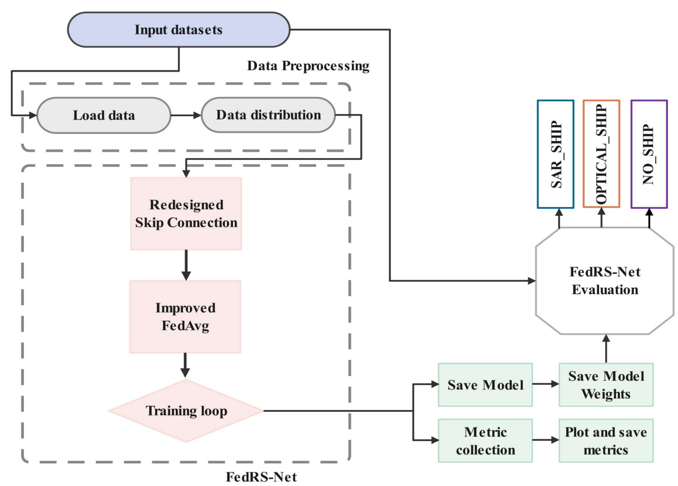

Fig. 3. Overview of FedRs-Net FlowChart.

IV. Experiments and Results

This section delineates the experiments conducted on the proposed FedRS-Net model.

A. Datasets: Dataset_A, Dataset_B, and Dataset_C

In this investigation, three datasets, namely, Dataset_A, Dataset_B, and Dataset_C, were used for the classification task. In this manuscript, Dataset_A refers to SAR_SHIP, Dataset_B refers to OPTICAL_SHIP, and Dataset_C refers to NO_SHIP. Dataset_A images were sourced online from the "multi-source and multi-scale SAR ship slice dataset" [12]. Dataset_B images were sourced from the Airbus Ship Detection dataset [13], which contained satellite optical images. Dataset_C is obtained by collecting images of the two modalities, SAR imageries from the MSAR-1.0 dataset[14,15] and optical satellite imageries from WHU-RS19 dataset[16]. Fig.4. gives an overview of images in the dataset, and as presented in Table II, 4962 images and 990 images were used for training and testing.

Fig. 4. Visualization of the datasets (SAR_Ship class is marked red, OPTICAL_Ship class is marked green while NO SHIP class is marked blue).

TABLE II
DATASETS SPLITTING DETAILS.

Class / Client	Training set	Testing set
Dataset_A (SAR_SHIP)	1654	330
Dataset_B (OPTICAL_SHIP)	1654	330
Dataset_C (NO_SHIP)	1654	330
TOTAL	4962	990

B. Results

Three distinct training scenarios were evaluated based on the number of clients. Fig.5., Fig.6., and Fig.7. displayed the training curves of these scenarios. Table III provides insights into the relationship between client quantity and the model's evaluation metrics across all classes. The "3 Clients" column displays the metrics from training the model with just three clients. For the SAR ship class, a precision of 100% is recorded. As we move to the "6 Clients" column, there's a discernible decrease in performance, with SAR ship precision sliding to 99.09%. This decreasing trend persists as we analyze the "9 Clients" column, representing training with 9 clients, where SAR ship precision further declines to 98.80%. However, the Optical ship class exhibits the opposite trend. With 3 clients, the model achieves 99.40% Optical ship precision. Increasing to 6 clients leads to a noticeable improvement, with precision rising to 99.70%. Finally, utilizing 9 clients results in the highest precision of 100% for the Optical ship class. This can be justified by the fact that the use of RSC may have varying effects across different data modalities. The RSC might be more adept at capturing features relevant to optical images than SAR images as the dataset size grows with more clients. With an increasing number of clients, the complexity and variety of data increase. The global model might be limited and becomes more efficient at learning features that are more common or simpler, such as the optical images in this case.

Furthermore, the communication cost is calculated by considering the size of the model updates that are transmitted during each round of the learning process. For our federated approach, the model, comprising 1,777,859 parameters, initially requires sending and receiving an uncompressed payload of approximately 6.78 MB from each client. Considering a typical 32-bit floating-point representation for each parameter, this size is derived from the model's parameter count multiplied by the size of each parameter in bytes, further divided by 1024^2 to convert from bytes to megabytes. Table IV shows the communication cost of FedRS-Net.

TABLE IV
COMMUNICATION COST OF FEDRS-NET

	3 Clients	**6 Clients**	**9 Clients**
Model Size (MB)	6.7819	6.7819	6.7819
Compressed Size (MB)	6.3072	6.3072	6.3072
Comm. Cost Uncompressed (GB)	139.0838	278.1677	417.2515
Comm. Cost Compressed (GB)	129.3479	258.6959	388.0439

The total communication cost for the entire training process across 3500 rounds can be substantial. Still, with our momentum-based gradient compression algorithm strategy, we observe a significant reduction in the required data transmission. As presented in Fig.8. for 3 clients, the uncompressed communication cost is approximately 139.08 GB, whereas the compressed communication is just about 129.35 GB. As the number of clients increases to 6 and then to 9, the communication cost without compression would linearly increase to 278.17 GB and 417.25 GB, respectively. However, with compression, these costs are kept to a minimal 258.70 GB and 388.04 GB, showcasing the impact of our compression approach.

C. Discussion

This study presented FedRS-Net, a novel federated learning framework for multi-modal remote sensing image classification. We integrate a compression mechanism within our FedRS-Net to address the challenge of high communication costs. Our chosen Momentum-based Gradient Compression algorithm significantly reduces the payload size by compressing the gradient updates before they are sent to the server. FedRS-Net yielded a compression ratio of 7%. This compression substantially decreases the required bandwidth and can lead to faster training times and lower network load.

TABLE III
EVALUATION RESULTS OF THE THREE SCENARIOS

Class / Client	3 Clients			6 Clients			9 Clients			
	Precision	Recall	F1 score	Precision	Recall	F1 score	Precision	Recall	F1 score	Support
SAR ship	1	1	0.9999	0.9909	1	0.9954	0.9880	1	0.9940	330
Optical ship	0.9940	1	0.9970	0.9970	0.9879	0.9923	1	0.9939	0.9969	330
No ship	1	0.9939	0.9969	0.9880	0.9878	0.9879	0.9939	0.9879	0.9808	330
Accuracy			**0.9980**			**0.9920**			**0.9939**	990

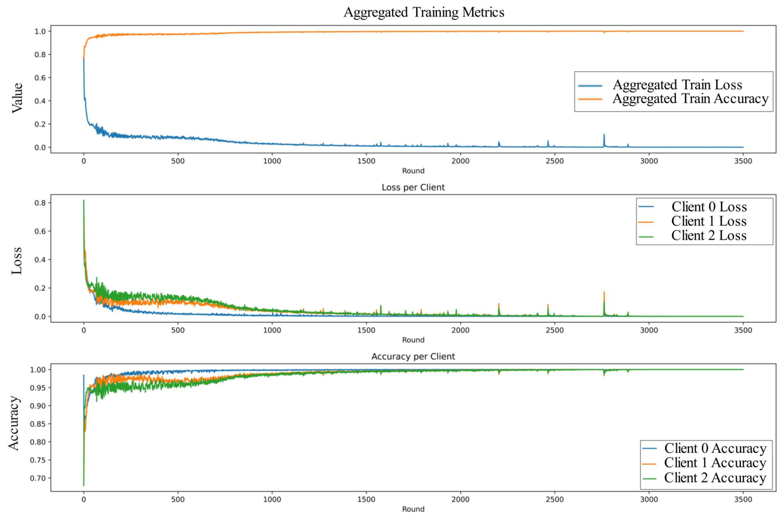

Fig. 5. The training curves of the proposed model (FedRS-Net) for federated learning on remote sensing image classification with 3 clients configuration.

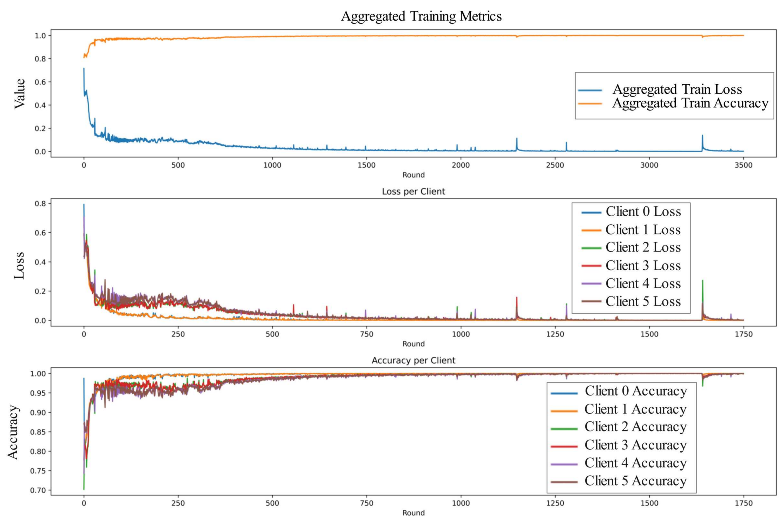

Fig. 6. The training curves of the proposed model (FedRS-Net) for federated learning on remote sensing image classification with 6 clients configuration.

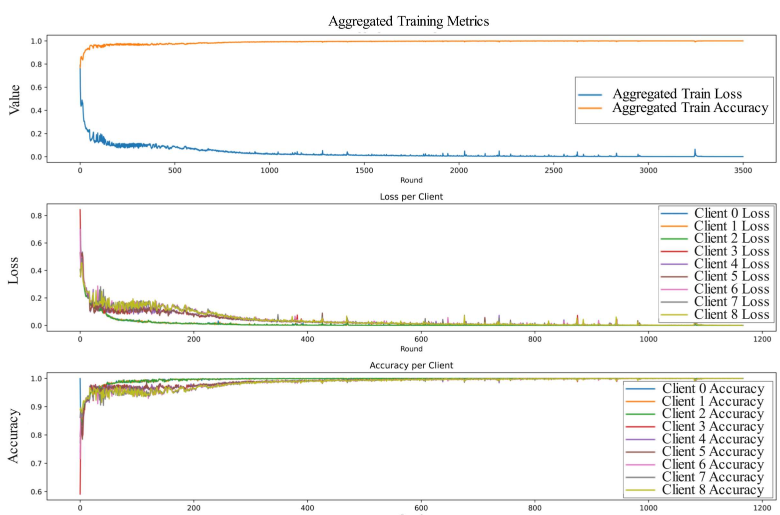

Fig. 7. The training curves of the proposed model (FedRS-Net) for federated learning on remote sensing image classification with 9 clients configuration.

The above results demonstrate FedRS-Net's ability to fuse SAR and optical satellite data in a privacy-preserving manner for robust maritime ship classification. FedRS-Net appears to be one of the first federated learning attempts for marine vessel classification using remote sensing data. Therefore, it seems difficult to compare our findings with some state-of-the-art federated learning models in remote sensing. However, it is worth noting that direct comparisons with federated learning techniques in the remote sensing domain are limited. Nevertheless, evaluating FedRS-Net against general remote sensing data classification methods could provide valuable insights into its relative performance. Table V provides a direct comparison of FedRS-Net and other classification methods.

TABLE V
COMPARISON OF FEDRS-NET WITH STATE-OF-THE-ART MODELS

Model	Accuracy (%)	F1 Score (%)	Source
ResNet-50	97.40	93.64	[17]
ResNet-101	96.20	93.90	[18]
Inception-V3	95.48	95.65	[19]
ResNet-152	95.80	95.81	[20]
FedRS-Net (Ours)	**99.80**	**99.79**	-

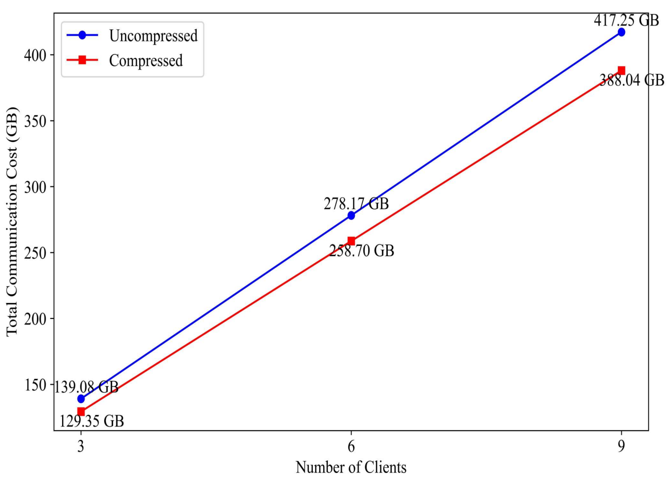

Fig. 8. Communication cost of compressed and uncompressed FedRS-Net.

V. CONCLUSION

This paper presents FedRS-Net, a federated learning framework uniquely designed for multi-modal remote sensing image classification, with an emphasis on the collaborative integration of SAR and optical data across agencies, all while ensuring data privacy. Our comprehensive evaluations, particularly on a maritime vessel dataset, demonstrated an impressive over 99.80% accuracy and 99.79% F1-score. When compared with existing solutions, FedRS-Net's tailored architecture and federated approach proved superior, outpacing state-of-the-art models by 2-3% in accuracy. While the current version of FedRS-Net showcases strong capabilities, there are opportunities for further improvement. Exploring peer-to-peer architectures and the potential integration of Blockchain-based transparency can pave the way for future research directions.

REFERENCES

[1] S. Thombre et al., "Sensors and AI Techniques for Situational Awareness in Autonomous Ships: A Review," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 1, pp. 64-83, 2022.

[2] Z. Zhang, L. Zhang, Y. Wang, P. Feng and R. He, "ShipRSImageNet: A Large-Scale Fine-Grained Dataset for Ship Detection in High-Resolution Optical Remote Sensing Images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 8458-8472, 2021.

[3] J. Si, B. Song, J. Wu, W. Lin, W. Huang and S. Chen, "Maritime Ship Detection Method for Satellite Images Based on Multiscale Feature Fusion," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 16, pp. 6642-6655, 2023.

[4] Z. Zhang, H. Zheng, J. Cao, X. Feng and G. Xie, "FRS-Net: An Efficient Ship Detection Network for Thin-Cloud and Fog-Covered High-Resolution Optical Satellite Imagery," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, pp. 2326-2340, 2022.

[5] B. W. Tienin and G. Cui, "A Convolutional Neural Network for Heterogeneous Ship Images Classification," *2021 CIE International Conference on Radar (Radar)*, Haikou, Hainan, China, 2021, pp. 1004-1008.

[6] B. W. Tienin, G. Cui and R. M. Esidang, "Comparative Ship Classification in Heterogeneous Dataset with Pre-trained Models," *2022 IEEE Radar Conference (RadarConf22)*, New York City, NY, USA, 2022, pp. 1-6.

[7] Z. Liu, X. Bai, C. Sun, F. Zhou and Y. Li, "Multi-Modal Ship Target Image Smoothing Based on Adaptive Mean Shift," *IEEE Access*, vol. 6, pp. 12573-12586, 2018.

[8] J. Liu, H. Chen, and Y. Wang, "Multi-Source Remote Sensing Image Fusion for Ship Target Detection and Recognition," *Remote Sensing*, vol. 13, no. 23, p. 4852, 2021.

[9] J. Mills, J. Hu and G. Min, "Multi-Task Federated Learning for Personalised Deep Neural Networks in Edge Computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 3, pp. 630-641, 2022.

[10] T. Li, A. K. Sahu, A. Talwalkar and V. Smith, "Federated Learning: Challenges, Methods, and Future Directions," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 50-60, 2020.

[11] Z. Jia, H. Zheng, R. Wang, and W. Zhou, "FedDAD: Solving the Islanding Problem of SAR Image Aircraft Detection Data," *Remote Sensing*, vol. 15, no. 14, p. 3620, 2023.

[12] Y. Wang, C. Wang, H. Zhang, et al., "A SAR Dataset of Ship Detection for Deep Learning under Complex Backgrounds," *Remote Sensing*, vol. 11, no. 7, p. 765, 2019.

[13] Airbus, "Airbus Ship Detection Challenge," Kaggle, 2019. [Online]. Available: https://www.kaggle.com/c/airbus-ship-detection. [Accessed: 21-April-2023].

[14] J. Chen, Z. Huang, R. Xia, B. Wu, L. Sheng, L. Sun, and B. Yao, "Large-Scale Multi-Class SAR Image Target Detection Dataset-1.0" *Journal of Radars*, 2022.

[15] R. Xia, J. Chen, Z. Huang, H. Wan, B. Wu, L. Sun, B. Yao, H. Xiang, and M. Xing, "CRTransSar: A Visual Transformer Based on Contextual Joint Representation Learning for SAR Ship Detection," *Remote Sensing*, vol. 14, p.1488, 2022.

[16] K. Chen, M. Wu, J. Liu, et al., "FGSD: A Dataset for Fine-Grained Ship Detection in High-Resolution Satellite Images," *arXiv preprint arXiv:2003.06832*, 2020.

[17] W. Bao, M. Huang, Y. Zhang, Y. Xu, X. Liu, and X. Xiang, "Boosting Ship Detection in SAR Images with Complementary Pretraining Techniques," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 8941–8954, 2021.

[18] D. Li, Q. Liang, H. Liu, Q. Liu, H. and Liu, and G. Liao, "A Novel Multidimensional Domain Deep Learning Network for SAR Ship Detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–13, 2022.

[19] Y. Wang, C. Wang, and H. Zhang, "Ship Classification in High-Resolution SAR Images Using Deep Learning of Small Datasets," *Sensors*, vol. 18, no. 9, p. 2929, 2018.

[20] L. A. Leonidas and Y. Jie, "Ship Classification Based on Improved Convolutional Neural Network Architecture for Intelligent Transport Systems," *Information*, vol. 12, no. 8, p. 302, 2021.